A Study on Textual Content Analysis in Handwritten Documents

Thesis submitted by

SAMIR MALAKAR

DOCTOR OF PHILOSOPHY (Engineering)

Department of Computer Science and Engineering,
Faculty Council of Engineering & Technology
Jadavpur University
Kolkata-700032, India

2019

JADAVPUR UNIVERSITY
KOLKATA-700032, INDIA

INDEX NO. 173/13/E

1. TITLE OF THE THESIS:

A Study on Textual Content Analysis in Handwritten Documents

2. NAME, DESIGNATION & INSTITUTION OF THE SUPERVISORS:

DR. MITA NASIPURI
PROFESSOR,
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,
JADAVPUR UNIVERSITY, KOLKATA-700032

DR. RAM SARKAR
ASSOCIATE PROFESSOR,
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,
JADAVPUR UNIVERSITY, KOLKATA-700032

3. LIST OF PUBLICATIONS:

a) JOURNAL:

i. **S. Malakar**, M. Ghosh, S. Bhowmik, R. Sarkar, and M. Nasipuri, "A GA based Hierarchical Feature Selection Approach for Handwritten Word Recognition," *Neural Comput. Appl.*, DOI: https://doi.org/10.1007/s00521-018-3937-8, (***Impact Factor: 4.213***)

ii. S. Bhowmik, **S. Malakar**, R. Sarkar, S. Basu, M. Kundu, and M. Nasipuri, "Off-line Bangla handwritten word recognition: a holistic approach," *Neural Comput. Appl.*, DOI: https://doi.org/10.1007/s00521-018-3389-1. (***Impact Factor: 4.213***)

iii. S. Sahoo, S. K. Nandi, S. Barua, Pallavi, S. Bhowmik, **S. Malakar**, and R. Sarkar, "Handwritten Bangla word recognition using negative refraction based shape transformation," *J. Intell. Fuzzy Syst.*, vol. 35, no. 2, pp. 1765–1777, 2018. DOI: https://doi.org/ 10.3233/JIFS-169712 (***Impact Factor: 1.426***)

iv. **S. Malakar**, M. Ghosh, R. Sarkar, and M. Nasipuri, "Development of a Two-Stage Segmentation-Based Word Searching Method for Handwritten Document Images," *J. Intell. Syst.*, 2018, DOI: https://doi.org/10.1515/jisys-2017-0384.

v. **S. Malakar**, P. Sharma, P. K. Singh, M. Das, R. Sarkar, and M. Nasipuri, "A Holistic Approach for Handwritten Hindi Word Recognition," *Int. J. Comput. Vis. Image Process.*, vol. 7, no. 1, pp. 59–78, 2017, DOI: https://doi.org/10.4018/IJCVIP.2017010104.

vi. S. Bhowmik, S. Polley, M. G. Roushan, **S. Malakar**, R. Sarkar, and M. Nasipuri, "A holistic word recognition technique for handwritten Bangla words," *Int. J. Appl. Pattern Recognit.*, vol. 2, no. 2, pp. 142–159, 2015, DOI: https://doi.org/10.1504/IJAPR.2015.069539.

vii. **S. Malakar**, D. Mohanta, R. Sarkar, N. Das, M. Nasipuri, and D. K. Basu, "A New Global Thresholding Approach for Document Image Binarization," *Int. J. Inf. Process.*, vol. 6, no. 2, pp. 48–59, 2011, ISSN : 0973-8215.

viii. **S. Malakar**, D. Mohanta, R. Sarkar, N. Das, M. Nasipuri, and D. K. Basu, "A Novel Noise-removal Technique for Document Images," *Int. J. Comput. Commun. Technol.*, vol. 2, no. 2–4, pp. 120–124, 2010, ISSN (ONLINE): 2231 – 0371, ISSN (PRINT): 0975 - 7449.

b) CONFERENCE:

i. A. Chatterjee, **S. Malakar**, R. Sarkar, and M. Nasipuri, "Handwritten Digit Recognition using DAISY Descriptor : A Study," in *Proceedings of Fifth International Conference on Emerging Applications of Information Technology (EAIT)*, 2018, pp. 1–4.

ii. S. Sahoo, S. K. Nandi, S. Barua, Pallavi, **S. Malakar**, and R. Sarkar, "Handwritten Bangla city name recognition using shape-context feature," in *Proceedings of the 6th International Conference on Frontiers in Intelligent Computing: Theory and Applications*, 2018, vol. 695, pp. 451–460.

iii. M. Ghosh, **S. Malakar**, S. Bhowmik, R. Sarkar, and M. Nasipuri, "Memetic algorithm based feature selection for handwritten city name recognition," in *Proceedings of International Conference on Computational Intelligence, Communications, and Business Analytics*, 2017, pp. 599–613.

iv. S. Barua, **S. Malakar**, S. Bhowmik, R. Sarkar, and M. Nasipuri, "Bangla handwritten city name recognition using gradient-based feature," in *Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications*, Springer, 2017, pp. 343–352.

v. S. Bhowmik, **S. Malakar**, R. Sarkar, and M. Nasipuri, "Handwritten bangla word recognition using elliptical features," in *Proceedings of 6th International Conference on Computational Intelligence and Communication Networks*, 2014, pp. 257–261.

vi. S. Bhowmik, M. G. Roushan, S. Polley, **S. Malakar**, R. Sarkar, and M. Nasipuri, "Handwritten Bangla Word Recognition using HOG Descriptor," in *Proceedings of Fourth International Conference on Emerging Applications of Information Technology (EAIT),* 2014, pp. 193-197.

vii. P. K. Singh, S. Mahanta, **S. Malakar**, R. Sarkar, and M. Nasipuri, "Development of a page segmentation technique for Bangla documents printed in italic style," in *Proceedings of 2nd International Conference on Business and Information Management.(ICBIM)*, 2014, pp. 120–125.

viii. **S. Malakar**, R. K. Das, R. Sarkar, S. Basu, and M. Nasipuri, “Handwritten and printed word identification using gray-scale feature vector and decision tree classifier,” in *Proceedings of International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA),* 2013, Procedia Technol., vol. 10, pp. 831–839.

ix. **S. Malakar**, S. Halder, R. Sarkar, N. Das, S. Basu, and M. Nasipuri, “Text line extraction from handwritten document pages using spiral run length smearing algorithm,” in *Proceedings of the International Conference on Communications, Devices and Intelligent Systems,* 2012, pp. 616–619.

x. **S. Malakar**, B. Seraogi, R. Sarkar, N. Das, S. Basu, and M. Nasipuri, “Two-stage skew correction of handwritten Bangla document images,” in *Proceedings of Third International Conference on Emerging Applications of Information Technology (EAIT),* 2012, pp. 303–306.

xi. R. Sarkar, S. Halder, **S. Malakar**, N. Das, S. Basu, and M. Nasipuri, “Text line extraction from handwritten document pages based on line contour estimation,” in *Proceedings of 3rd International Conference on Computing, Communication and Networking Technologies (ICCCNT 2012),* 2012, pp. 1–8.

xii. **S. Malakar**, P. Ghosh, R. Sarkar, N. Das, S. Basu, and M. Nasipuri, “An improved offline handwritten character segmentation algorithm for Bangla script,” in *Proceedings of 5th Indian International Conference on Artificial Intelligence*, 2011, pp. 71–90.

xiii. **S. Malakar**, D. Mohanta, R. Sarkar, N. Das, M. Nasipuri, and D. K. Basu, “Binarization of the noisy document images: A new approach,” in *Proceedings of International Conference on Information Processing*, 2011, pp. 511–520.

c) BOOK CHAPTER:

i. M. Ghosh, **S. Malakar**, S. Bhowmik, R. Sarkar, and M. Nasipuri, “Feature Selection for Handwritten Word Recognition Using Memetic Algorithm,” in *Advances in Intelligent Computing,* Springer, 2019, pp. 103–124.

4. LIST OF PATENTS: **None**

5. LIST OF PRESENTATIONS IN INTERNATIONAL/NATIONAL CONFERENCE:

i. "A Study on Content Retrieval from Handwritten Documents," in Doctoral Symposium under *Fourth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG 2013)* at IIT Jodhpur, Jodhpur, India during 18-22 December, 2013.

ii. “Handwritten and printed word identification using gray-scale feature vector and decision tree classifier,” in *International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA)* at Kalyani University, Kalyani, West Bengal, India during 27-28 September, 2013.

Certificate from the Supervisors

*This is to certify that the thesis entitled "*A Study on Textual Content Analysis in Handwritten Documents*" submitted by Shri Samir Malakar, who got his name registered on 2nd May 2013 for the award of Ph.D. (Engg.) degree of Jadavpur University is absolutely based upon his own work under the supervision of Prof. Mita Nasipuri and Dr. Ram Sarkar and that neither his thesis nor any part of the thesis has been submitted for any degree/diploma or any other academic award anywhere before.*

....
(**Dr. MITA NASIPURI**)
Professor, Dept. of CSE
Jadavpur University,
Kolkata-700032

....
(**Dr. RAM SARKAR**)
Associate Professor, Dept. of CSE
Jadavpur University,
Kolkata-700032

Dedicated to My Family

Acknowledgements

I am deeply indebted to many people who have contributed in making the completion of this thesis work possible. Firstly, I would like to express my earnest gratitude and respect to my supervisors, Prof. Mita Nasipuri and Dr. Ram Sarkar for their relentless guidance, tremendous support, endless care, invaluable suggestion, and constant monitoring during the course of this work.

I would also like to express my sincere respect to Prof. Mahantapas Kundu, Head, Department of Computer Science and Engineering, Jadavpur University; Prof. Subhadip Basu, Dr. Nibaran Das and Prof. Debotosh Bhattacharjee, Department of Computer Science and Engineering, Jadavpur University, for providing me constant inspiration, support and useful suggestions during the course of this thesis work. I am also thankful to the faculty and staff members of Department of Computer Science and Engineering, Jadavpur University for their support during my course of work.

I am also grateful to Mr. Showmik Bhowmik, PhD Scholar, Department of Computer Science and Engineering, Jadavpur University and Mr. Manosij Ghosh, Final year student, Bachelor of Engineering, Department of Computer Science and Engineering, Jadavpur University for their whole-hearted support and invaluable suggestions during the course of this research work.

I am also thankful to all my co-researchers Mr. Sajal Mahanta, Department of Computer Science and Technology, Acharya Prafulla Chandra Roy Polytechnic College; Dr. Pawan kumar Singh, Department of Computer Science and Engineering, Calcutta Institute of Technology; Md. Galib Roushan, Department of Computer Science and Technology, Dumkal Polytechnic; Mr. Sanjib Polley, Kolkata Police Directorate; Mr. Dheeraj Mohanta, Consultant, Cognizant Technology solutions Pvt. Ltd.; Mr. Bhagesh Seraogi, Software Engineer, ARC Document Solutions; Mr. Sougata Halder, Team Lead (App Dev), Ensim India Pvt. Ltd.; Mr. Rahul Kumar Das, Software Engineer, Tiatros Inc.; Mr. Pankaj Sharma, Software Engineer, ATOS India Pvt. Ltd.; Mr. Pankaj Kumar Singh, Associate Consultant, Capgemini; Ms. Maitrayee Das, Associate, PwC India; Ms. Shilpi Barua, Assistant Manager (IT), Oriental Bank of Commerce; Mr. Agneet Chatterjee, Final year student, Bachelor of Engineering, Department of Computer Science and Engineering, Jadavpur University; Mr. Samanway Sahoo, Mr. Subham Kumar Nandi, Mr. Sourav Barua and Ms. Pallavi, Final year students, Bachelor of Technology, Department of Computer Science and Technology, Indian Institute of Engineering Science and Technology, Shibpur, India; for their friendly support during my research work.

I am also indebted to the fellow researchers including Mr. Neelotpal Chakraborty, Mr. Suman Kumar Bera, Mr. Anup Kumar Halder, Mr. Nirmal Das, Mr. Sourav Pramanik, Mr. Bidhan Barai and Ms. Kaushiki Roy of Centre for Microprocessor Application for Training, Education and Research (CMATER) Laboratory, Jadavpur University and Mr. Shibaprasad Sen, Department of Computer Science and Engineering, Future Institute of Engineering and Management, Kolkata for constantly motivating me to accomplish the work.

During the span of this research work, I had to collect handwritten samples from different people to build the voluminous database. Without these handwritten samples the work could not been completed. I am thankful to each of the writers who has provided their writing samples altruistically. I am pleased to convey my gratitude to Mr. Mainak Acharya, Mr. Shobhan Nandi, Ms. Madhumita Pal, Ms. Srabanti Sarkar, Ms. Nilanjana Sahu, Ms. Jyotipriya Rao and Ms. Sunreeta Sen, who have contributed voluntarily during the course of preparation of the databases used in this thesis.

Finally, I owe my encompassing debt to my respected mother, elder brother, sister, niece and my friends whose love and sacrifice, continued support, encouragement, good wishes and inspiration enabled me to complete this thesis.

Place: Kolkata

Date: --

(SAMIR MALAKAR)

CONTENT

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

Handwriting is the process of drawing a set of artificial graphic marks representing the units of a specific language, which in general varies from one language to another, on a flat surface like paper, writing board and Personal Digital Assistants (PDAs), with a marking instrument like pen, pencil, joystick, finger, etc. The development of handwriting helps in recording the history, events, culture, literature, law, science, mathematics, and many more.

Before invention of modern printing technologies like printing press, typewriter, etc., people mainly used pen-and-paper for various types of documentation. Although handwriting undergoes continuous changes and adapts to cultural and technological advancements, still widespread acceptance of digital computers seemingly challenges the future of handwriting. However, in numerous situations, a pen together with a piece of paper or a small notepad is much more convenient than a keyboard. For example, in general, most of the students in a classroom still do not take notes on a notebook computer; rather rely on a pen and an exercise book for writing class notes.

In addition to these, although the technology has advanced today, a large section of the educated people still prefer age-old pen-and-paper media to keep notes of their activities, maintain daily accounts of income and expenditure etc. in written form. Most of the doctors also use pen-and-paper for listing the observations about their patients and prescribing medicines. Even, almost every activity of the offices and societies in developing countries involves papers, which are in the form of petition files, application forms, reports, letters etc. Also, some other sources of the handwritten documents are old manuscripts, personal records, medical reports and documents from educational institutes or offices, etc. All these facts imply the presence of huge amount of handwritten documents. It is also expected to see exponential growth of handwritten documents as the day passes.

Apart from these, penmanship or handwriting is a characteristic of an individual and writing style varies from person to person. Due to this unique property of individual's writing, signature is still considered as one of the significant authentication mark of a person while security is concerned in places like banking, educational institutes and offices. Individual's handwriting is adequately affected due to variations in their state of mind, mood, age, gender, behavior, place of dwelling, profession etc. These scenarios lead to the generation of research areas like writer identification [1-2], signature verification [3-5], writer's demographics classification [6-7], forensic study [8-9] and behavior analysis [10-12].

Generally, handwritten documents are prepared hastily and are not managed properly after their creation. The possible reason behind this may be either due to variation in individual's management skill and strategy or absence of globally accepted rules and regulations for such management. As a result, these handwritten documents get degraded as time passes. Even, searching of some important document(s), required in a rush, becomes almost impossible in manual mode. Moreover, due to lack of proper concern, some documents get misplaced or lost. Some confidential information may also be leaked during transcription of confidential document.

Not only this, the amount of spaces these documents are occupying is increasing with time. But, the advancement of office automation with the arrival of information technology has deeply influenced and enhanced the ability to create, store, manipulate, and retrieve electronic documents. Word processing applications allow creating and editing electronic documents easily. Also, compression of the documents' electronic representation allows efficient storage and transmission. For documents, whose entire life cycle is electronic, these tasks are relatively straightforward, and users have therefore been able to treat these documents as maintainable and searchable entities. However, for the documents which exist in paper form, especially the handwritten documents, converting them in an electronic format/ representation is not an easy task. Such documents need to be scanned to convert them into digital image form, which can be stored and manipulated by computers.

With the advent of technology, the economic feasibility of maintaining large databases of document images has increased. This, in turn, created a demand for accessing and manipulating the information contained within these images in an efficient way. The requirement may be fulfilled if the documents are kept in digital form or in image form with adequate indexing so that searching and managing the documents become easier.

The above discussion suggests that there is a demand for a system which, at least, can reasonably manage these ever increasing handwritten documents coming from different sources, and stored in image format for later use. For example, museum archives contain old fragile documents having scientific or historical or artistic value. Development of such a system is only possible when the textual content of these document images could be automatically analyzed as well as classified according to the context and indexed with proper keyword based tagging.

Unfortunately, analysis of textual content in these document images is not an easy task to accomplish. It is mainly because the size of a collection is often substantial and the current handwritten Optical Character Recognition (OCR) system works poorly with such a large lexicon size [13-14]. Considering these facts, here, during this thesis work an alternative attempt is made for classifying handwritten documents. This attempt skips the conversion of documents into machine encoded forms which is the main objective any OCR based system.

In a nutshell, the work presented in this thesis is mainly targeted at handwritten documents written in *Bangla* script. Here, *Bangla* script is chosen for various reason: firstly for its richness and secondly for its character shape complexity over *Roman* script. Thirdly the works available in the literature on analysis of textual contents in handwritten *Bangla* documents are not significant enough. Fourthly *Bangla* is an important language of eastern and north eastern India [15]. It is the second most popular language in India [16] and the seventh most popular language in the world [17]. Finally, most of techniques that are devised for *Bangla* script can be easily extended for other *Matra* based scripts like Devanagari, Syloti, Gurumukhi with required variations.

1.1 *Bangla* Script: An Overview

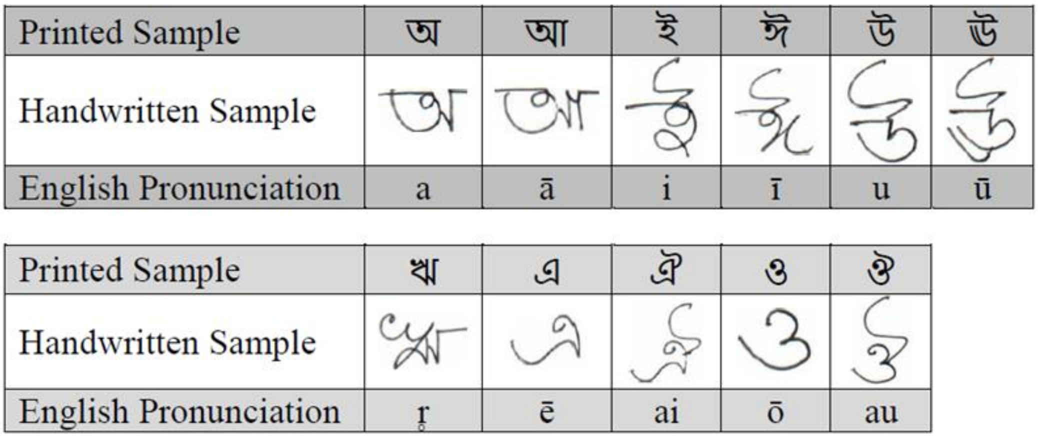
Bangla language is mostly used in the eastern and north eastern India and Bangladesh. Also, a substantial number of immigrant communities in the countries like Nepal, Singapore, United Kingdom, the United States [18] use *Bangla* language. It is also national language of Bangladesh. With more than 300 million native speaker of *Bangla* (around 262 million in Bangladesh [19] and 97 million in India [16]), it is the 7th most popular language in world [17, 20]. In addition to this, out of 22 constitution recognized regional languages [16] in India, *Bangla* is the second most used language, after Hindi written in Devanagari script.

Eastern Nagari script, which is the 5th most widely used writing system in the world [21], is composed of scripts like *Bangla*, Assamese and Purbi. The usage of Eastern Nagari script is associated with the two main languages *viz.*, Bengali and Assamese. In addition to these, many other languages such as Manipuri, Bodo, Karbi, Maithili and Angika etc. are also found to be written in this script in the past. Modern Sylheti is often written using this script as well. Bengali alphabet is mostly used in these scripts. Hence, considering the said findings, it can be concluded that *Bangla* is rich as a language as well as a script.

Apart from these, *Bangla* script (in *Bangla* “বাংলালিপি” (Bānlālipi)) [22] has a complex set of characters, which is used for writing and representing basic set of speech sounds of *Bangla* language. Depending on the nature of formation and usage, the graphemes that has been used in *Bangla* script can be divided into 5 categories *viz.*, a) fundamental letters, b) numerals, c) modifiers, d) compound characters, e) diacritical and punctuation marks. All these symbols taken together form *Bangla* alphabet set which is termed as “বাংলাবর্ণমালা” (Bānlābarṇamālā) in *Bangla*.

1.1.1 Basic Characters

The *Bangla* alphabet contains 50 basic characters that are also used in constructing the compound characters and modifiers in the script. Basic characters are also of two types *namely*, vowel and consonant, that are called as “স্বরবর্ণ” (swôrôbôrnô) and “ব্যঞ্জনবর্ণ” (Byañjanabarṇa) respectively in *Bangla* language. The total number of vowels and consonants in *Bangla* script are 11 and 39 respectively that have been shown in Fig. 1.1 and Fig. 1.2.



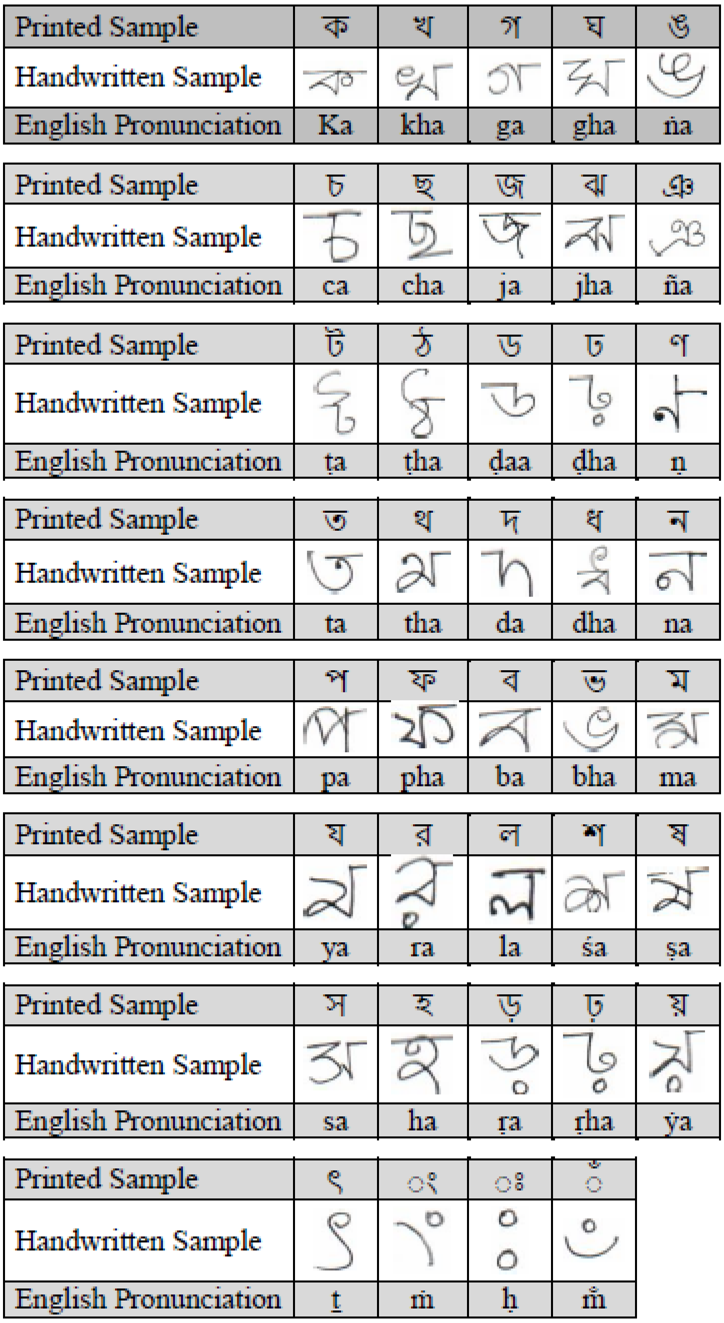
Printed Sample	অ	আ	ই	ঈ	উ	ঊ
Handwritten Sample	অ	আ	ই	ঈ	উ	ঊ
English Pronunciation	a	ā	i	ī	u	ū

Printed Sample	ঋ	এ	ঐ	ও	ঔ
Handwritten Sample	ঋ	এ	ঐ	ও	ঔ
English Pronunciation	ṛ	ē	ai	ō	au

Fig. 1.1 Vowels is Bangla alphabet, both in printed and handwritten forms, with their English pronunciation

1.1.2 Numerals

Bangla script has 10 numerical digits (graphemes indicating the numbers from 0 to 9) [23]. The numeral set is shown in Fig. 1.3 along with their handwritten and printed samples. Numbers larger than 9 are written in Bangla using a positional base 10 numeral system (the decimal system).



Printed Sample	ক	খ	গ	ঘ	ঙ
Handwritten Sample					
English Pronunciation	Ka	kha	ga	gha	ṅa

Printed Sample	চ	ছ	জ	ঝ	ঞ
Handwritten Sample					
English Pronunciation	ca	cha	ja	jha	ña

Printed Sample	ট	ঠ	ড	ঢ	ণ
Handwritten Sample					
English Pronunciation	ṭa	ṭha	ḍaa	ḍha	ṇ

Printed Sample	ত	থ	দ	ধ	ন
Handwritten Sample					
English Pronunciation	ta	tha	da	dha	na

Printed Sample	প	ফ	ব	ভ	ম
Handwritten Sample					
English Pronunciation	pa	pha	ba	bha	ma

Printed Sample	য	র	ল	শ	ষ
Handwritten Sample					
English Pronunciation	ya	ra	la	śa	ṣa

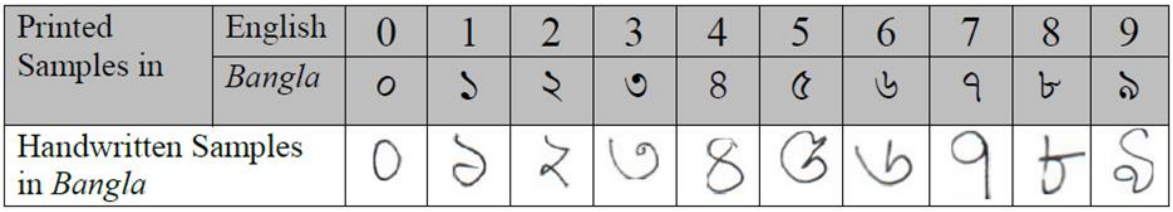
Printed Sample	স	হ	ড়	ঢ়	য়
Handwritten Sample					
English Pronunciation	sa	ha	ṛa	ṛha	ẏa

Printed Sample	ৎ	ং	ঃ	ঁ
Handwritten Sample				
English Pronunciation	ṯ	ṁ	ḥ	m̐

Fig. 1.2 Consonants in *Bangla* alphabet, both in printed and handwritten forms, with their English pronunciation

1.1.3 Modifiers

A vowel/consonant following a consonant takes a special shape without affecting the shape of the former consonant. These shapes are called modifiers. A total 13 modifiers are there in *Bangla* script out of which 10 are vowel modifiers and 3 are consonant modifiers. The vowel modifiers are read as vowel letter – “কার” (Kāra), e.g., “আ-কার” (Ā-kāra (◌া)) and “ই-কার” (I-kāra (ি◌)), “ঈ-কার” (Ī-kāra (◌ী)), whereas the consonant modifiers are read as “র-ফলা” (Ra-phalā), “য-ফলা” (Ya-phalā) and “রেফ” (Rēpha). These two categories of modifiers are shown in Fig. 1.4(a-b).



Printed Samples in	English	0	1	2	3	4	5	6	7	8	9
	Bangla	০	১	২	৩	৪	৫	৬	৭	৮	৯
Handwritten Samples in *Bangla*		০	১	২	৩	৪	৫	৬	৭	৮	৯

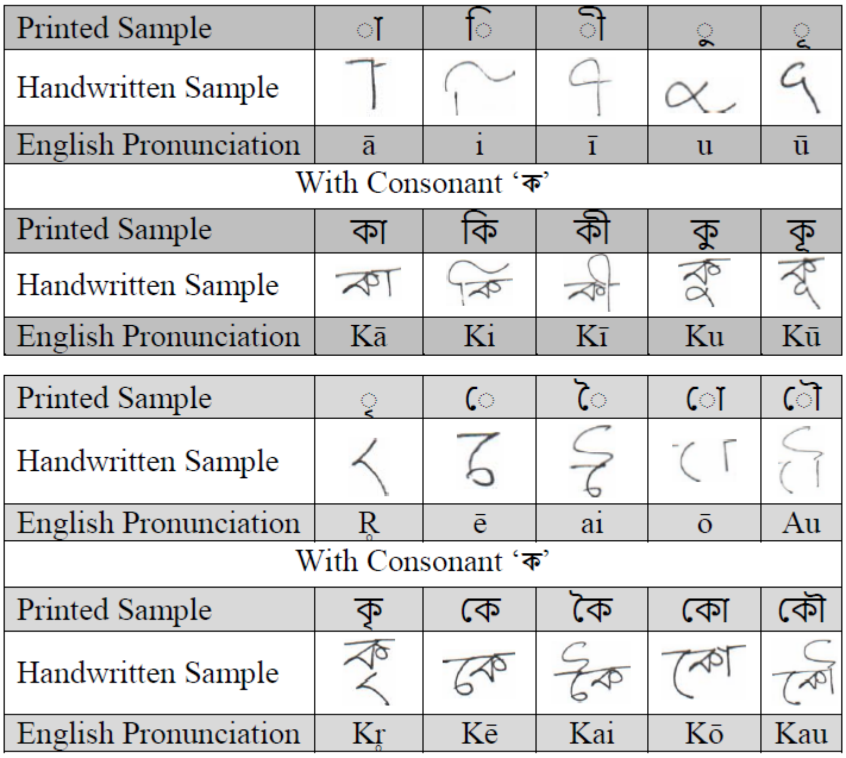
Fig. 1.3 Bangla numerals along with the corresponding symbols used in English

1.1.4 Compound Characters

Compound characters, in *Bangla*, that are read as “যুক্তাক্ষর” (Yuktākṣara), are complex shaped characters which consist of more than one consonant, occasionally followed by a vowel, and pronounced simultaneously. According to the work, described in [23], nearly 334 compound characters are found in *Bangla* script. A few examples of such characters are shown in Fig. 1.5, which also include the appearance sequence of the characters that construct the corresponding compound character.

1.1.5 Diacritical and Punctuation Marks

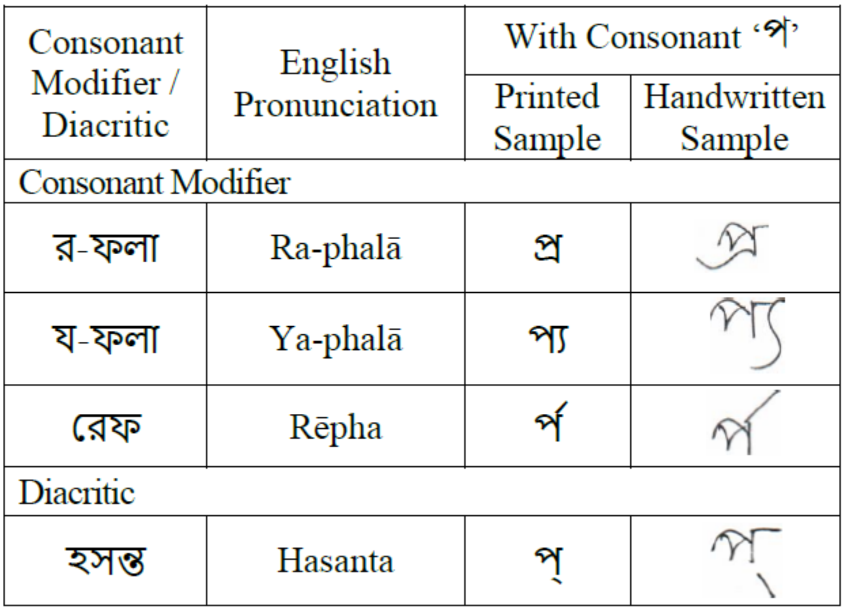
The grapheme ‘◌্’, read as “হসন্ত” (Hasanta) in *Bangla* language, appears beneath a basic letter, e.g., ‘ক্’, ‘খ্’, ‘গ্’and ‘ঘ্’, and presence of which indicates a difference in pronunciation from the same letter when remains unmarked. The usage of Hasanta in *Bangla* words is shown in Fig. 1.4(b). In addition, *Bangla* punctuation marks, except ‘।’ which is called as “দাড়ি” (Dāṛi) and is equivalent to a full stop in English language, are same as western scripts in terms of usage and shape.



Printed Sample	◌া	ি◌	◌ী	◌ু	◌ূ
Handwritten Sample	া	ি	ী	ু	ূ
English Pronunciation	ā	i	ī	u	ū
With Consonant 'ক'					
Printed Sample	কা	কি	কী	কু	কূ
Handwritten Sample	কা	কি	কী	কু	কূ
English Pronunciation	Kā	Ki	Kī	Ku	Kū

Printed Sample	◌ৃ	ে◌	ৈ◌	ে◌া	ে◌ৗ
Handwritten Sample	ৃ	ে	ৈ	ো	ৌ
English Pronunciation	R̥	ē	ai	ō	Au
With Consonant 'ক'					
Printed Sample	কৃ	কে	কৈ	কো	কৌ
Handwritten Sample	কৃ	কে	কৈ	কো	কৌ
English Pronunciation	Kr̥	Kē	Kai	Kō	Kau

(a) Vowel modifier



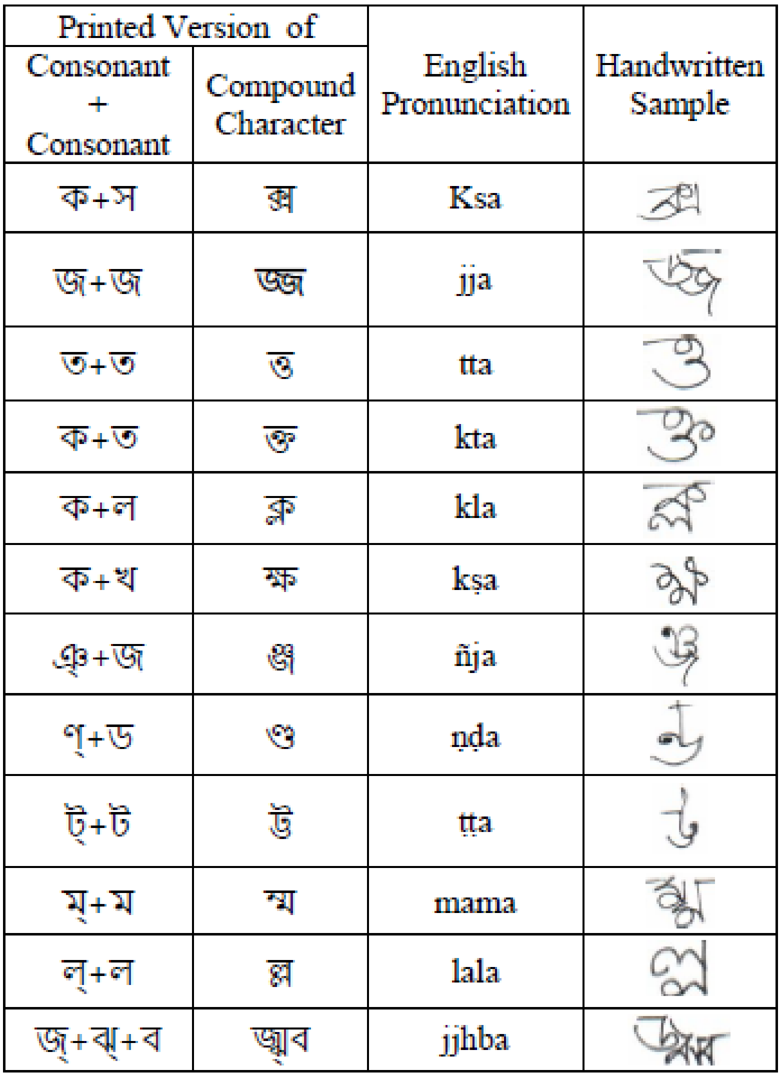
Consonant Modifier / Diacritic	English Pronunciation	With Consonant 'প'	
		Printed Sample	Handwritten Sample
Consonant Modifier			
র-ফলা	Ra-phalā	প্র	প্র
য-ফলা	Ya-phalā	প্য	প্য
রেফ	Rēpha	র্প	র্প
Diacritic			
হসন্ত	Hasanta	প্	প্

(b) Constant modifiers and diacritic

Fig. 1.4 Showing *Bangla* modifiers and diacritic symbols when appeared with consonant

1.1.6 *Bangla* Words

Characters as well as words in *Bangla* script are written from left to right and the concept of case sensitivity, i.e., upper or lower case, is missing. In addition, *Bangla* script has a unique characteristics, known as *Matra* or head line (in Bengali "মাত্রা" (Mātrā)), which isolates it from other Bramhic script like Odia, Gujarati, Telugu etc. [22]. *Matra* is a small horizontal line, not exceeding the maximum width of a character in length, which passes through the top portions of the characters. In a word, the *Matras* of consecutive characters are joined together to form a common *Matra* of the word. Also, certain characters or modifiers or diacritics have an elongated portion crossing over their *Matra*, which are known as ascendant, or appear below the base line of a characters, which are called descendants. The ascendant(s), descendant (s), common *Matra* and modifier(s) of a *Bangla* word image are illustrated in Fig. 1.6.



Printed Version of		English Pronunciation	Handwritten Sample
Consonant + Consonant	Compound Character		
ক+স	ক্স	Ksa	ক্স
জ+জ	জ্জ	jja	জ্জ
ত+ত	ত্ত	tta	ত্ত
ক+ত	ক্ত	kta	ক্ত
ক+ল	ক্ল	kla	ক্ল
ক+ষ	ক্ষ	kṣa	ক্ষ
ঞ+জ	ঞ্জ	ñja	ঞ্জ
ণ্+ড	ণ্ড	ṇḍa	ণ্ড
ট্+ট	ট্ট	ṭṭa	ট্ট
ম্+ম	ম্ম	mama	ম্ম
ল্+ল	ল্ল	lala	ল্ল
জ্+ঝ্+ব	জ্ঝ্ব	jjhba	জ্ঝ্ব

Fig. 1.5 Some of the *Bangla* compound characters. It also includes English pronunciation, printed graphics, constituent basic characters and one handwritten sample of compound character

Depending on the position of *Matra* and base line, a *Bangla* word image can be divided horizontally into 3 zones *viz.*, upper zone, middle zone and lower zone. The horizontal lines marking these 3 zones, namely, $R1, R2, R3$ and $R4$ and are defined as staring row of upper zone, ending row of upper or starting row of middle zone, ending row of middle zone or starting row of lower zone and ending row of lower zone respectively. The position of zones along with the zone defining horizontal lines are shown in Fig. 1.7.

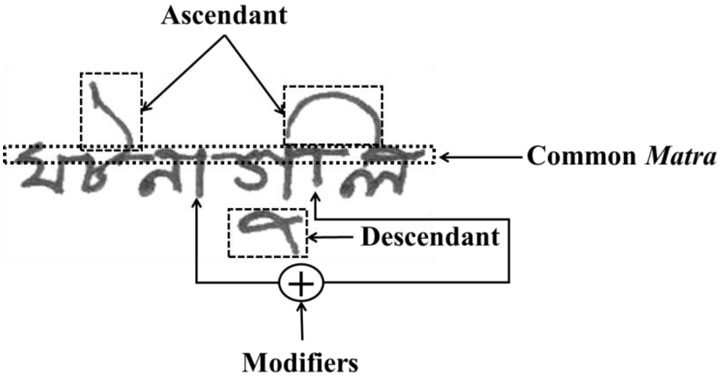


Fig. 1.6 The common *Matra*, two Ascendants, one Descendant and two Modifiers in a sample handwritten *Bangla* word image.

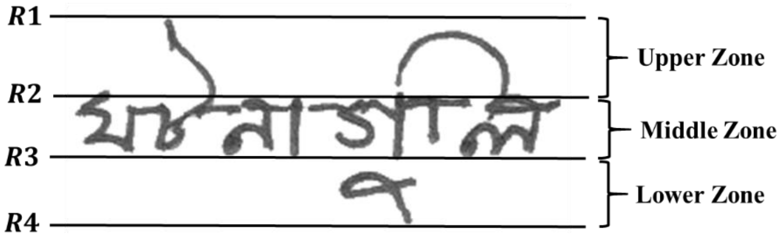


Fig. 1.7 Positions of four zonal boundaries and three zones of a handwritten *Bangla* word sample

1.2 Analysis of Textual Content in Handwritten Documents: An Overview

Handwritten documents, in general, contain noisy components, skewness at page, text line (TL) and word level, character shape and size variations, writing style variations among different individuals, even for the same individual. Such variations in handwritten documents not only restricts complete conversion of handwritten document into corresponding machine encoded form (i.e., OCRing of documents) but also, becomes a hindrance to intelligent analysis of the textual contents present therein. However, irrespective of script and language, a sufficient number of works that deal with different applications like writer identification and demographic~~s~~ (age, sex, place of inhabitant etc.) classification, word spotting and conversion of documents into machine understandable form are found in literature. Apart from these conventional applications, here, an effective and automatic mechanism, which may be used for managing handwritten documents, is introduced in this thesis. For this, a keyword based handwritten document classification method is proposed.

Any such system, in general, consists of a number of stages (refer to Fig. 1.8), that needed to be implemented to obtain a generalized system together with an objective of lessening the overall time requirement without compromising the overall performance of the system. The major stages that need to be studied are i) preprocessing of document images, ii) TL extraction and word extraction, iii) word recognition iv) keyword searching and document classification. In addition to these, preparation of database and development of automatic evaluation tool for assessment of any designed system are the other important requirements.

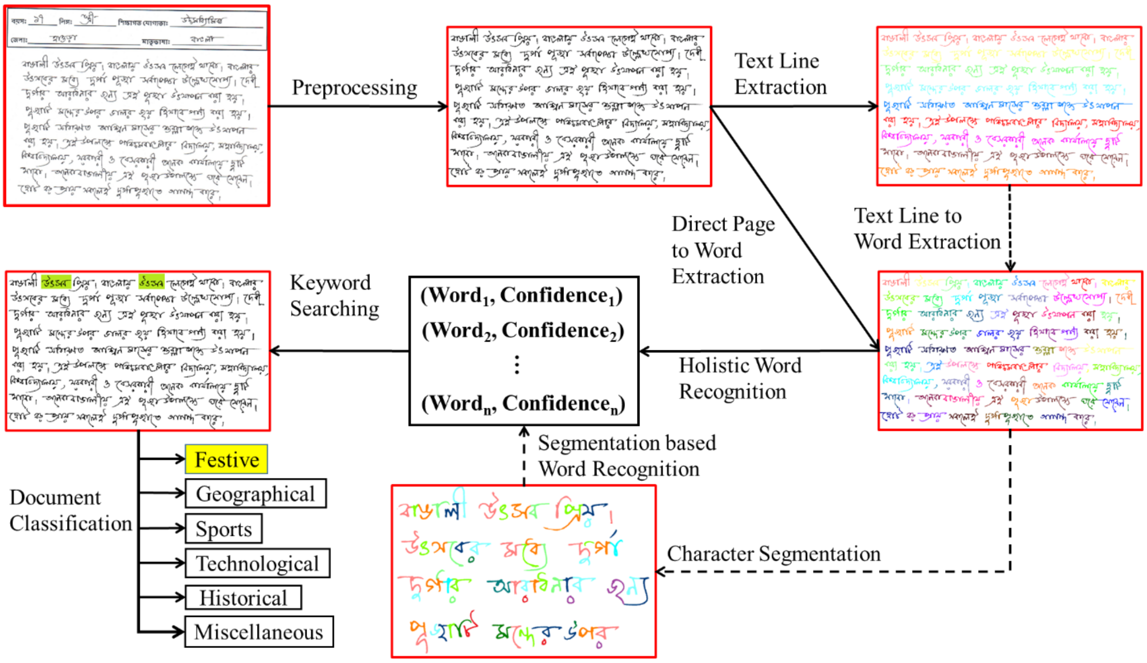

Preprocessing
Text Line Extraction
Direct Page to Word Extraction
Text Line to Word Extraction
Keyword Searching
(Word$_1$, Confidence$_1$)
(Word$_2$, Confidence$_2$)
⋮
(Word$_n$, Confidence$_n$)
Holistic Word Recognition
Segmentation based Word Recognition
Character Segmentation
Document Classification
Festive
Geographical
Sports
Technological
Historical
Miscellaneous

Fig. 1.8 Illustration of different stages of a handwritten document classification system which classifies the documents into one of the six categories viz., festive, geographical, sport, technological, historical and miscellaneous. Processes that are connected using solid arrow are the stages which are finally used here to build the said system whereas those are connected with dotted arrows are used in alternative approaches

A brief description of all the stages, which are shown in Fig. 1.8, is provided in the following subsections. Also, some notable challenges, which are mostly encountered while working with handwritten *Bangla* documents, for analysis of textual content in handwritten documents are included therein.

1.2.1 Preparation of Database

Preparation of database (i.e., choice of suitable dataset) for any pattern recognition and/or image processing experiments plays vital role. Any document processing research is almost impossible without adequate and proper collection of documents. In this context, researchers in literature, have used either in-house database or standard databases to conduct the required experiments.

It is to be noted that very few databases of handwritten documents in *Bangla* are available till date. For example 100 handwritten document pages are available in CMATERdb1.1.1 [24] and another 50 such samples are made available through International Conference on Document Analysis and Recognition (ICDAR) [25]. The first database could be used for TL extraction only while the second one can be used for TL extraction and word extraction purposes. These databases are not large enough for confirming robustness for a devised technique. No other database is found which may be used for researches like word spotting and document classification. Even, a database which could be useful for evaluating any character extraction algorithm is missing in literature. Last but not the least, database for handwritten *Bangla* word recognition in holistic way [26] is also absent. All these issues make analysis of textual content in handwritten documents a real challenge.

1.2.2 Preprocessing of Document Images

Offline documents like historical documents, manuscripts, records etc. may contain noise due to degradation. Document level skewness occurs commonly during manual scanning of the documents and also due to the writing style of the writer. Existence of noise and/or skew in document images affects the overall performance of any system used for analyzing their content. Also, working with binarized images is easier than gray-scale or color images. So, a number of effective preprocessing strategies are introduced by the researchers to make the document images ready for subsequent analysis. The preprocessing mechanism generally consists of filtering, binarization, noise removal and skew correction. To remove the unwanted noise present in the scanned documents, some spatial image filtering mechanisms, e.g., mean and median filtering [27], and Gaussian smoothing [28] are adopted in literature.

Since, working with binarized images is easier than that of gray-scale images as binarized images consists of two types of pixels, one representing the foreground or objects and other one representing backgrounds. Therefore, document images are frequently converted to binarized images. Conversion of a gray-scale image into a binarized image is mainly performed using two approaches *namely*, global thresholding and adaptive thresholding. Otsu's [29] method is an example of global thresholding approach while Niblack's [30] and Sauvola's [31] algorithms are representative of the other one.

The document images may be degraded due to aging of the document, poor paper quality, fading of ink etc. These types of degradation lead to the addition or loss of object pixels in binarized image. So to fill the loss of data pixels or to remove extra data pixels in the binarized image noise removal techniques are usually performed. For example, connected component (CC) based mechanism [32] and morphological operators like close [33], Run Length Smoothing Algorithm (RLSA) [34] are used by researchers for removing noise pixels.

During scanning improper placement of the document on the scanner bed produces document level skew. Also, writing style of individuals may add skew to TLs and words inside a document image. The document level skew will weigh down the overall performance of the system. To cope up with this problem several attempts have been made by the researchers using Hough Transform (HT) [35], Eigen vector line fitting [36], Radon Transform [37] etc. based approaches for skew correction. Apart from these, sometimes, suppression of redundant information (e.g., printed text [38], symbols [39] and graphics [40]) from document images, layout analysis [41] etc. are also performed prior to actual textual content analysis

1.2.3 Text Line Extraction

TL is considered as the fundamental unit during TL based content analysis in handwritten documents. The researchers, in literature, have extracted TLs from document images and then used these extracted TLs for converting them into machine-encoded form [42-43] or searching a query image [44-45]. Also, conventional word extraction algorithms have employed TL extraction prior to word extraction [46-47].

Single and/or multi orientated TLs, overlapping and/or touching TLs and non-uniform inter-TL spacing commonly found in handwritten document images. Single oriented TLs may appear in document images due to page level skewness or writers' upward/downward writing style. But the multi-oriented TLs in a handwriting document are found due to skewed writing style of writers. Even, skewed TLs may also be intentionally inserted into documents due to some special purposes which are found in artworks and advertisements. Also, overlapping and/or touching TLs are very common in handwritten document images. Such TLs are generated for the reasons like intrinsic writing style of writers, preparation of document in a hurry and presence of ascendants and descendants in *Bangla* words. All these issues make the problem of TL extraction from handwritten documents a challenging one.

To resolve these issues the researchers have applied different CC based [48], morphology based [49] and partition based approaches [50]. The work reported in [50] presents a novel technique for segmentation of multi-oriented handwritten TLs using water flow method. It has used hypothetical horizontal water flow from both sides of the document image. This method is good for extracting TLs from handwritten documents provided the TLs do not contain touching/overlapping components and large line-level skew. To handle this problem, in [48], the authors have applied a CC based approach. Once again this work performs poorly when document contains touching TLs and/or irregular inter-TL spacing since this method threads the components along a line based on distance between their centroids. Also, a document page may contain a large number of CCs which in turn increases the overall processing time. The work described in [51] has used piece-wise water-flow technique for the said issue. Here, the traditional water-flow technique has been employed in each of the vertical fragments prior to the final detection of TLs present in the underlying document image. Finally, distance metric is used to join all these generated line segments (LS) in each of the fragments with those in adjacent fragments. It can handle multi-oriented TLs but fails for touching and closely overlapped TLs.

1.2.4 Word Extraction

Word is also used as a basic unit while working on applications like word recognition [52], keyword spotting [44], character extraction [46] and word level script identification [53-54].Here arises the need for word extraction from document images. Two basic approaches, based on the nature of input (i.e., TL or document page), for word extraction are found in literature. In the first approach, word extraction follows TL extraction [49] or already segmented TL has been provided to the word extraction module [46, 49]. The other approach may be carried out by localizing the words in the document image directly without extracting the TLs present therein [55], which is here termed as page-to-word extraction technique.

Algorithms, those follow the first approach, are mostly gap metric based approach or morphology based approach. The algorithms that follow gap metric based approach consider that the gap between consecutive words is longer than the gap between two consecutive characters in a word. Hence, a predefined threshold value is set to detect the words in a TL.

One such work could be found in [56]. But, due to non-uniform distribution of inter-word and intra-word spaces in handwritten texts, the task of word extraction using gap metric may fail. On the other hand, the work, described in [57], tries to detect the words that present in a TL by joining CCs that are spatially close to each other. Spatial closeness is determined by spiral run length smearing algorithm (SRLSA). This method satisfactorily handles non-uniform inter-word and intra-word gap but fails when it faces touching and/or a word/character overlapping characters within the successive words in a TL or overlapping (sometimes touching) with punctuation mark.

It is worth mentioning that one can avoid TL extraction prior to word extraction technique if the underlying application has less impact on ordering of words in a document. For example, applications like medicine name searching in a prescription or medical report [58], suppression of printed word(s) from document containing both handwritten and printed words [38] and word level script identification from mixed script document page [59], pay less attention on ordering of words in a TL. Also, documents such as doctor’s prescription and artwork may not contain the concrete presence of TLs. In these cases, applying TL extraction technique prior to word extraction not only increases the overall execution time but also includes the errors, which occur during TL extraction phase. Here lies the need for extracting words directly from document page images and the technique by which it can be possible is termed as page-to-word extraction technique. Unfortunately, no such significant work is found in literature to handle this issue while considering handwritten *Bangla* document images.

1.2.5 Character Extraction

Irrespective of script/ language, the fundamental units of a word are characters. A finite and pre-defined set of symbols is used to represent the character set. The number of meaningful words for any language, in general, is much higher than the number characters it possess. Hence, a general trend which has been followed in literature is to segment a word image into constituent character or character like shapes, sometimes called as character segmentation, prior to word recognition.

Usually, the problem of character extraction is script dependent. Due to the presence of *Matra*, it is obvious that character extraction methods for *Bangla* script become different from *Matra* less scripts like Roman, Tamil, Telegu and Odia. But the presence of non-uniform and wavy *Matra* adds challenge to the problem of handwritten *Bangla* word segmentation into characters of character like shapes. Apart from this, skewness, elongated part of middle zone characters, presence of part of a character in *Matra* region and touching characters and/or modified shapes beyond *Matra* region make the issue more critical.

From technical point of view, research attempts on character extraction from handwritten *Bangla* words can be categorized as (i) Word based approach which considers the entire word as input to extract character present therein [60], (ii) CC based approach which considers only the CC(s), containing multiple characters, in a word image to perform word extraction [61] and (iii) recognition based approach which detects the characters, present in a word image, based on underlying classifier's feedback through implicit segmentation [62]. The techniques that follow second or third approach are suitable for character extraction but they need huge samples, collected in offline mode, for training the classifier under consideration. Therefore, these approaches require huge manual effort. Not only this, these techniques are heavily dependent on classifier performance and the nature of collected training samples and they are also very time consuming. In this context, it is noteworthy that researchers who follow the said techniques have evaluated their algorithms on in-house database and have recorded performances using manual evaluation.

1.2.6 Word Recognition

Irrespective of models, discussed earlier, word recognition is the core part of any content analysis system for handwritten documents. As per the work, described in [63], the recognition of the words, in general, is done using two different approaches such as a) Analytic and b) Holistic. The analytic approach, in general, is preferred when the size of lexicon is too large and/or the set of words for the domain where the technique is applied is not predefined. In this approach, the words are segmented into constituent character(s) or sub-character(s) and then recognition of the individual components is carried out for formation of actual machine code of the characters [15]. Hence, performance of any word recognition system that follows this approach is mostly affected by character extraction ambiguities from handwritten word images [63]. For instance while segmenting a word it may generate less/more characters than expected. Sometimes some characters may be found in word images as inherently segmented. So deciding the number of patterns becomes a critical job for the researchers.

The holistic approach tries to recognize the whole word directly without applying a character extraction process i.e., it accepts a word image as input and recognizes it. Which means that the techniques following this approach can avoid problem related to character extraction, which becomes more critical while considering Bangla handwritten words. However, it is worthy to mention that if the number of words to be classified is predefined and small then this approach is effective since the success of the holistic approach depends on the size of the lexicon. Due to these advantages, this approach has been used successfully in document retrieval systems like keyword based document searching [44], grouping of documents based on some document-type specifying words like “application”, “cancellation” and “urgent” [64] and detection of table and/or figure caption [65]. In spite of its adequate real life applications, holistic word recognition is still almost untouched for recognition of handwritten *Bangla* words.

1.2.7 Keyword based Document Classification

As said earlier that the volume of existing handwritten documents is already large and it is ever increasing. Hence, storing and managing of handwritten documents pages, at least, in image format is required since they contain important information. For quick retrieval of an important document from a pool document images, they should be categorized as per individual user choice before storing into database. This implies that documents should be stored/ arranged/ organized according to their category. For example, all the articles of a particular type or written by a single author are generally kept together in any library system.

Textual content (here, text with ASCII representation) based document classification is a well-studied research problem due to its omnipresent applications in daily lives. It, mainly, deals with two major issues *namely* selection of appropriate set of keywords for classification of documents and handling of large scale data coming from web contents like emails, science journals, e-books, learning materials, news and social media [66]. Hence OCRing of the handwritten documents is the most acceptable solution since doing so content can be classified like it is done for machine understandable texts like emails, blogs and messages.

Although handwriting recognition has started its journey about five decades ago, still no such methodology has been found in the literature which provides an integrated approach to the classification of handwritten document images. In this scenario, keyword-based handwritten document classification may provide an effective solution to classify the handwritten document images. The essential requirement for this is a keyword searching technique, which is also known as word spotting method in literature [44].

Works on word (keyword) searching, described in the literature, are found to differ on implementation aspects like (i) how a query word is fed to a word searching system, (ii) whether a keyword is searched from whole page without applying page segmentation or from segmented components like TL and word and (iii) how a target word is matched with a query word. Depending on how a query keyword is fed to the system, keyword spotting technique are classified into two categories *viz.*, query by string (QBS) [65] and query by example (QBE) [67]. In the QBE protocol, an image of the query word is used, whereas in the QBS paradigm, an arbitrary text string is placed as input to the system.

Based on the matching schema, i.e., how a keyword is matched with a target word, keyword spotting techniques are classified as either recognition-free [68]or recognition based [69] methods. In the former category, in general, some distance based measures are used without recognizing the word images. These methods, mostly, achieve better result while working with machine printed document page images. However, considering the unconstrained nature of handwriting, possibility of failure of these methods is very high. But, the other type of approaches includes some recognition technique and thereby it is supposed to perform better than its counterpart for handwritten document. But, the notable drawback of this approach is the need of ample number of training samples and predefined lexicon.

Along with these, based on the searchable region, keyword spotting techniques are classified into two categories, *namely*, segmentation-free approach [70] and segmentation based approach [71]. In the first category of approaches no page segmentation technique, like TL and word extraction, is used for spotting a word inside a document page. These methods are good choice while considering degraded documents or having very few example images to work with. On other hand, segmentation based approach segments a document page either into TLs or words and then perform searching. Methods that take TLs or word as input for searching keywords from document image have mostly used Hidden Markov Model (HMM) which uses character and/or language model [72], and needs sufficient number labelled data that represent underlying character or language model.

In this regard, it is worth mentioning that keyword searching mechanism is almost invisible in literature while considering handwritten *Bangla* documents. Even, no significant work, which has classified / indexed handwritten documents based on the textual contents, is found in literature of analysis of handwritten documents. It may be due to inherent problems associated with every stage of content analysis system of handwritten *Bangla* document images and absence of suitable data for performing such research.

1.3 Scope of the Thesis

The work presented in this thesis deals with the problems related to analysis of textual content in handwritten documents in *Bangla* script. The major objective of this work is to devise an automatic system that can search a user provided keyword in handwritten Bangla documents and also can classify the documents into personalized categories on the basis of the set of keywords supplied. To search a keyword from a handwritten document, all the words present in the document image are extracted first and then a two-stage word matching technique is employed. It has already been mentioned that words from a handwritten document can be extracted following two major approaches *namely*, word extraction that follows TL extraction [47] and page-to-word extraction [55]. Hence, both the approaches are studied here and the former one is chosen due to its advantages.

It is to be noted that during the two-stage keyword searching technique, count of ascendants, descendants and middle zone characters, which can be estimated using a suitable character extraction technique, have been used as features in the first stage while holistic word recognition technique is employed in later stage. Therefore, a character extraction technique and holistic word recognition are also developed during this thesis work. Also, some document image preprocessing methods such as filtering, binarization, noisy components removal and page level skew correction are introduced. Apart from these, several sets of suitable image data are prepared and evaluation tools for automatic assessment of TL, word and character extraction techniques are developed.

The databases, developed under this work, are of two categories *viz.*, database of handwritten document page images and databases of isolated handwritten word images. The first database consists of a total of 300 document pages with contents belonging to the six different predefined categories *viz.*, festival, geographical, sport, technological, historical and miscellaneous. For each category of contents, 50 handwritten document pages from different writers are collected. The isolated word image databases are prepared for three different purposes *viz.*, assessing character extraction model (5000 word images), evaluating holistic word recognition technique (18000 word images: 150 word images for each of the 120 popular city names of West Bengal, India) and performing recognition based keyword searching (3000 keyword samples i.e., 150 image samples for each of the 20 keywords, used to search from documents). The city name database [73] is already made freely available to the research community. Detail descriptions of the databases containing document page images and the said categories of word images are described in Chapters 2, 5, 6 and 7 respectively.

The databases developed under this thesis work for experimenting TL, word and character extraction techniques are accompanied by separate databases containing the corresponding GT images. To prepare the GT images the corresponding TL/word/character extraction technique is applied first and then the erroneously extracted components (i.e., TL, word and character) are manually corrected either using GTgen software [74] or MS paint software. Detail descriptions of TL, word and character level GT image preparation techniques are provided in Chapters 3, 4 and 5 respectively. Also, an automatic evaluation protocol (here one-to-one pixel mapping is considered), which is the backbone of all the designed evaluation tools, is developed during the course of thesis work. It uses one-to-one pixel mapping between segmented image and corresponding GT image and returns evaluation results in terms of true positive (TP), false negative (FN) and false plosive (FP), recall, precision and F-measure [75]. The detail description of this evaluation technique is described in Chapter 3.

Handwritten samples (i.e., document page images or isolated word images) collected here may contain noisy components or suffer from page level skewness and that is why here a set of image preprocessing techniques is adopted. First, the document page images are filtered using middle of modal class (MMC) based filtering technique [76]. Next, the MMC filtered handwritten document page images are passed through a global thresholding approach based binarization technique, termed as Ratio based binarization [21, 32]. Such generated binarized image may contain either salt and pepper noise or small cluster of background/object pixels which are then reduced using morphological close operator and a CC based approach [21]. To handle page level skewness, a Hough Transform (HT) based technique [77] is used. All these document level preprocessing methods are described in Chapter 2.

A TL extraction technique is developed by hybridizing the contour based approach [78] and SRLSA based approach [57]. Here, a handwritten document page image is first partitioned vertically into a number of fragments of equal width and then the object pixels in all these fragments are smeared using existing SRLSA [79]. Next, line segments (LSs) in each vertical segment is estimated by identifying upper and lower contours, as described in [78]. After that, intra-fragments LSs having lesser height than average height of all LSs are joined with the closest LSs in terms of vertical distance. At end, threading of LSs in the neighboring fragment is performed. The detail description of the TL extraction technique along with associated experimental results are described in Chapter 3.

As already mentioned, for extracting words from handwritten document page images two modules have been investigated here. The modules are (i) word extraction from TL and (ii) page-to-word extraction. In the first module, a SRLSA based word extraction technique, described in the work [46], is used. It is worth mentioning that the input TLs for this system have been extracted from handwritten documents using the current TL extraction algorithm. This means errors occur during the TL extraction technique have also been carried forward into this word extraction technique.

Whereas in the second module, a CC based page-to-word extraction technique is followed. Such attempt has been taken mainly to bypass the errors, which occur during application of TL extraction method and also to minimize the overall execution time. This technique follows CC based approach which is a bottom-up approach used for page segmentation [80]. In this technique, first all the CCs present in a document page are extracted and the CCs are classified as small sized or large sized based on their heights and widths. Next, a small sized CC gets joined with the closest CC among its 8-neighbors. Afterward, joining of horizontally close CCs is carried out to obtain the final set of word images. The experimental results along with detail description of these two word extraction techniques are provided in Chapter 4.

During extraction of the characters or character like shapes from a handwritten *Bangla* word image, segmentation at *Matra* region and separation of lower zone are performed. Before doing the actual segmentation, the zonal boundaries i.e., R1, R2, R3 and R4 of word image are estimated. R1 and R4 are the first row from top and bottom respectively having at least one object pixel. To estimate R2 and R3, a mask based approach is adopted here where the width of the mask is same as width of word image and height of the mask varies based on length of the runs of background and object pixels in vertical direction. Masks that contain R2 and R3 are selected by analyzing the runs of object pixels and count of row-wise transition points (object to background and vice versa) in horizontal direction.

For detection of *Matra* pixels, a trapezoidal *fuzzy membership function* [81] and horizontalness feature [82] are used here. Then, to decide some of the *Matra* pixels as segmentation points, a bell-shaped *fuzzy membership function* [82] is used. In this stage, two features *viz.*, distance of farthest object pixel from R2 and number of object pixels for each column are considered. However, detection of segmentation points leave out some object pixels which eventually cause under segmentation. Also, converting the segmentation points into background pixels leads to loss of object pixels [83]. Hence, a concept of creating segmentation lines is incorporated here which again generates some additional segmentation line(s) that causes over segmentation. Therefore, another mechanism, which performs rejection or merging segmentation lines, if required, is applied. Finally, a technique to separate out the lower zone modifier(s) is devised. The detail description of the method and experimental results are provided in Chapter 5.

For recognition of handwritten *Bangla* word images in holistic way, three types of features *namely*, elliptical [84], gradient based [85] and topological [86] are extracted from word images. In general, width of a word image is larger than its height; hence, a word image can be enclosed within an elliptical shape rather than a circular shape. Considering this very fact, here a set of elliptical features is extracted from the word images. Histogram of Oriented Gradients (HOG) feature descriptor [87] motivates in designing the gradient based features. Also, to get an idea about geometrical characteristic, a set of six features *namely*, area, aspect ratio, pixel density, longest run length, centroid and projection length is extracted from a word image. All these features are extracted locally and globally from each of the word images.

These extracted features may contain irrelevant and/or noisy features. Hence, inspired by the work described in [88] which has been introduced during the course of this thesis work, all these extracted features are fed to a hierarchical feature selection (HFS) technique [88] to get optimized feature vector. In this HFS model, first these feature vectors are optimized separately and then all the optimized features are combined and optimized further. Memetic Algorithm (MA) [89] is used for feature selection. Finally, with this optimized feature vector the word images are classified using an MLP. The detail description along with experimental results are described in Chapter 6.

For searching a keyword from a handwritten document image, first the document image is segmented into word images using a page-to-word extraction method. Next, punctuation marks and irrelevant words with respect to a given query keyword are suppressed using a set of decision rules which are formulated using a method described in the work [38]. Four zonal features *viz.*, mean of row-wise transition points in the middle zone and the numbers of middle zone characters, ascendants and decedents are used for this purpose. The remaining target words in the document page are probable candidate keywords. Finally, the keyword under consideration is searched from these candidate keywords which is performed using the presently devised holistic word recognition technique.

Also, based on presence of personalized set of keywords, supplied by user for each of the document categories, the entire document page images are classified into respective categories. For classification, an MLP based classifier and two features, called *logarithmic term frequency* and *grouped term frequency,* are used. Such choice of features helps in diminishing the effect of erroneous retrieval of keywords. The detail description of keyword searching technique and the designed document classification method along with the results are provided in Chapter 7.

The thesis is finally concluded in Chapter 8 where the observations of the present work in summarized.

The work presented in this thesis is a result of consistent efforts to cover almost all the major stages to build an automated system that can classify the textual contents in handwritten *Bangla* documents. Firstly, such attempt resulting into an integrated approach towards classifying handwritten *Bangla* document is unique. Secondly, the improved intermediate stages might help in constructing an OCR system for handwritten *Bangla* document images in future. Thirdly, the results obtained through application of intermediate stages of the present study on freely written *Bangla* texts can help identifying the areas where the researchers need to pay more attention. Fourthly, the devised techniques could be useful for the documents written in other *Matra* based scripts like Gurumukhi, Devanagari, Sylloti. Finally, the databases developed under this work can be of great help for testing, evaluating and comparing performances of algorithms for TL, word and character extractions, holistic word recognition, keyword spotting and document classification for handwritten *Bangla* script.

Chapter 2

HANDWRITTEN *BANGLA* DOCUMENT PAGE IMAGE DATABASE PREPARATION

2.1 Introduction

Database plays a key role for research related to analysis of textual contents in handwritten document images. In general, different stages of such analysis, described in the previous chapter, vastly rely on certain database(s). In fact, it is not be possible to assess any algorithm without proper data. This issue becomes more prominent for researches on analysis of textual contents in handwritten documents due to lack of appropriate data. The reason behind such lack of data might be due to amount of manual effort required to prepare one such database. Also, the requirements like high preparation time, man-power, incorporating writing variations in documents and fulfilling some standard are the other constraints. Some standard databases on handwritten samples found in literature are described in brief below.

CMATERdb: The Pattern Recognition Database Repository [90] is an open access (for non-commercial use only) database used for different pattern recognition problems. It is maintained by "Center for Microprocessor Applications for Training Education and Research" (CMATER) research laboratory, Jadavpur University, Kolkata, India. It presently contains databases of multi-script handwritten document pages, isolated characters, digits and words of city names etc. It provides the ground truth (GT) images related to script identification and text line (TL) extraction purposes.

ISI Handwritten Character Databases [91] contain isolated character and numeral images for both online and offline modes. The data contain handwritten isolated characters for two Indic scripts (*Bangla* and Devanagari) and numeral of three scripts (*Bangla*, Devanagari and Odiya). Few samples of the said database are open for viewing while entire dataset is available on the basis of request. It is under the supervision of Indian Statistical Institute, Kolkata, India.

IAM Handwriting Database [92] contains form documents of English text written in Roman script which are primarily used for writer identification and verification. It has also been used in several other research works like text line and word extraction, slant and skew correction, sentence and word recognition. IAM database is prepared by Research Group on Computer Vision and Artificial Intelligence, Institute of Computer Science, University of Bern, Switzerland.

NIST Database [93] is another well-known database of Roman script. It consists of images of hand printed forms i.e., document pages containing both printed and handwritten texts. These forms are primarily intended for the research work in form processing [94]. Apart from this, the database can also be used for recognition of characters and digits. It is available on the web through the name EMNIST [95]. On the other hand, MNIST [96] provides a database of handwritten digits. Both databases are collection of binarized images which means the gray-scale pixel intensities are absent.

QUWI Database [97], a large-scale page level handwritten document image database, prepared for writer identification and recognition research. This dataset contains handwritten Arabic and English texts of same content. The original dataset contains 4068 number of document pages (equal number of Arabic and English documents) written by 1017 different persons (2 samples per person per script). However, a portion of the entire database is made freely available for research purpose through different competitions or on request basis.

International Conference on Document Analysis and Recognition (ICDAR) [98] and International Conference on Frontiers of Handwriting Recognition (ICFHR) [99] are two leading conferences on document analysis. These conferences organize different international level competitions related to automatic analysis of handwriting. The databases that are used in those competitions are made freely available (sometimes along with their GT information) by the organizers to the research community for further advancement of said research problems. In general, these databases are prepared in such a way that they can be used for the specified problem only.

Apart from the above, the works [100-102] present some database for handwriting analysis. The mentioned databases reveal the fact that they are mostly prepared and benchmarked to sort out one or two specific issues of handwriting content analysis research. Also, it is worth mentioning that a multi-purpose database along with their GT information is important. Therefore, in this thesis, a database containing 300 handwritten document page images is prepared.

2.1.1 Objective of the Chapter

The objectives that covered in this chapter are as follows:

- Collection of new handwritten samples to meet the requirements stated above.
- Design of a method to extract required handwritten textual content parts from pre-formatted forms and then store them as binarized and gray-scale images. These images will be called as document pages in the following chapters.

2.2 Data Collection

Data collection is the basic step of database preparation in any data oriented research work. Therefore, in this work, handwritten samples in *Bangla* script with different textual contents are collected from different individuals. For collection of writing samples from different writers and also to store their basic information like age, sex, educational qualification, native place (district name here) and mother tongue, a form is prepared following the style used for preparing IAM database. A sample form is shown in Fig. 2.1. These forms are either printed using HP LaserJet printer or copied using standard photocopier. Therefore, the quality of the hard-copy forms varies and sometime addition of noise pixels therein is also found.

Here, five different textual contents, shown in Fig. 2.2 (a-e), are prepared out of which four contain digit strings. The length of these textual contents are so prepared that people can write them freely within the specified space in the pre-formatted form (see Fig. 2.1). Then, different volunteer writers, varying in age, sex, educational qualification and place of inhabitance, are requested to provide their handwriting samples. According to the works [6, 103-104], the choice of such varied groups of writers incorporates varieties in writing styles. The authors have written a content on single pre-formatted datasheet using black/blue ink gel pen. The maximum number of different contents written by an individual writer is two while no writer has contributed twice for a single content. Two examples of such collected form is shown in Fig. 2.3 (a-b).

Total 250 different writing samples are collected from 200 different writers. The distributions of writers' age, sex and educational qualification are shown in Fig. 2.4 (a-c). The writers are mostly inhabitant of the districts of Nadia, Purulia, Bankura, Jalpaiguri, Murshidabad, Howrah and Kolktata of West Bengal, India. These collected forms are grouped into 5 different sets, named as Set-A to Set-E, based on their content. Each of the sets contains exactly 50 samples. The database name and their categorical information along with basic information, such as number of words (character string), digit strings and symbols (except space count) are shown in Table 2.1.

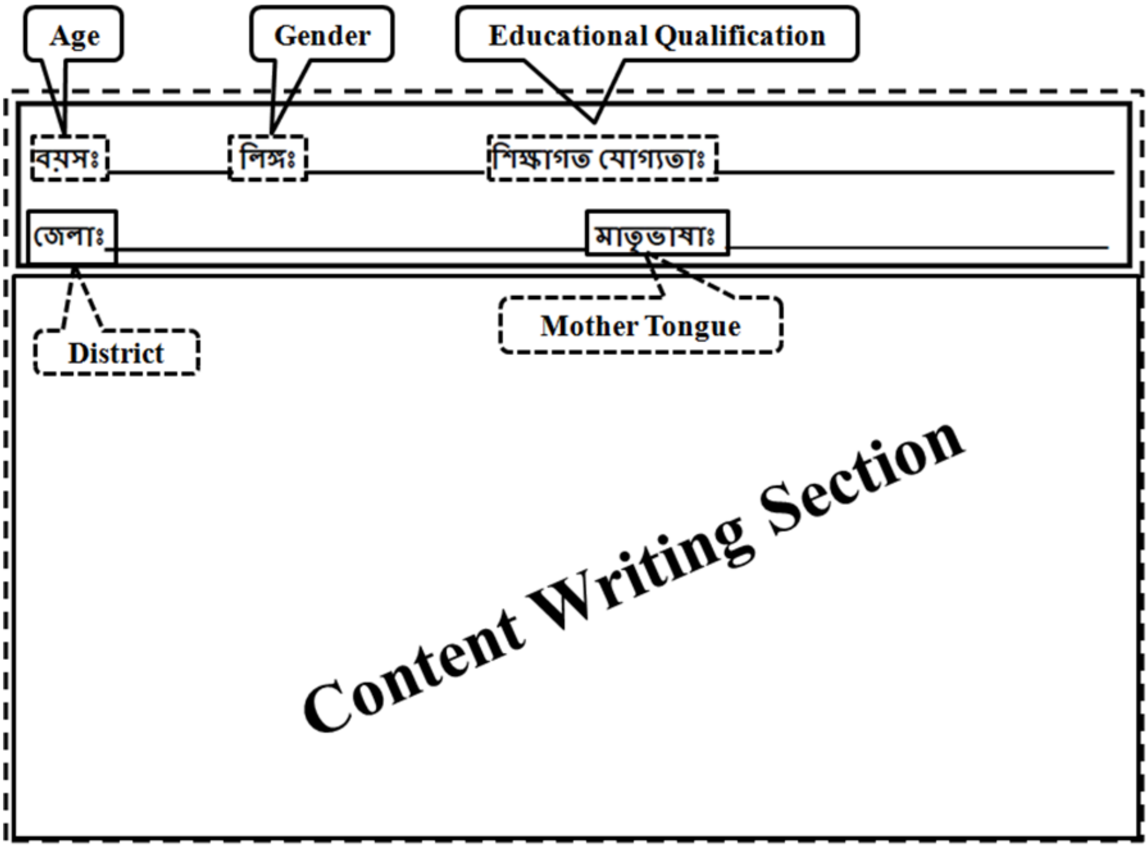
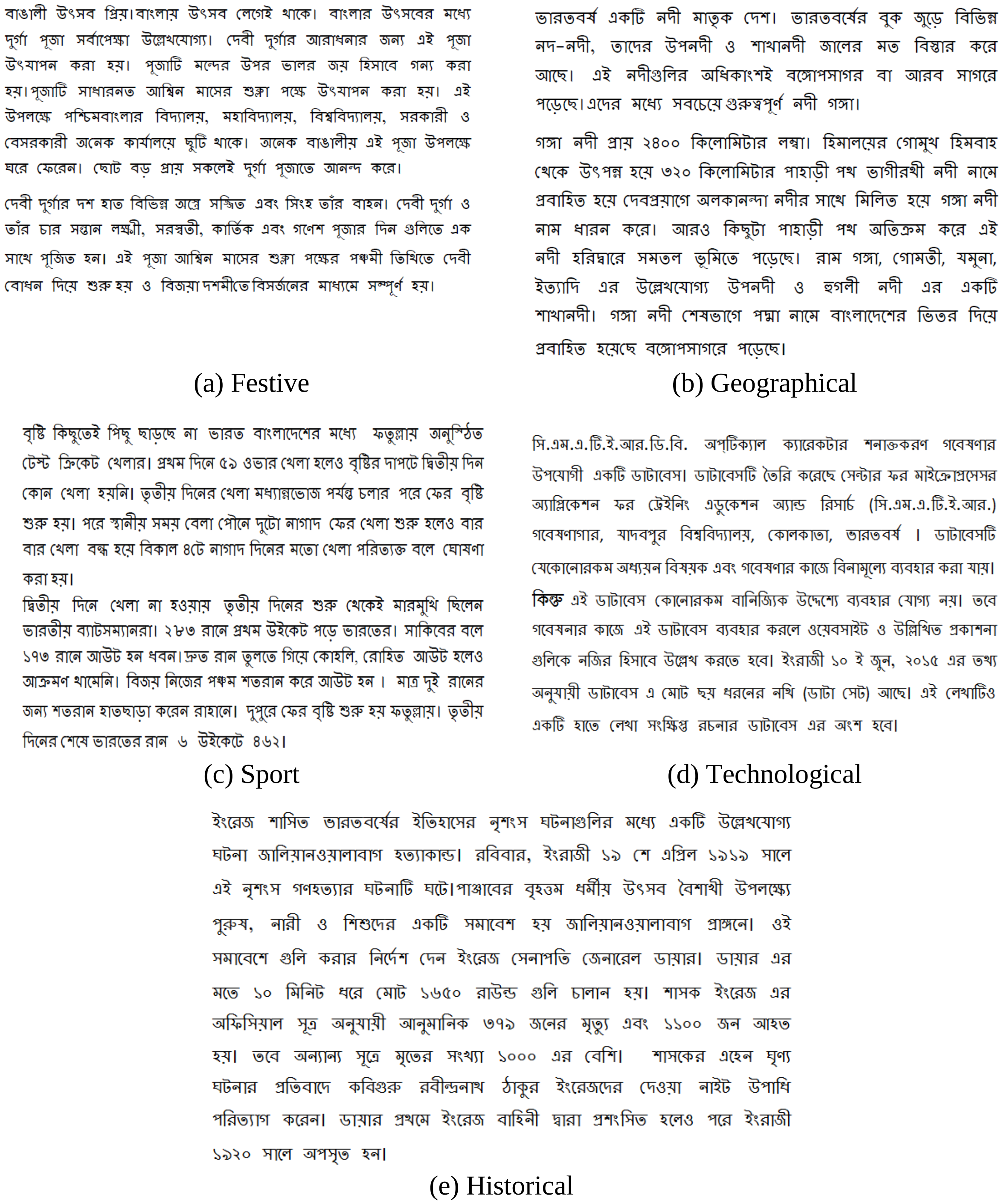


Fig. 2.1 Instance of form which is used for collecting handwriting samples

Along with these aforesaid databases, 50 handwritten document pages from CMATERdb1.1.1 [24] have also been included in the final database. These non-categorized documents are included in the database to incorporate more variations in terms of writers and contents. It would be helpful to study the robustness of different works, described in later chapters of this thesis. This included part is treated as a new set and is named as Set-F. A sample page from Set-F is shown in Fig. 2.5. Therefore, in total, a database containing 300 pages of handwriting samples is considered in the present work and it is divided into 6 sets based on the contents.

All these collected handwritten samples are then scanned using flat-bedded scanner at 300 dots per inch (dpi) in 24-bit color map and saved in *bmp* format. These digitized document pages are converted into gray-scale image by applying luminance formula [105]. Let, $P_{rgb} = \{f(x,y,z): (x,y,z) \in [1,H_P] \times [1,W_P] \times [1,3] \wedge f(x,y,z) \in \{0,1,2,\dots,255\}\}$ and $P_g = \{g(x,y): (x,y) \in [1,H_P] \times [1,W_P] \wedge g(x,y) \in \{0,1,2,\dots,255\}\}$ represent a 24-bit color image and gray-scale image respectively. Here W_P and H_P are the width and height of a page. The gray-scale conversion of page can be defined by eq. (2.1).

$$g(x,y) = f(x,y,1) \times 0.299 + f(x,y,2) \times 0.587 + f(x,y,3) \times 0.114 \tag{2.1}$$



বাঙালী উৎসব প্রিয়।বাংলায় উৎসব লেগেই থাকে। বাংলার উৎসবের মধ্যে দুর্গা পূজা সর্বাপেক্ষা উল্লেখযোগ্য। দেবী দুর্গার আরাধনার জন্য এই পূজা উৎযাপন করা হয়। পূজাটি মন্দের উপর ভালর জয় হিসাবে গন্য করা হয়।পূজাটি সাধারনত আশ্বিন মাসের শুক্লা পক্ষে উৎযাপন করা হয়। এই উপলক্ষে পশ্চিমবাংলার বিদ্যালয়, মহাবিদ্যালয়, বিশ্ববিদ্যালয়, সরকারী ও বেসরকারী অনেক কার্যালয়ে ছুটি থাকে। অনেক বাঙালীয় এই পূজা উপলক্ষে ঘরে ফেরেন। ছোট বড় প্রায় সকলেই দুর্গা পূজাতে আনন্দ করে।

দেবী দুর্গার দশ হাত বিভিন্ন অস্ত্রে সজ্জিত এবং সিংহ তাঁর বাহন। দেবী দুর্গা ও তাঁর চার সন্তান লক্ষ্মী, সরস্বতী, কার্তিক এবং গণেশ পূজার দিন গুলিতে এক সাথে পূজিত হন। এই পূজা আশ্বিন মাসের শুক্লা পক্ষের পঞ্চমী তিথিতে দেবী বোধন দিয়ে শুরু হয় ও বিজয়া দশমীতে বিসর্জনের মাধ্যমে সম্পূর্ণ হয়।

(a) Festive

ভারতবর্ষ একটি নদী মাতৃক দেশ। ভারতবর্ষের বুক জুড়ে বিভিন্ন নদ–নদী, তাদের উপনদী ও শাখানদী জালের মত বিস্তার করে আছে। এই নদীগুলির অধিকাংশই বঙ্গোপসাগর বা আরব সাগরে পড়েছে।এদের মধ্যে সবচেয়ে গুরুত্বপূর্ণ নদী গঙ্গা।

গঙ্গা নদী প্রায় ২৪০০ কিলোমিটার লম্বা। হিমালয়ের গোমুখ হিমবাহ থেকে উৎপন্ন হয়ে ৩২০ কিলোমিটার পাহাড়ী পথ ভাগীরথী নদী নামে প্রবাহিত হয়ে দেবপ্রয়াগে অলকানন্দা নদীর সাথে মিলিত হয়ে গঙ্গা নদী নাম ধারন করে। আরও কিছুটা পাহাড়ী পথ অতিক্রম করে এই নদী হরিদ্বারে সমতল ভূমিতে পড়েছে। রাম গঙ্গা, গোমতী, যমুনা, ইত্যাদি এর উল্লেখযোগ্য উপনদী ও হুগলী নদী এর একটি শাখানদী। গঙ্গা নদী শেষভাগে পদ্মা নামে বাংলাদেশের ভিতর দিয়ে প্রবাহিত হয়েছে বঙ্গোপসাগরে পড়েছে।

(b) Geographical

বৃষ্টি কিছুতেই পিছু ছাড়ছে না ভারত বাংলাদেশের মধ্যে ফতুল্লায় অনুস্ঠিত টেস্ট ক্রিকেট খেলার। প্রথম দিনে ৫৯ ওভার খেলা হলেও বৃষ্টির দাপটে দ্বিতীয় দিন কোন খেলা হয়নি। তৃতীয় দিনের খেলা মধ্যান্নভোজ পর্যন্ত চলার পরে ফের বৃষ্টি শুরু হয়। পরে স্থানীয় সময় বেলা পৌনে দুটো নাগাদ ফের খেলা শুরু হলেও বার বার খেলা বন্ধ হয়ে বিকাল ৪টে নাগাদ দিনের মতো খেলা পরিত্যক্ত বলে ঘোষণা করা হয়।

দ্বিতীয় দিনে খেলা না হওয়ায় তৃতীয় দিনের শুরু থেকেই মারমুখি ছিলেন ভারতীয় ব্যাটসম্যানরা। ২৮৩ রানে প্রথম উইকেট পড়ে ভারতের। সাকিবের বলে ১৭৩ রানে আউট হন ধবন।দ্রুত রান তুলতে গিয়ে কোহলি, রোহিত আউট হলেও আক্রমণ থামেনি। বিজয় নিজের পঞ্চম শতরান করে আউট হন । মাত্র দুই রানের জন্য শতরান হাতছাড়া করেন রাহানে। দুপুরে ফের বৃষ্টি শুরু হয় ফতুল্লায়। তৃতীয় দিনের শেষে ভারতের রান ৬ উইকেটে ৪৬২।

(c) Sport

সি.এম.এ.টি.ই.আর.ডি.বি. অপ্টিক্যাল ক্যারেকটার শনাক্তকরণ গবেষণার উপযোগী একটি ডাটাবেস। ডাটাবেসটি তৈরি করেছে সেন্টার ফর মাইক্রোপ্রসেসর অ্যাপ্লিকেশন ফর ট্রেইনিং এডুকেশন অ্যান্ড রিসার্চ (সি.এম.এ.টি.ই.আর.) গবেষণাগার, যাদবপুর বিশ্ববিদ্যালয়, কোলকাতা, ভারতবর্ষ । ডাটাবেসটি যেকোনোরকম অধ্যয়ন বিষয়ক এবং গবেষণার কাজে বিনামূল্যে ব্যবহার করা যায়। **কিন্তু** এই ডাটাবেস কোনোরকম বানিজ্যিক উদ্দেশ্যে ব্যবহার যোগ্য নয়। তবে গবেষনার কাজে এই ডাটাবেস ব্যবহার করলে ওয়েবসাইট ও উল্লিখিত প্রকাশনা গুলিকে নজির হিসাবে উল্লেখ করতে হবে। ইংরাজী ১০ ই জুন, ২০১৫ এর তথ্য অনুযায়ী ডাটাবেস এ মোট ছয় ধরনের নথি (ডাটা সেট) আছে। এই লেখাটিও একটি হাতে লেখা সংক্ষিপ্ত রচনার ডাটাবেস এর অংশ হবে।

(d) Technological

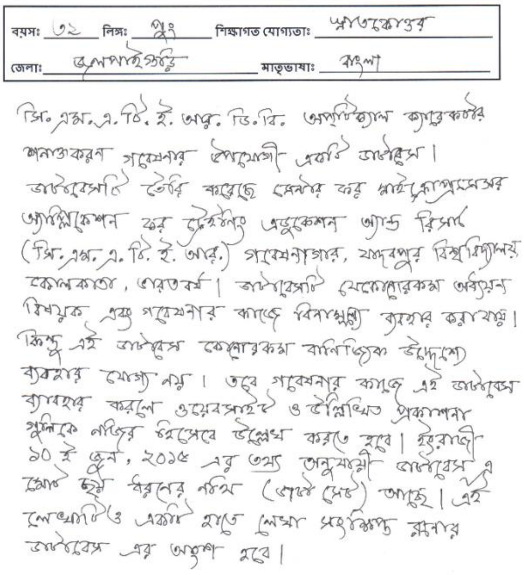
ইংরেজ শাসিত ভারতবর্ষের ইতিহাসের নৃশংস ঘটনাগুলির মধ্যে একটি উল্লেখযোগ্য ঘটনা জালিয়ানওয়ালাবাগ হত্যাকান্ড। রবিবার, ইংরাজী ১৯ শে এপ্রিল ১৯১৯ সালে এই নৃশংস গণহত্যার ঘটনাটি ঘটে।পাঞ্জাবের বৃহত্তম ধর্মীয় উৎসব বৈশাখী উপলক্ষ্যে পুরুষ, নারী ও শিশুদের একটি সমাবেশ হয় জালিয়ানওয়ালাবাগ প্রাঙ্গনে। ওই সমাবেশে গুলি করার নির্দেশ দেন ইংরেজ সেনাপতি জেনারেল ডায়ার। ডায়ার এর মতে ১০ মিনিট ধরে মোট ১৬৫০ রাউন্ড গুলি চালান হয়। শাসক ইংরেজ এর অফিসিয়াল সূত্র অনুযায়ী আনুমানিক ৩৭৯ জনের মৃত্যু এবং ১১০০ জন আহত হয়। তবে অন্যান্য সূত্রে মৃতের সংখ্যা ১০০০ এর বেশি। শাসকের এহেন ঘৃণ্য ঘটনার প্রতিবাদে কবিগুরু রবীন্দ্রনাথ ঠাকুর ইংরেজদের দেওয়া নাইট উপাধি পরিত্যাগ করেন। ডায়ার প্রথমে ইংরেজ বাহিনী দ্বারা প্রশংসিত হলেও পরে ইংরাজী ১৯২০ সালে অপসৃত হন।

(e) Historical

Fig. 2.2 Different types of contents used for data collection

Table 2.1 Database name and the corresponding information

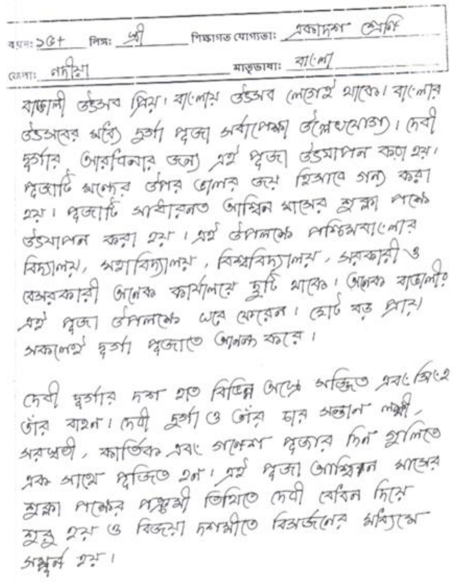
Set	Category	Number of		
		Words	Digit string	Character
A	Festive	118	0	594
B	Geographical	103	2	530
C	Sport	123	6	596
D	Technological	89	2	546
E	Historical	103	8	593



বয়সঃ ৩২ লিঙ্গঃ পুং শিক্ষাগত যোগ্যতাঃ স্নাতকোত্তর
জেলাঃ জলপাইগুড়ি মাতৃভাষাঃ বাংলা

সি. এম. এ. টি. ই. আর. ডি. বি. অপটিক্যাল ক্যারেক্টার
সনাক্তকরণ গবেষণার উপযোগী একটি ডাটাবেস।
ডাটাবেসটি তৈরি করেছে সেন্টার ফর মাইক্রোপ্রসেসর
অ্যাপ্লিকেশন ফর ট্রেইনিং এডুকেশন অ্যান্ড রিসার্চ
(সি. এম. এ. টি. ই. আর.) গবেষণাগার, যাদবপুর বিশ্ববিদ্যালয়,
কোলকাতা, ভারতবর্ষ। ডাটাবেসটি যেকোনোরকম অর্থায়ন
বিযুক্ত এবং গবেষণার কাজে বিনামূল্যে ব্যবহার করা যায়।
কিন্তু এই ডাটাবেস কোনোরকম বাণিজ্যিক উদ্দেশ্যে
ব্যবহার যোগ্য নয়। তবে গবেষণার কাজে এই ডাটাবেস
ব্যবহার করলে ওয়েবসাইট ও উল্লিখিত প্রকাশনা
গুলিকে নজির হিসেবে উল্লেখ করতে হবে। ইংরাজী
১০ ই জুন, ২০১৫ এর তথ্য অনুযায়ী ডাটাবেস এ
মোট ছয় ধরনের নথি (ডাটা সেট) আছে। এই
লেখাটিও একটি হাতে লেখা সংক্ষিপ্ত রচনার
ডাটাবেস এর অংশ হবে।

(a) An instance from Set-A



বয়সঃ ১৬+ লিঙ্গঃ স্ত্রী শিক্ষাগত যোগ্যতাঃ একাদশ শ্রেণি
জেলাঃ নদীয়া মাতৃভাষাঃ বাংলা

বাঙালী উৎসব প্রিয়। বাংলায় উৎসব লেগেই থাকে। বাংলার
উৎসবের মধ্যে দুর্গা পূজা সর্বাপেক্ষা উল্লেখযোগ্য। দেবী
দুর্গার আরাধনার জন্য এই পূজা উদযাপন করা হয়।
পূজাটি মন্দের উপর ভালোর জয় হিসাবে গন্য করা
হয়। পূজাটি সাধারনত আশ্বিন মাসের শুক্ল পক্ষে
উদযাপন করা হয়। এই উপলক্ষে পশ্চিমবাংলার
বিদ্যালয়, মহাবিদ্যালয়, বিশ্ববিদ্যালয়, সরকারী ও
বেসরকারী অনেক কার্যালয়ে ছুটি থাকে। অনেক বাঙালীরা
এই পূজা উপলক্ষে ঘরে ফেরেন। ছোট বড় প্রায়
সকলেই দুর্গা পূজাতে আনন্দ করে।

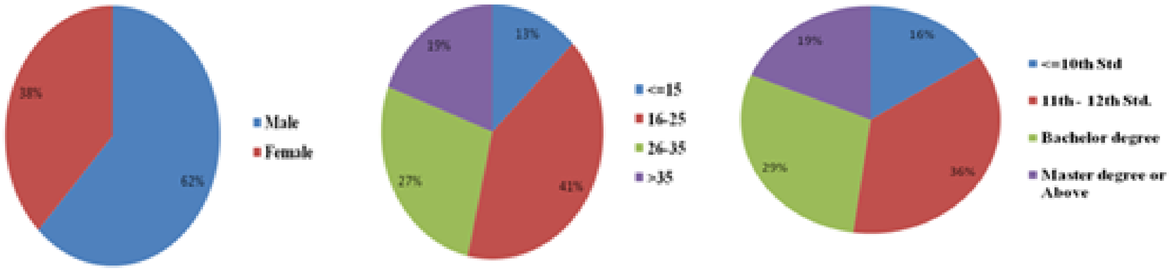
দেবী দুর্গার দশ হাত বিভিন্ন অস্ত্রে সজ্জিত এবং সিংহ
তাঁর বাহন। দেবী দুর্গা ও তাঁর চার সন্তান লক্ষ্মী,
সরস্বতী, কার্তিক এবং গণেশ পূজার দিন গুলিতে
এক সাথে পূজিত হন। এই পূজা আশ্বিন মাসের
শুক্লা পক্ষের ষষ্ঠী তিথিতে দেবী বোধন দিয়ে
শুরু হয় ও বিজয়া দশমীতে বিসর্জনের মাধ্যমে
সম্পূর্ন হয়।

(b) An instance from Set-B

Fig. 2.3 Sample of form like handwritten document page images

2.3 Data Processing

During this phase, the collected scanned document pages have been modified for further processing. First and foremost, all the document pages, mentioned previously, are enhanced through Middle of Modal Class (MMC) filtering technique [76] and then, passed through a ratio based binarization process [21, 32] to generate binarized images. Next, unwanted object pixels are removed from these binarized images and a Hough Transform (HT) [77] based page level skew correction is performed. Finally, form pages that belong to Set-A to Set-E are horizontally divided into two parts. The upper part contain writer’s information and lower one is the content portion. However, document pages that belong to Set-F do not undergo such division since they contain handwritten content only. All these data are stored both as binarized and gray-scale image for further use. A schematic diagram of the present preprocessing technique has been depicted in Fig. 2.6.

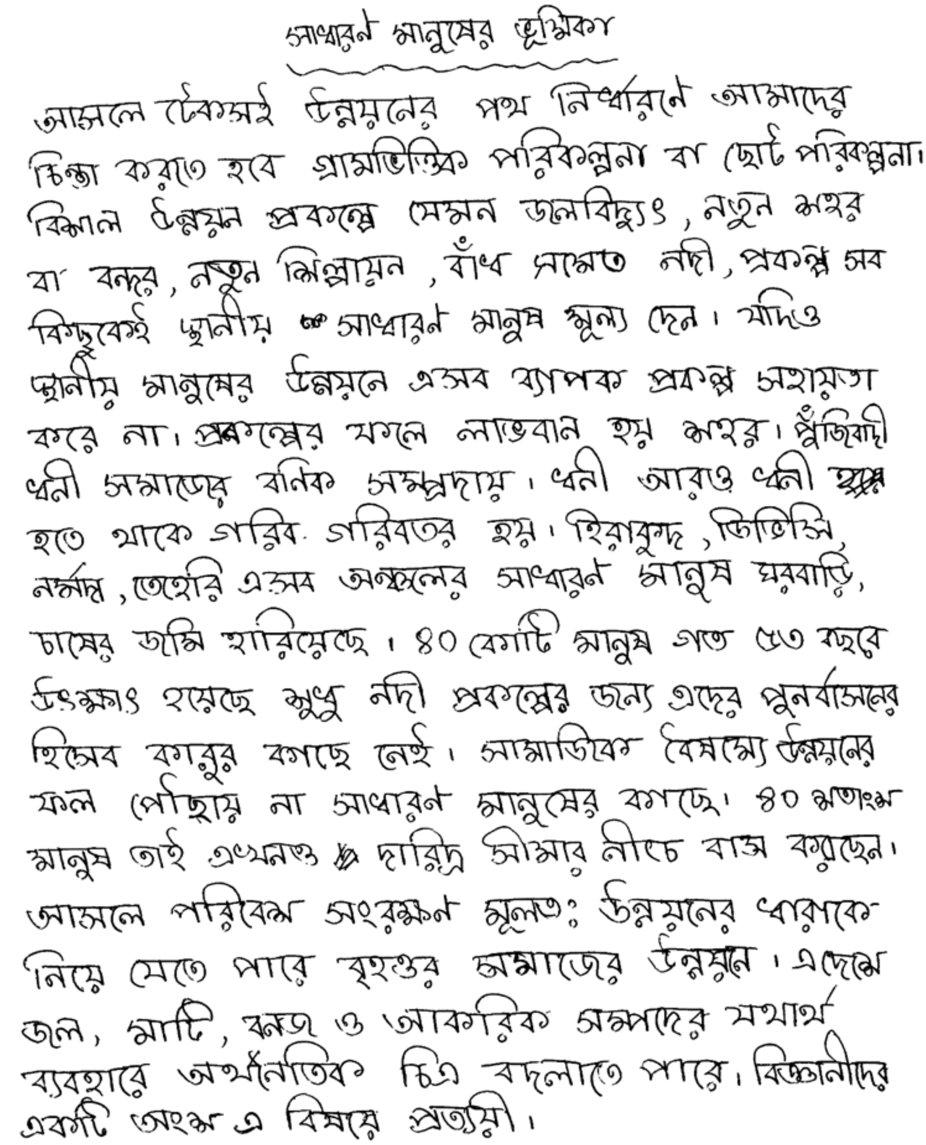


(a) (b) (c)

Fig. 2.4 Illustration of writers' information: (a) Sex (b) Age (in years) and (c) Educational Qualification

2.3.1 MMC filtering

Document image enhancement, a sub-field of image processing, is an important prerequisite for extracting the object pixels from a document image i.e., preparation of noise free image containing only the text information. A number of works [27, 76, 106-109] have been done on filtering techniques, which are applicable for digitized text pages or photographs, video signals, medical images etc. In the work [106], a diffusion based filtering technique (both linear and non-linear) has been implemented by solving an initial boundary value problem for the two dimensional diffusion equations with a special non-linear source. The authors [107] have designed a fuzzy function based image enhancement technique for image which is corrupted by impulse noise. In the works [27, 108-109], different non-linear filters (*namely*, mean [27], median [108] and order statistics filter [109]) have been introduced, where mean and median filters are very commonly used filtering mechanism but these fail to produce noise-free images or even introduce some distortions on the texts in the form of gulfs or capes. During the course of this thesis work, MMC filtering mechanism [76] has been introduced which performs better than its counterparts i.e., mean and median filter for noisy images (see Fig. 2.7). In this figure, to realize the effect of MMC filtering over others, a global threshold-based binarization technique is applied on the filtered images. The threshold value is estimated by calculating the mean of all modified gray level values of all the pixels in a document image.



সাধারন মানুষের ভূমিকা

আসলে টেকসই উন্নয়নের পথ নির্ধারনে আমাদের
চিন্তা করতে হবে গ্রামভিত্তিক পরিকল্পনা বা ছোট পরিকল্পনা।
বিশাল উন্নয়ন প্রকল্পে যেমন জলবিদ্যুৎ, নতুন শহর
বা বন্দর, নতুন শিল্পায়ন, বাঁধ সমেত নদী, প্রকল্প সব
কিছুকেই স্থানীয় সাধারন মানুষ মূল্য দেন। যদিও
স্থানীয় মানুষের উন্নয়নে এসব ব্যাপক প্রকল্প সহায়তা
করে না। প্রকল্পের ফলে লাভবান হয় শহর। পুঁজিবাদী
ধনী সমাজের বনিক সম্প্রদায়। ধনী আরও ধনী
হতে থাকে গরিব গরিবতর হয়। হিরাকুন্দ, ডিভিসি,
নর্মদা, তেহেরি এসব অঞ্চলের সাধারন মানুষ ঘরবাড়ি,
চাষের জমি হারিয়েছে। ৪০ কোটি মানুষ গত ৫৩ বছরে
উৎখাত হয়েছে শুধু নদী প্রকল্পের জন্য এদের পুনর্বাসনের
হিসেব কারুর কাছে নেই। সামাজিক বৈষম্যে উন্নয়নের
ফল পৌছায় না সাধারন মানুষের কাছে। ৪০ শতাংশ
মানুষ তাই এখনও দারিদ্র সীমার নীচে বাস করছেন।
আসলে পরিবেশ সংরক্ষন মূলত: উন্নয়নের ধারাকে
নিয়ে যেতে পারে বৃহত্তর সমাজের উন্নয়নে। এদেশে
জল, মাটি, বনজ ও আকরিক সম্পদের যথার্থ
ব্যবহারে অর্থনৈতিক চিত্র বদলাতে পারে। বিজ্ঞানীদের
একটি অংশ এ বিষয়ে প্রত্যয়ী।

Fig. 2.5 Instance of document page taken from CMATERdb1.1.1 (belonging to Set-F)

MMC is an adaptive statistical filtering method like mean, median, order-statistics filtering mechanisms. Here, a square window ($\mathcal{W}$) of size $w \times w$ pixels, where $w \in \mathbb{Z}^+$ and the value of w is chosen as $w = 2n + 1, \forall n \in \mathbb{N}$, is selected to perform the filtering. Then each window is slid over the document image from left to right and from top to bottom. Selection of new intensity value of the central pixel of $\mathcal{W}$ is carried out using modal value of pixel intensities enclosed by $\mathcal{W}$. Selection of mode over other central tendency statistics *like* mean, median is considered because it indicates the data that are most likely to occur. The MMC filtering mechanism is described in Algorithm 2.1 and an instance of MMC filtered output image is shown in Fig. 2.8. The original image is taken from present database (Set-A).

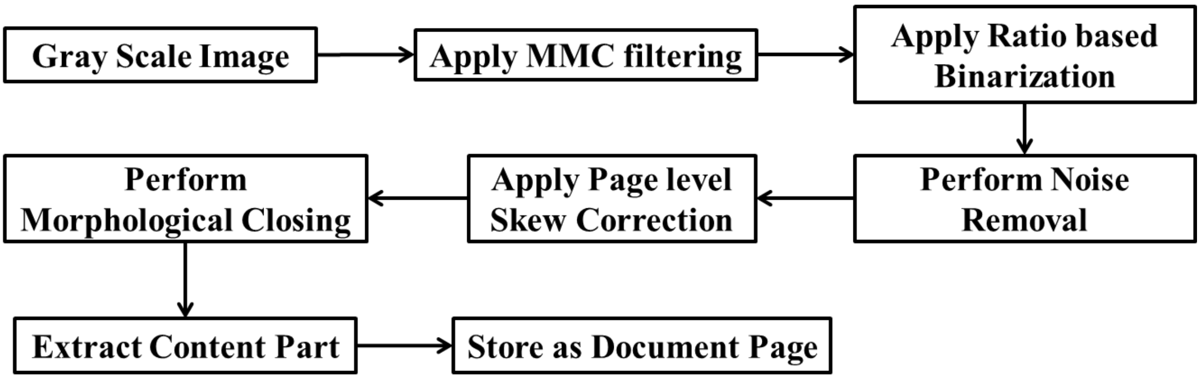


Fig. 2.6 Schematic diagram of handwritten content part extraction technique from form like datasheets which are collected here

Algorithm 2.1 MMC filtering

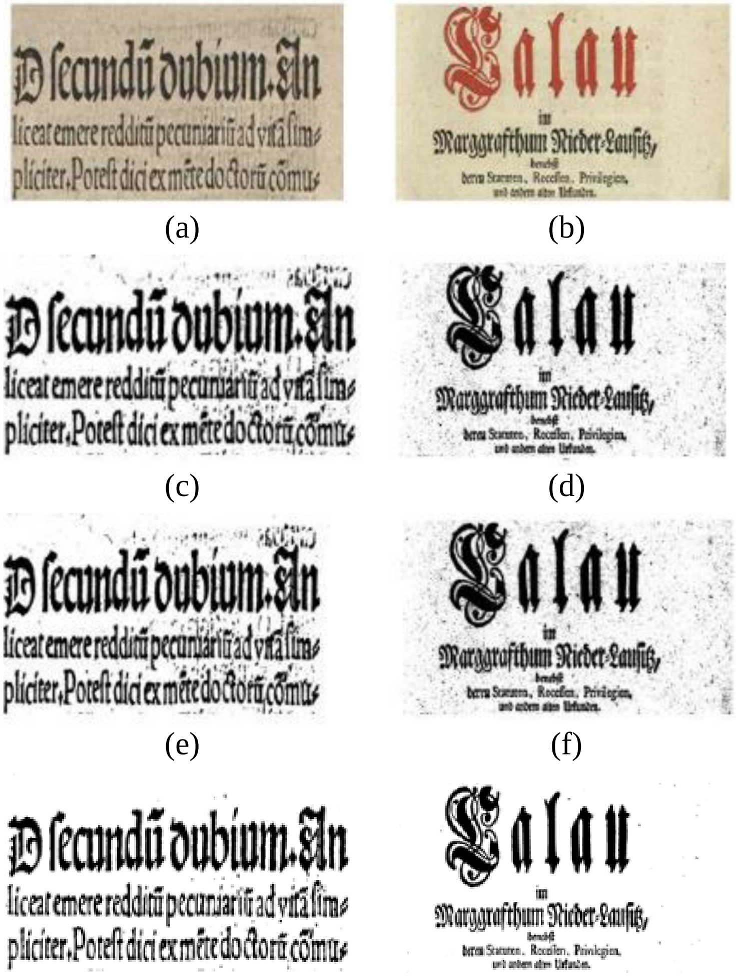
Input:

(i) Document page image, $\mathcal{P}_g = \{f(x,y) : (x,y) \in [1, H_P] \times [1, W_P] \wedge f(x,y) \in \{0,1,2,\dots,255\}\}$, where W_P and H_P are the width and height of $\mathcal{P}_g$ respectively.

(ii) A window, $\mathcal{W} = \{(x,y) : (x,y) \in [1,w] \times [1,w]\}$

Output: Filtered $\mathcal{P}_g$**,** $\mathcal{P}_f = \{g(x,y) : (x,y) \in [1, H_P] \times [1, W_P] \wedge g(x,y) \in \{0,1,2,\dots,255\}\}$.

```
Start
Step 1.  // Calculation of  minimum and maximum pixel intensities of $\mathcal{P}$
         Calculate $Min_P = \min\{f(x,y)\}, \forall (x,y) \in [1, H_P] \times [1, W_P]$
         Calculate $Max_P = \max\{f(x,y)\}, \forall (x,y) \in [1, H_P] \times [1, W_P]$
Step 2.  // Selecting modal class range
         $n_{MC} = w \times w - 1$, where $n_{MC}$ represent number of modal class. The number is
         selected using pigeon hole principle [38] which says if $m$ pigeons fly into $n$ pigeonholes
         ($m > n$), then at least one pigeonhole will contain two or more pigeons.
         $MC_R^i = [Min_P + (i-1) \times \frac{Max_P - Min_P}{n_{MC}}, Min_P + i \times \frac{Max_P - Min_P}{n_{MC}}]$, where$i = 1, 2, \dots, n_{MC}$ and
         $MC_R^i$ represents modal class range of $i^{th}$modal class. This choice of modal class range
         is image specific and helps in better choice of mode.
Step 3.  // Selection of new pixel intensity for some position of $\mathcal{W}$ over $\mathcal{P}$
         Initialize $\mathcal{H}(i)$ by zero, $\forall\ i = 1, 2, \dots, n_{MC}$
         $\mathcal{H}(i) = \mathcal{H}(i) + 1$, if $f(x,y) \in MC_R^i$ , $\forall (x,y) \in \mathcal{W}$
         Calculate$MC_{Index} = \max_i\{\mathcal{H}(i)\}$, $\forall i \in [1, n_{MC}]$, where $MC_{Index}$ is the modal class index
         Set, $MMC_{Intensity} = \left\lfloor \frac{MC_R^{t-1} + MC_R^t}{2} \right\rfloor$, where $t = MC_{Index}$ and $MMC_{Intensity}$ is the middle of
         modal class intensity.
Step 4.  // Transforming $\mathcal{P}_g$ to $\mathcal{P}_f$ for selected position of $\mathcal{W}$ in Step 3.
         $\mathcal{P}_f(c_x, c_y) = MMC_{Intensity}$, where $(c_x, c_y)$ is the center position of window $\mathcal{W}$.
Step 5.  Repeat Step 3 and Step 4 for all positions of $\mathcal{W}$ over $\mathcal{P}_g$
End
```


D ſecundū dubium. An
liceat emere redditū pecuniariū ad vitā ſim-
pliciter. Poteſt dici ex mēte doctorū cōmu-
Calau
im
Marggrafthum Nieder-Lauſitz,
benebſt
deren Statuten, Receſſen, Privilegien,
und andern alten Urkunden.

(a) (b)

(c) (d)

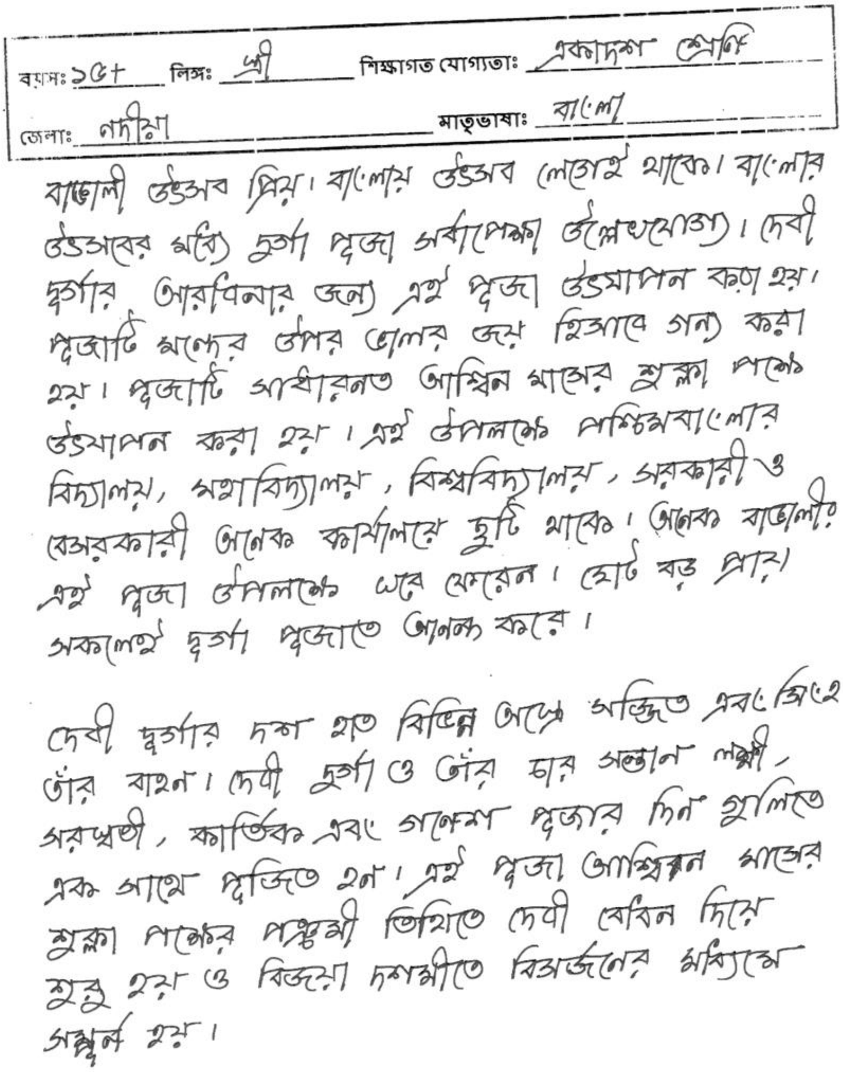
(e) (f)

(g) (h)

Fig. 2.7 Displaying the effect of MMC filtering over mean and median filters (window size 3 × 3 pixels) on noisy images. Here (a-b) Original Images (c-d) corresponding binarized image after applying mean filter (e-f) corresponding binarized image on applying median filter and (g-h) corresponding binarized image on applying MMC filter. These images are taken from the work [76]

2.3.2 Ratio based Binarization

A faster image binarization algorithm, which can distinguish object and background pixels of a document image, is the utmost requirement for performing content analysis of handwritten document pages. Binarization can be viewed as separating the object pixels from the background ones in a gray-scale image where object pixels appear darker than the background ones or vice versa. Numerous techniques have already been proposed to carry out this task. The fastest way among all available techniques is global threshold based image segmentation method. A thresholding technique tries to estimate a threshold (Th) value that can isolate object pixels from background ones.



বয়সঃ ১৫+ লিঙ্গঃ স্ত্রী শিক্ষাগত যোগ্যতাঃ একাদশ শ্রেণি
জেলাঃ নদীয়া মাতৃভাষাঃ বাংলা

বাঙালী উৎসব প্রিয়। বাংলায় উৎসব লেগেই থাকে। বাংলার উৎসবের মধ্যে দুর্গা পূজা সর্বাপেক্ষা উল্লেখযোগ্য। দেবী দুর্গার আরাধনার জন্য এই পূজা উদ্‌যাপন করা হয়। পূজাটি মন্দের উপর ভালোর জয় হিসাবে গন্য করা হয়। পূজাটি সাধারনত আশ্বিন মাসের শুক্লা পক্ষে উদ্‌যাপন করা হয়। এই উপলক্ষে পশ্চিমবাংলার বিদ্যালয়, মহাবিদ্যালয়, বিশ্ববিদ্যালয়, সরকারী ও বেসরকারী অনেক কার্যালয়ে ছুটি থাকে। অনেক বাঙালীরা এই পূজা উপলক্ষে ঘুরে বেড়ান। ছোট বড় প্রায় সকলেই দুর্গা পূজাতে আনন্দ করে।

দেবী দুর্গার দশ হাত বিভিন্ন অস্ত্রে সজ্জিত এবং সিংহ তাঁর বাহন। দেবী দুর্গা ও তাঁর চার সন্তান লক্ষ্মী, সরস্বতী, কার্তিক এবং গনেশ পূজার দিন গুলিতে এক সাথে পূজিত হন। এই পূজা আশ্বিন মাসের শুক্লা পক্ষের পঞ্চমী তিথিতে দেবী বোধন দিয়ে শুরু হয় ও বিজয়া দশমীতে বিসর্জনের মাধ্যমে সম্পূর্ণ হয়।

Fig. 2.8 Gray-scale version of a MMC filtered handwritten document page

Principally, Th is calculated using any global or local thresholding technique. Global thresholding uses a constant value irrespective of pixel position, whereas in its counterpart, threshold is calculated for each individual pixel. The optimal choice of Th is always a real challenge for the researchers. To choose optimal threshold value different types of iterative [33], recursive [110], multi-spectral [111], hierarchical [112] etc. thresholding strategies are devised by the researchers. Here, a global thresholding technique, entitled ratio based binarization [21, 32], is proposed which is a non-iterative in nature. It performs better than iterative [113] and Otsu's thresholding method [29] in noisy environment (see Fig. 2.9). Therefore, this binarization technique becomes useful as input image may contain noisy pixels. From the Algorithm 2.1, it is clear that $\mathcal{P}_f$ contains only $n \times n - 1$ i.e., $n^2 - 1$ different gray values after MMC filtering. Therefore, to select an optimal Th to binarize a document page, these distinct gray values are divided into three categories *namely*, Obvious Object *(OO)*, Obvious Background *(OB)* and Mixture of Object and Background *(MOB)*. Let, the ratio of OO, OB and MOB gray values is $a{:}\,b{:}\,c$, where $a, b, c \in \mathbb{N}$ and $a + b + c = n^2 - 1$. Such selection of ratio helps in pre-assigning pixels with intensity value below and above certain range (i.e., [a+1, b+1]) of gray values as object and background pixels respectively. Average of all MOB category pixels is considered as Th value here. Finally, $\mathcal{P}_g$ is converted to binarized image (i.e., $\mathcal{P}_b$) using this calculated Th value using eq. (1.1).

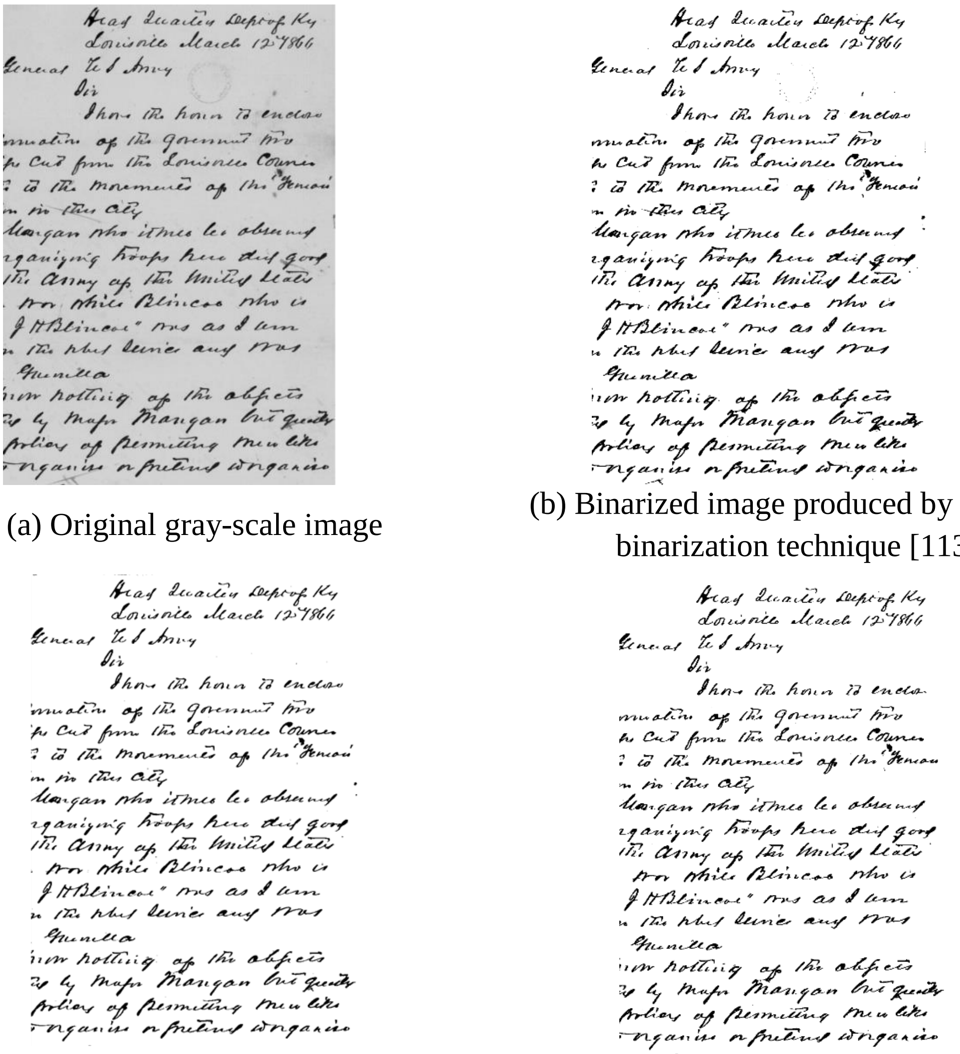
$$\mathcal{P}_b(x, y) = \begin{cases} 1, & if\ \mathcal{P}_g(x, y) \leq Th \\ 0, & otherwise \end{cases} \tag{1.1}$$

The values of n, a, b and c are set as 5, 2, 3 are 3 respectively. The values are decided experimentally on small number of samples using subjective measure. A binarized image of document page (i.e., $\mathcal{P}_b$) can formally be defined as $\mathcal{P}_b = \{f(x, y){:}\,(x, y) \in [1, H_P] \times [1, W_P] \wedge f(x, y) \in \{0, 1\}\}$. This means a document page image is converted to a binarized image which contain foreground/object pixels (intensity=1) and background pixels (intensity=0). An instance of binarized image using the ratio based technique is shown on Fig. 2.10.

2.3.3 Noise Removal

Binarized document page image, $\mathcal{P}_b$, generated in the previous stage, may contain unwanted object/background pixels (i.e., noise pixels). These pixels can broadly be categorized as (i) small cluster of object/background pixels (see Fig. 2.11 (a)) and (ii) spurious extensions [114] along character boundary (see Fig. 2.11 (a)). Please note that these kinds of noise are found frequently while binarizing noisy/degraded documents [21]. Therefore, in the current step, $\mathcal{P}_b$ is processed further to erase out these unwanted pixels.

To remove unwanted object pixel clusters, the binarized image is first labeled using 8-way connected component labelling (CCL) algorithm [115] and then all the connected components (CC) of size lesser than a pre-defined threshold value T_1 are removed. To remove the small cluster of background pixels, first the binarized image is complemented by interchanging object and background pixels and then the said CCL based technique is applied. In this case, CCs having pixels less than T_2 (a threshold value) have been removed. The values of T_1 and T_2 have been set empirically. Finally, to get rid of third category of noise, i.e., spurious extensions, morphological close operator with 3×3 structuring element is applied on the binarized image. An image after applying noise removal technique is shown in Fig. 2.11 (b).



(a) Original gray-scale image

(b) Binarized image produced by iterative binarization technique [113]

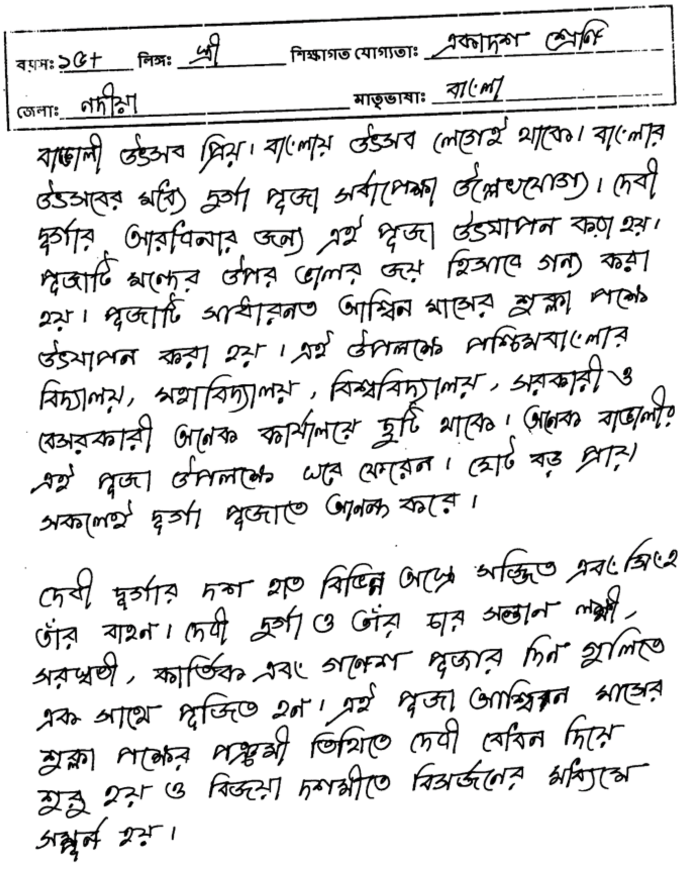
(c) Binarized image by Otsu’s binarization technique [29]

(d) Binarized image by ratio based binarization technique [21]

Fig. 2.9 Results of different global binarization techniques when applied on a noisy image

2.3.4 Page Level Skew Correction

Sometimes it is found that some of the collected filled-in form document images suffer from page-level skew. Reason for such skewedness is mostly due to wrong placement of (i) blank form on the photo copier’s bed to produce multiple copies or (ii) a filled up form on the scanner bed during digitization. One such instance of skewed document page is displayed in Fig. 2.10. This implies that a form level skew correction is required. Generally, this type skew correction, which is also sometimes called as window based skew correction, considers the entire document image as an atomic component. Detection and correction of such skew angle for document images are less challenging as the entire document requires only rotation with a particular angle to achieve skew corrected document image. The correction method, first, estimates the skew-angle and then corrects it.



বয়সঃ ১৫+ লিঙ্গঃ স্ত্রী শিক্ষাগত যোগ্যতাঃ একাদশ শ্রেণি

জেলাঃ নদীয়া মাতৃভাষাঃ বাংলা

বাঙালী উৎসব প্রিয়। বাংলায় উৎসব লেগেই থাকে। বাংলার উৎসবের মধ্যে দুর্গা পূজা সর্বাপেক্ষা উল্লেখযোগ্য। দেবী দূর্গার আরাধনার জন্য এই পূজা উদযাপন করা হয়। পূজাটি মন্দের উপর ভালোর জয় হিসাবে গন্য করা হয়। পূজাটি সাধারনত আশ্বিন মাসের শুক্লা পক্ষে উদযাপন করা হয়। এই উপলক্ষে পশ্চিমবাংলার বিদ্যালয়, মহাবিদ্যালয়, বিশ্ববিদ্যালয়, সরকারী ও বেসরকারী অনেক কার্যালয়ে ছুটি থাকে। অনেক বাঙালীরা এই পূজা উপলক্ষে ঘুরে বেরেন। ছোট বড় প্রায় সকলেই দূর্গা পূজাতে আনন্দ করে।

দেবী দূর্গার দশ হাত বিভিন্ন অস্ত্রে সজ্জিত এবং সিংহ তাঁর বাহন। দেবী দুর্গা ও তাঁর চার সন্তান লক্ষ্মী, সরস্বতী, কার্তিক এবং গনেশ পূজার দিন গুলিতে এক সাথে পূজিত হন। এই পূজা আশ্বিন মাসের শুক্লা পক্ষের পঞ্চমী তিথিতে দেবী বোধন দিয়ে শুরু হয় ও বিজয়া দশমীতে বিসর্জনের মাধ্যমে সম্পূর্ন হয়।

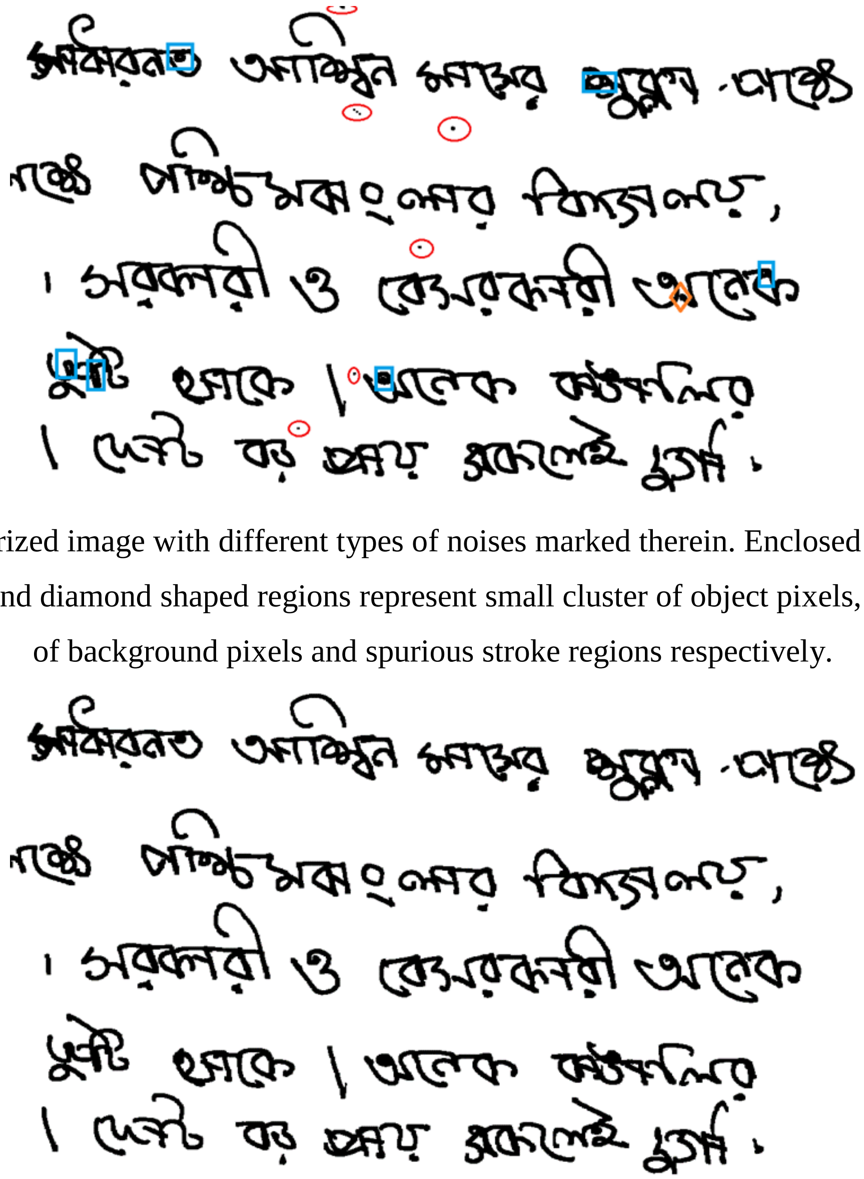
Fig. 2.10 Result after applying ratio based binarization on MMC filtered handwritten document page shown in Fig. 2.8

Researchers have devised several algorithms [77, 116-120] to estimate the skew angle. These algorithms are basically based on HT [77, 116], hierarchical HT [117], vertical projections with minimum bounding box [118], mathematical morphology [119] and by comparing the orientation of individual text regions with the arrangement of background space [120]. Out of these mechanisms, here, a HT based method, as described in [77], is used. The choice of HT method is due to the presence of prominent boundary line enclosing the author's information part (see Fig. 2.10) of the document pages and its following advantages

- It is robust to the presence of additional structures in the image.
- It is tolerant to noise.
- It is robust to partial or slightly deformed shapes.

2.3.4.1 Skew Angle Detection and Correction

In this module, the skew angle is detected using HT such that it can identify any $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, where θ is the angle at which the pages are aligned with positive direction of x-axis. The steps are described in Algorithm 2.2.



(a) Binarized image with different types of noises marked therein. Enclosed circular, rectangular and diamond shaped regions represent small cluster of object pixels, small cluster of background pixels and spurious stroke regions respectively.

(b) Noise free image

Fig. 2.11 Illustration of performance of present noise removal technique. A portion of document page is considered here

Algorithm 2.2 Skew Angle Detection

```
Input:
Binarized document page image, $\mathcal{P}_b = \{f(x,y): (x,y) \in [1,H_P] \times [1,W_P] \wedge f(x,y) \in \{0,1\}\}$,
where $W_P$ and $H_P$ are the width and height of $\mathcal{P}_b$ respectively.
Output:
Skew angle: $\theta$
Start

Step 1.   // Initializing Accumulator Array

          Quantize the HT space to identify the maximum and minimum values of $\rho$ and $\emptyset$,
          where $\rho\ is$ the perpendicular distance from the origin of a straight line $\mathrm{x}\cos(\emptyset) + y\sin(\emptyset) = \rho$
          and $\theta$ is the angle of inclination of the perpendicular with respect to the x-axis.
          Generate an accumulator array $A(\rho,\emptyset)$. Set all the array elements to zero.

Step 2.   // Calculating accumulator array elements

          $\forall (x_i, y_i)\ \epsilon\ S$, where $S$ is the set of object pixel points in $\mathcal{P}_b$ and $\forall\ \emptyset\ \epsilon\ [-\frac{\pi}{2}, \frac{\pi}{2}]$ perform
          the following steps:
                    i.    Compute $\rho$ from the equation $\rho = \mathrm{x_i}\cos(\theta) + y_i\ \sin(\theta)$

                    ii.   Increment  $A(\rho,\theta)$ by 1

Step 3.   // Estimating θ

          Find $\max\ \{A(\rho,\emptyset)\}$. Let the value is $max_A$.

          Find the ∅ index for the cell containing $max_A$. Say the value is $\emptyset_{estimated}$

          Set $\theta = \frac{\pi}{2} - \emptyset_{estimated}$

End
```

In the implementation, $\emptyset$ varies from $-\frac{\pi}{2}$ to $\frac{\pi}{2}$ with fractional increment say, δ. So, the number of columns in $A(\rho,\emptyset)$ is $\left(\frac{\frac{\pi}{2} - \left(-\frac{\pi}{2}\right)}{\delta}\right) \times \rho$ and the value of δ is set as $\delta\ = 0.2 \times \frac{\pi}{180}$. Two different instances of an accumulator array are shown in Fig. 2.12(a-b). In Fig 2.12(b), negative ρ value is found. This may happen when $\emptyset \in [-90^{\circ}, 0^{\circ})$ and $|y_i \sin\emptyset| > |x_i \cos\emptyset\,|$.

2.3.4.2 Skew Correction

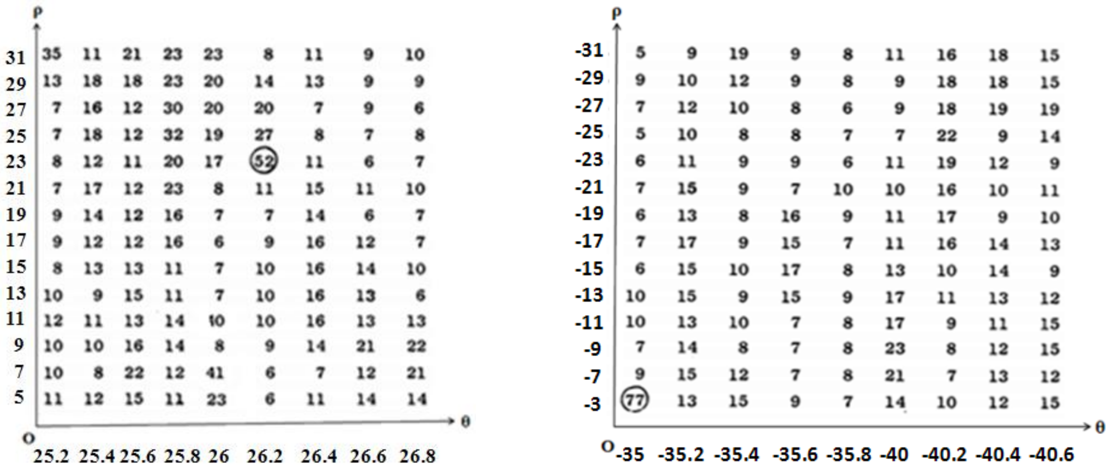
Based on the detected skew angle (θ), $\mathcal{P}_b$ and $\mathcal{P}_g$ are rotated using affine transformation. The new pixel position after affine transformation is decided by applying bicubic interpolation algorithm [121]. It is to be noted that for smaller skew angle (i.e., $\theta \in [-\frac{\pi}{180}, \frac{\pi}{180}]$) no rotation is performed to maintain the original quality of the image. The rotation is performed using the following rules:

i. If $\theta \in (\frac{\pi}{180}, \frac{\pi}{2}]$, then the $\mathcal{P}_b$ and $\mathcal{P}_g$are rotated clock-wise with angle θ.

ii. If $\theta \in [-\frac{\pi}{2}, -\frac{\pi}{180})$, then the $\mathcal{P}_b$ and $\mathcal{P}_g$ are rotated anti clock-wise with angle θ.

iii. If $\theta \in [-\frac{\pi}{180}, \frac{\pi}{180}]$, no such rotation operation is performed.

A skew-corrected filled-in form image is shown in Fig. 2.13.



	25.2	25.4	25.6	25.8	26	26.2	26.4	26.6	26.8
31	35	11	21	23	23	8	11	9	10
29	13	18	18	23	20	14	13	9	9
27	7	16	12	30	20	20	7	9	6
25	7	18	12	32	19	27	8	7	8
23	8	12	11	20	17	52	11	6	7
21	7	17	12	23	8	11	15	11	10
19	9	14	12	16	7	7	14	6	7
17	9	12	12	16	6	9	16	12	7
15	8	13	13	11	7	10	16	14	10
13	10	9	15	11	7	10	16	13	6
11	12	11	13	14	10	10	16	13	13
9	10	10	16	14	8	9	14	21	22
7	10	8	22	12	41	6	7	12	21
5	11	12	15	11	23	6	11	14	14

(a) Instance of accumulator array with positive indices

	-35	-35.2	-35.4	-35.6	-35.8	-40	-40.2	-40.4	-40.6
-31	5	9	19	9	8	11	16	18	15
-29	9	10	12	9	8	9	18	18	15
-27	7	12	10	8	6	9	18	19	19
-25	5	10	8	8	7	7	22	9	14
-23	6	11	9	9	6	11	19	12	9
-21	7	15	9	7	10	10	16	10	11
-19	6	13	8	16	9	11	17	9	10
-17	7	17	9	15	7	11	16	14	13
-15	6	15	10	17	8	13	10	14	9
-13	10	15	9	15	9	17	11	13	12
-11	10	13	10	7	8	17	9	11	15
-9	7	14	8	7	8	23	8	12	15
-7	9	15	12	7	8	21	7	13	12
-3	77	13	15	9	7	14	10	12	15

(b) Instance of accumulator array with negative indices

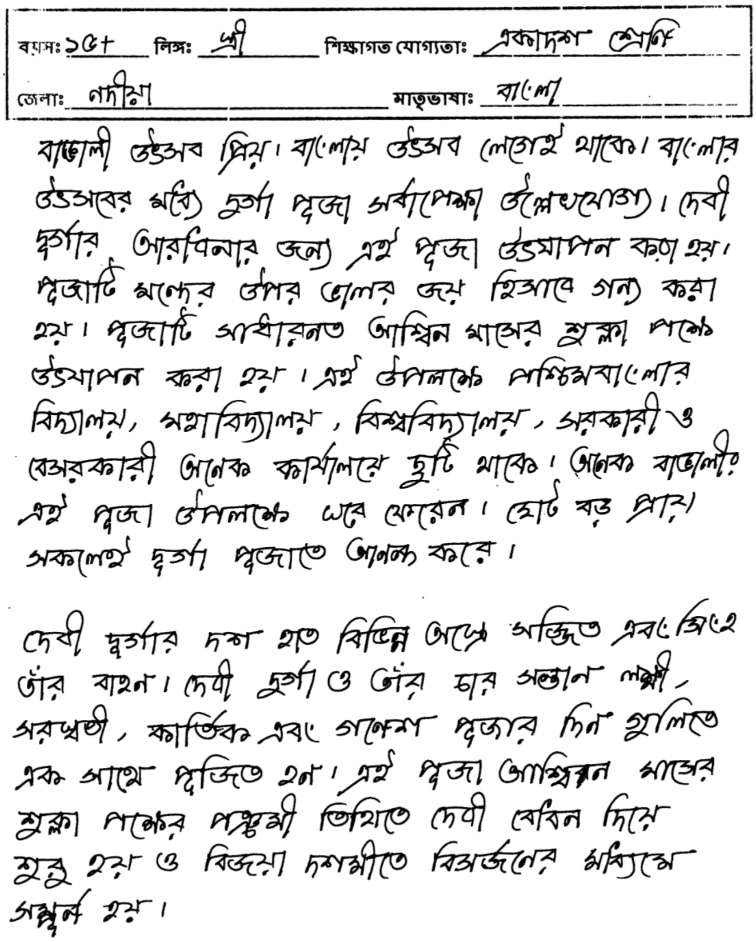
Fig. 2.12 (a-b) Sample instances of accumulator

2.3.4 Partitioning of Form Document Page

It is already stated that to conduct experiments for this thesis work, a new database has been prepared. The form document pages that belong to Set-A to Set-E have two different parts which are *namely*, content part and writer information part. In the present section, an automated system has been devised to crop out the content part (i.e., portion containing only handwritten text) from the skew corrected document pages which is described through Algorithm 2.3.

Content part from $\mathcal{P}_g{}'$s (i.e., from gray-scale form document image) is first cropped out using S_R and E_R and then the background region surrounding the text parts are removed using the values of TR, BR, LC and RC that are calculated in Algorithm 2.3. The final content part is stored as gray-scale image for future use. The extracted content and author's information parts for a single filled-in form document are shown in Fig. 2.14. More reference examples of content part could be found in Appendix (Figs. A1-A6).

Now onward the content part are considered as page image unless specified otherwise. The gray-scale page image is termed as $\mathcal{P}_g$ (see Fig. 2.14(a)) while its binarized form is represented by $\mathcal{P}_b$ (see Fig. 2.14(b)). However, simply to mean a page image the notation $\mathcal{P}$ is used. The height and width of the page image are termed as $\mathcal{P}_H$ and P_W that are calculated as $\mathcal{P}_H = TR - BR + 1$ and $\mathcal{P}_W = RC - LC + 1$.



বয়সঃ ১৫+ লিঙ্গঃ স্ত্রী শিক্ষাগত যোগ্যতাঃ একাদশ শ্রেণি

জেলাঃ নদীয়া মাতৃভাষাঃ বাংলা

বাঙালী উৎসব প্রিয়। বাংলায় উৎসব লেগেই থাকে। বাংলার
উৎসবের মধ্যে দুর্গা পূজা সর্বাপেক্ষা উল্লেখযোগ্য। দেবী
দুর্গার আরাধনার জন্য এই পূজা উদ্‌যাপন করা হয়।
পূজাটি মন্দের উপর ভালোর জয় হিসাবে গন্য করা
হয়। পূজাটি সাধারনত আশ্বিন মাসের শুক্লা পক্ষে
উদ্‌যাপন করা হয়। এই উপলক্ষে পশ্চিমবাংলার
বিদ্যালয়, মহাবিদ্যালয়, বিশ্ববিদ্যালয়, সরকারী ও
বেসরকারী অনেক কার্যালয়ে ছুটি থাকে। অনেক বাঙালীরা
এই পূজা উপলক্ষে ঘুরে বেড়ান। ছোট বড় প্রায়
সকলেই দুর্গা পূজাতে আনন্দ করে।

দেবী দুর্গার দশ হাত বিভিন্ন অস্ত্রে সজ্জিত এবং সিংহ
তাঁর বাহন। দেবী দুর্গা ও তাঁর চার সন্তান লক্ষ্মী,
সরস্বতী, কার্তিক এবং গনেশ পূজার দিন গুলিতে
এক সাথে পূজিত হন। এই পূজা আশ্বিন মাসের
শুক্লা পক্ষের পঞ্চমী তিথিতে দেবী বোধন দিয়ে
শুরু হয় ও বিজয়া দশমীতে বিসর্জনের মাধ্যমে
সম্পূর্ন হয়।

Fig. 2.13 An instance of a skew corrected filled in form document image. The output is generated by applying present page-level skew correction technique on the image shown in Fig. 2.10

Algorithm 2.3 Partitioning a document page

Input:

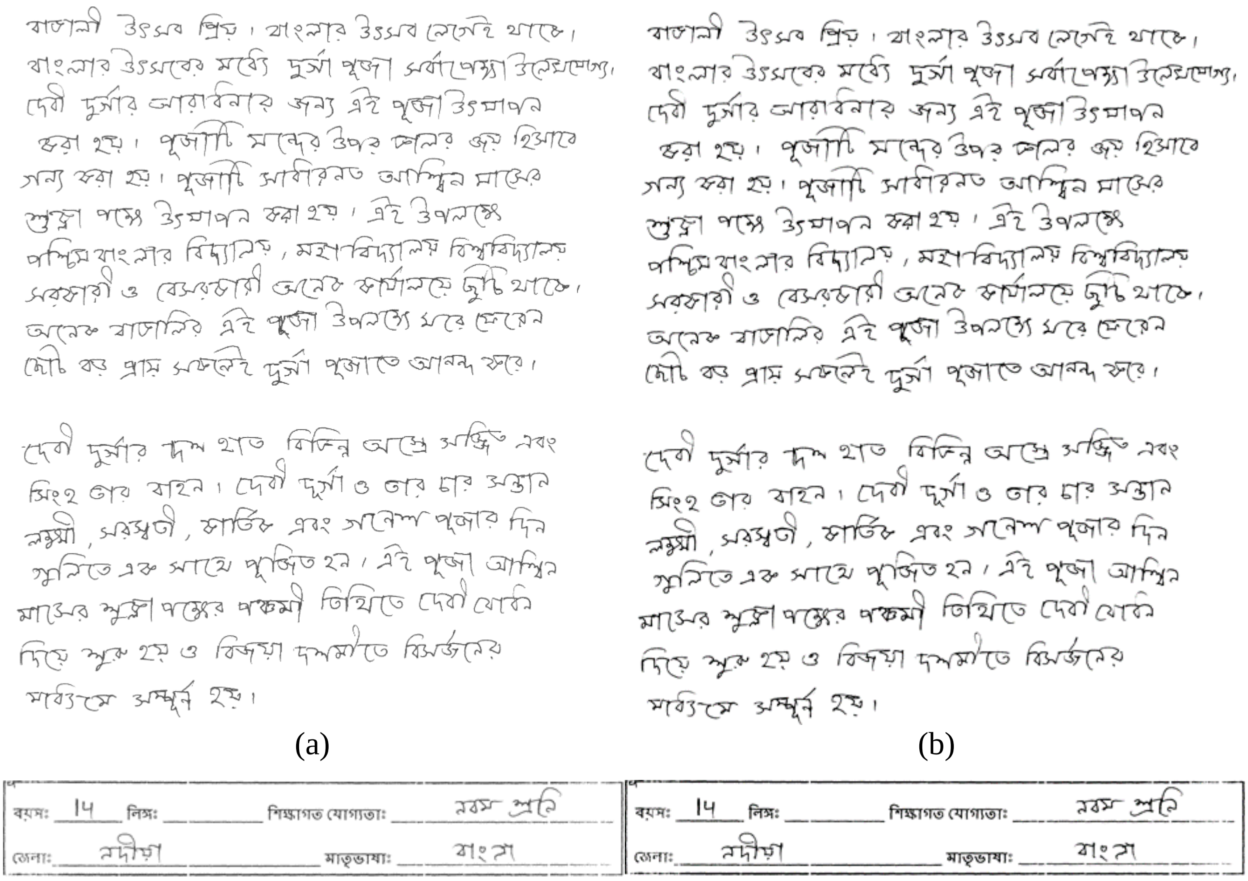
Skew corrected binarized document page image, $\mathcal{P}_b = \{f(x,y): (x,y) \in [1, H_P] \times [1, W_P] \wedge f(x,y) \in \{0,1\}\}$, where W_P and H_P are the width and height of $\mathcal{P}_b$ respectively.

Output:

(i) Content part, $\mathcal{C}_b = \{f(x,y): (x,y) \in [1, H_\mathcal{C}] \times [1, W_\mathcal{C}] \wedge f(x,y) \in \{0,1\}\}, where\ W_\mathcal{C}\ and\ H_\mathcal{C}\ are\ the\ width\ and\ height\ of\ \mathcal{C}_b\ respectively.$

(ii) Writer's information part, $\mathcal{A}_b = \{f(x,y): (x,y) \in [1, H_\mathcal{A}] \times [1, W_\mathcal{A}] \wedge f(x,y) \in \{0,1\}\}, where\ W_\mathcal{A}\ and\ H_\mathcal{A}\ are\ the\ width\ and\ height\ of\ \mathcal{A}_b\ respectively.$

```
Start

Step 1.  // Calculation of normalized horizontal histogram $h(i)$, i=1 to $H_P$

         Calculate $h(i) = \frac{|\{j: f(i,j) = '1' \wedge 1 \le j \le W_P|}{W_P}$, where, $1 \le i \le H_P$

Step 2.  // Selection of separation line on $\mathcal{P}_b$ which is also the end row of $\mathcal{A}_b$

         Let, $S_R$ and $E_R$ are the start row and end row of $\mathcal{A}_b$ respectively.

         $S_R = \min\{i: h(i) \neq 0 \wedge 1 \le i \le H_P\}$

         Construct a set of row indices (say, $R$) such that $R = \{r: r \ge \delta_1 \wedge (r - S_R) \ge \delta_2\}$,
         where $\delta_i$'s ($i = 1,2$) are predefined threshold values.

         Set $E_R = \min\{R\}$.

Step 3.  // Separation of $\mathcal{P}_b$into $\mathcal{C}_b$ and $\mathcal{A}_b$

         Crop out the region between $S_R$ and $E_R$ from $\mathcal{P}_b$and store as $\mathcal{A}_b$

         Crop out the region within $(E_R + 1)$ and $H_P$ and store as $\mathcal{C}_b$

Step 4.  // Removing background region surrounding $\mathcal{C}_b$ to store with minimal bounding box
         enclosing all the object pixels with in it.

         Let, $TR$, $BR$, $LC$ and $RC$ are respectively the top most row, bottom most row, left
         most column and right most column of $\mathcal{C}_b$ that contain at least one object pixel.

         Find normalized horizontal (say, $h_h(i)$, $1 \le i \le H_{\mathcal{C}}$) and vertical (say, $h_v(i)$, $1 \le i \le$
         $W_{\mathcal{C}}$) histogram of $\mathcal{C}_b$

         $h_h(i) = \frac{|\{j: f(i,j) = '1' \wedge 1 \le j \le H_{\mathcal{C}}\}|}{W_{\mathcal{C}}}$, where, $1 \le i \le H_{\mathcal{C}}$

         $h_v(j) = \frac{|\{i: f(i,j) = '1' \wedge 1 \le i \le W_{\mathcal{C}}\}|}{W_{\mathcal{C}}}$, where, $1 \le j \le W_{\mathcal{C}}$

         Calculate $TR$, $BR$, $LC$ and $RC$ as

         $TR = \min\{i: h_h(i) \neq 0 \wedge 1 \le i \le H_{\mathcal{C}}\}$

         $BR = \max\{i: h_h(i) \neq 0 \wedge 1 \le i \le H_{\mathcal{C}}\}$.

         $LC = \min\{i: h_v(i) \neq 0 \wedge 1 \le i \le W_{\mathcal{C}}\}$

         $RC = \max\{i: h_v(i) \neq 0 \wedge 1 \le i \le W_{\mathcal{C}}\}$.

         Crop $\mathcal{C}_b$based on $TR$, $BR$, $LC$ and $RC$ information.

End
```



বাঙালী উৎসব প্রিয়। বাংলার উৎসব লেগেই থাকে।
বাংলার উৎসবের মধ্যে দুর্গা পূজা সর্বশ্রেষ্ঠ উৎসবগুলো।
দেবী দুর্গার আরাধনার জন্য এই পূজা উৎযাপন
করা হয়। পূজাটি মন্দের উপর ভালোর জয় হিসাবে
পালন করা হয়। পূজাটি সাধারণত আশ্বিন মাসের
শুক্লা পক্ষে উৎযাপন করা হয়। এই উপলক্ষে
পশ্চিমবাংলার বিদ্যালয়, মহাবিদ্যালয় বিশ্ববিদ্যালয়
সরকারী ও বেসরকারী অনেক কার্যালয়ে ছুটি থাকে।
অনেক বাঙালির এই পূজা উপলক্ষে ঘরে ফেরেন
ছোট বড় প্রায় সকলেই দুর্গা পূজাতে আনন্দ করে।

দেবী দুর্গার দশ হাত বিভিন্ন অস্ত্রে সজ্জিত এবং
সিংহ তার বাহন। দেবী দুর্গা ও তার চার সন্তান
লক্ষ্মী, সরস্বতী, কার্তিক এবং গনেশ পূজার দিন
গুলিতে এক সাথে পূজিত হন। এই পূজা আশ্বিন
মাসের শুক্লা পক্ষের পঞ্চমী তিথিতে দেবী বোধন
দিয়ে শুরু হয় ও বিজয়া দশমীতে বিসর্জনের
মাধ্যমে সম্পূর্ন হয়।

(a)

বাঙালী উৎসব প্রিয়। বাংলার উৎসব লেগেই থাকে।
বাংলার উৎসবের মধ্যে দুর্গা পূজা সর্বশ্রেষ্ঠ উৎসবগুলো।
দেবী দুর্গার আরাধনার জন্য এই পূজা উৎযাপন
করা হয়। পূজাটি মন্দের উপর ভালোর জয় হিসাবে
পালন করা হয়। পূজাটি সাধারণত আশ্বিন মাসের
শুক্লা পক্ষে উৎযাপন করা হয়। এই উপলক্ষে
পশ্চিমবাংলার বিদ্যালয়, মহাবিদ্যালয় বিশ্ববিদ্যালয়
সরকারী ও বেসরকারী অনেক কার্যালয়ে ছুটি থাকে।
অনেক বাঙালির এই পূজা উপলক্ষে ঘরে ফেরেন
ছোট বড় প্রায় সকলেই দুর্গা পূজাতে আনন্দ করে।

দেবী দুর্গার দশ হাত বিভিন্ন অস্ত্রে সজ্জিত এবং
সিংহ তার বাহন। দেবী দুর্গা ও তার চার সন্তান
লক্ষ্মী, সরস্বতী, কার্তিক এবং গনেশ পূজার দিন
গুলিতে এক সাথে পূজিত হন। এই পূজা আশ্বিন
মাসের শুক্লা পক্ষের পঞ্চমী তিথিতে দেবী বোধন
দিয়ে শুরু হয় ও বিজয়া দশমীতে বিসর্জনের
মাধ্যমে সম্পূর্ন হয়।

(b)

বয়স: 14 লিঙ্গ: ______ শিক্ষাগত যোগ্যতা: নবম শ্রেণি
জেলা: নদীয়া মাতৃভাষা: বাংলা

(c)

বয়স: 14 লিঙ্গ: ______ শিক্ষাগত যোগ্যতা: নবম শ্রেণি
জেলা: নদীয়া মাতৃভাষা: বাংলা

(d)

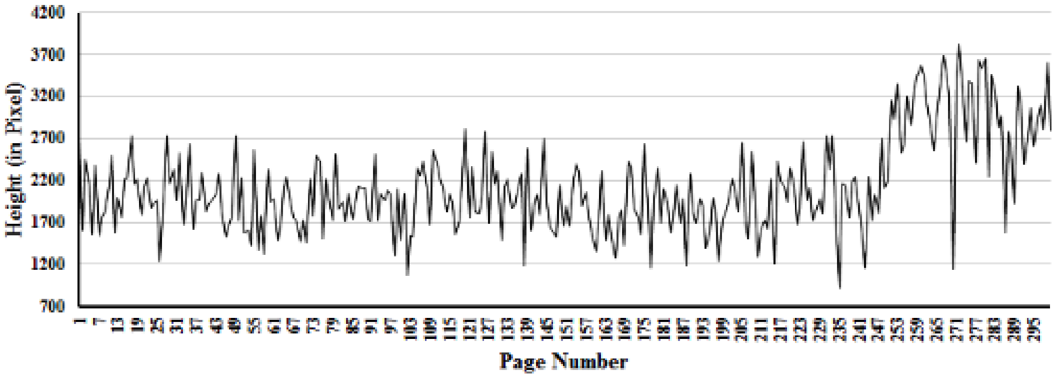
Fig. 2.14 Displaying content part (a, b) and authors information part (c, d) that are extracted from a filled-in form. In this figure (a, c) are gray-scale images while (b, d) are the corresponding binarized image

2.4 Meta-data

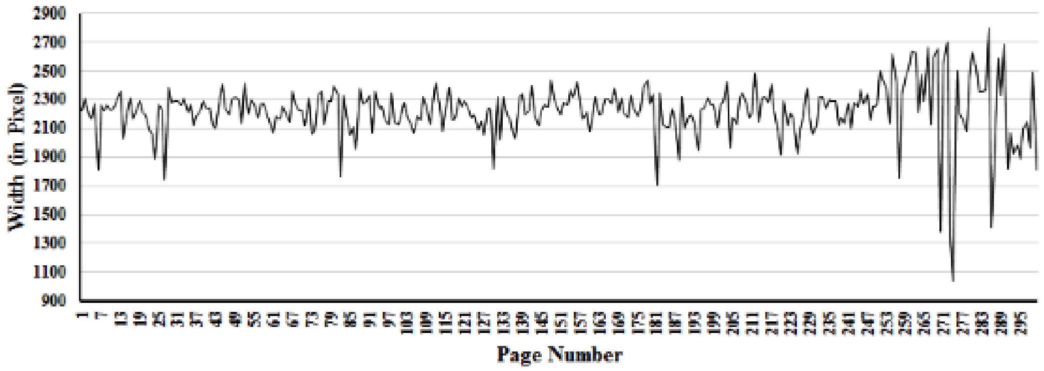
As already mentioned, the present databases contains 300 documents which belong to six different datasets *namely*, Set-A, B, C, D, E and F. The writing samples except the samples of Set-F are collected in pre-defined form pages following IAM data collection technique. Next, the content section from this form like document pages are cropped out and stored. Whereas the pages, belonging to Set-F, are kept as it is. Now onward all these samples are considered as document pages. To describe variety of these document pages (i.e., $\mathcal{P}$), features as shown in Table 2.2 are extracted from each document page. Also the page-wise information of the mentioned features of the pages are shown graphically in Fig. 2.15 (a-l). From these charts it is clear that the database contains ample writing variations. Moreover, to describe the nature of the collected data samples, the statistics like maximum, minimum, mean, standard deviation (SD), first quartile (Q1), second quartile (Q2) i.e., median, third quartile (Q3) and inter quartile range (IQR) [122] are also calculated for each of the features that extracted from each of the document pages. These statistical information are recorded in Table 2.3.

Table 2.2 Description of statistical parameters used to define page images

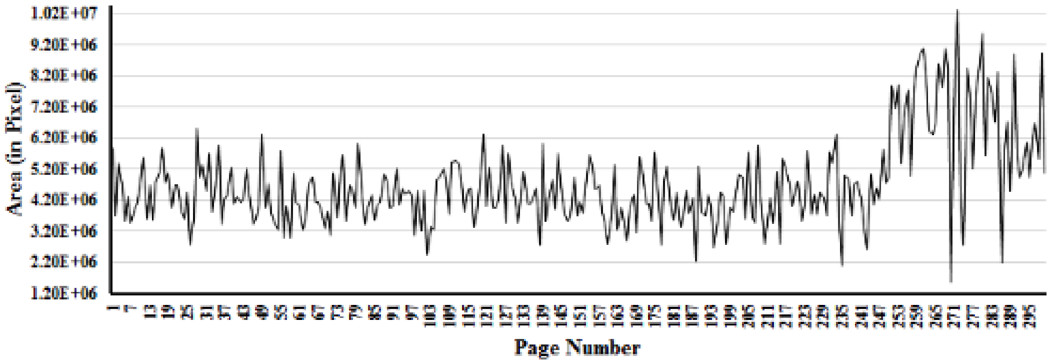
SL #	Term	Description	Remarks	Variation shown in
1	$H_{\mathcal{C}}, W_{\mathcal{C}}$	Height and width of $\mathcal{P}$.	Represent the region of object pixels within an input page image.	Fig 2.15 (a- b)
2	$Area$	Measured as $\mathcal{P}_H \times \mathcal{P}_W$.	Fundamental geometric information for page images	Fig 2.15 (c)
3	$\#CC$	Number of CCs in $\mathcal{P}$.	Describes the information like 1, 2, 3, … N- connected pages	Fig 2.15 (d)
4	AR	Aspect ratio measured as $\frac{\mathcal{P}_W}{\mathcal{P}_H}$.	Provides information related to geometrical structure of an image	Fig 2.15 (e)
5	$pixelDen$	Number of object pixels per unit area word image i.e., $\frac{totalPix}{area}$	Provides variation of image resolution from one image to another	Fig 2.15 (f)
6	$avgStrokeHt$	Horizontal stroke characteristic	Indicate the writers' writing style	Fig 2.15 (g)
7	$avgStrokeWd$	Vertical stroke characteristic		Fig 2.15 (h)
8	$lineCount$	Number of text lines (TLs) present in $\mathcal{P}$	Basic information about content of an text image	Fig 2.15 (i)
9	$wordCount$	Number of words present in $\mathcal{P}$		Fig 2.15 (j)
10	$avgCCPerLine$	Average number of CC per text line in $\mathcal{P}$	Present the complexity in extracting TL/word extraction from document image	Fig 2.15 (k)
11	$avgCCPerWord$	Average number of CC per word in $\mathcal{P}$		Fig 2.15 (l)



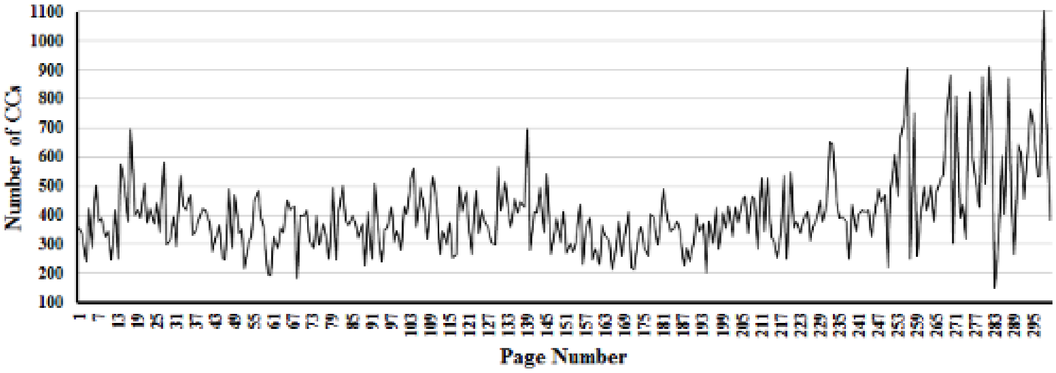
(a) Variation in document page height



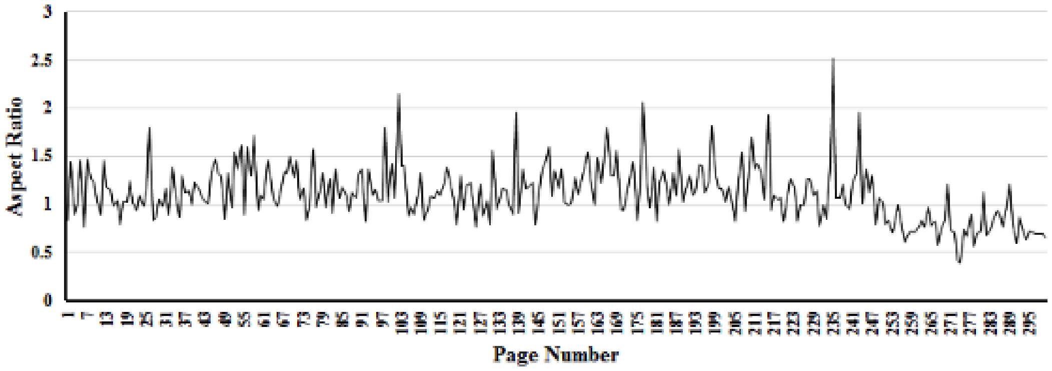
(b) Variation in document page width



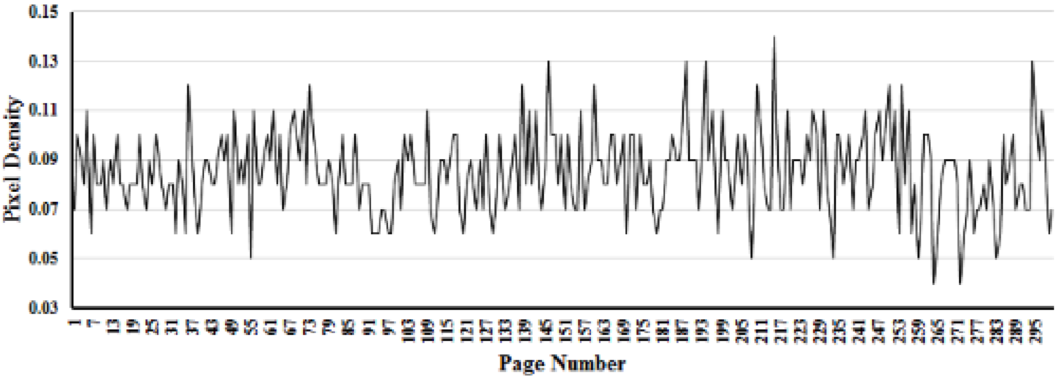
(c) Variation in area feature



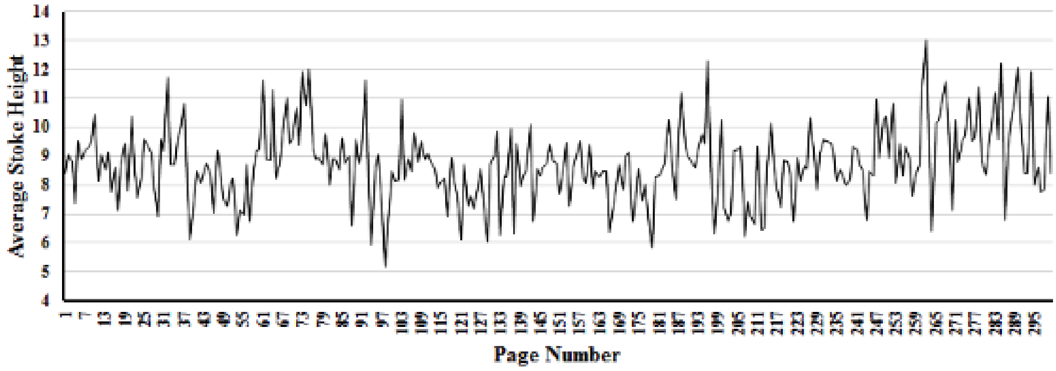
(d) Variation in number of CCs



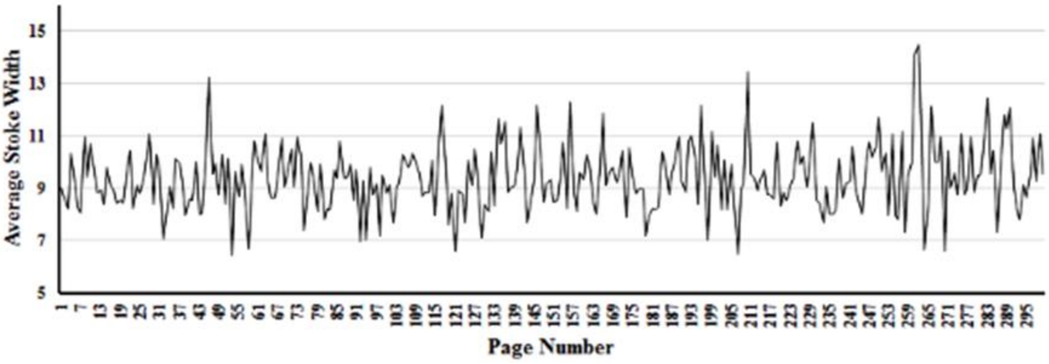
(e) Variation in aspect ratio



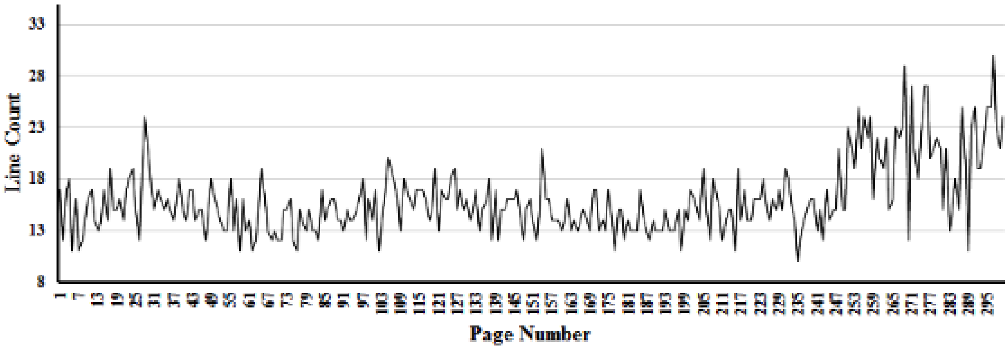
(f) Variation in pixel density feature



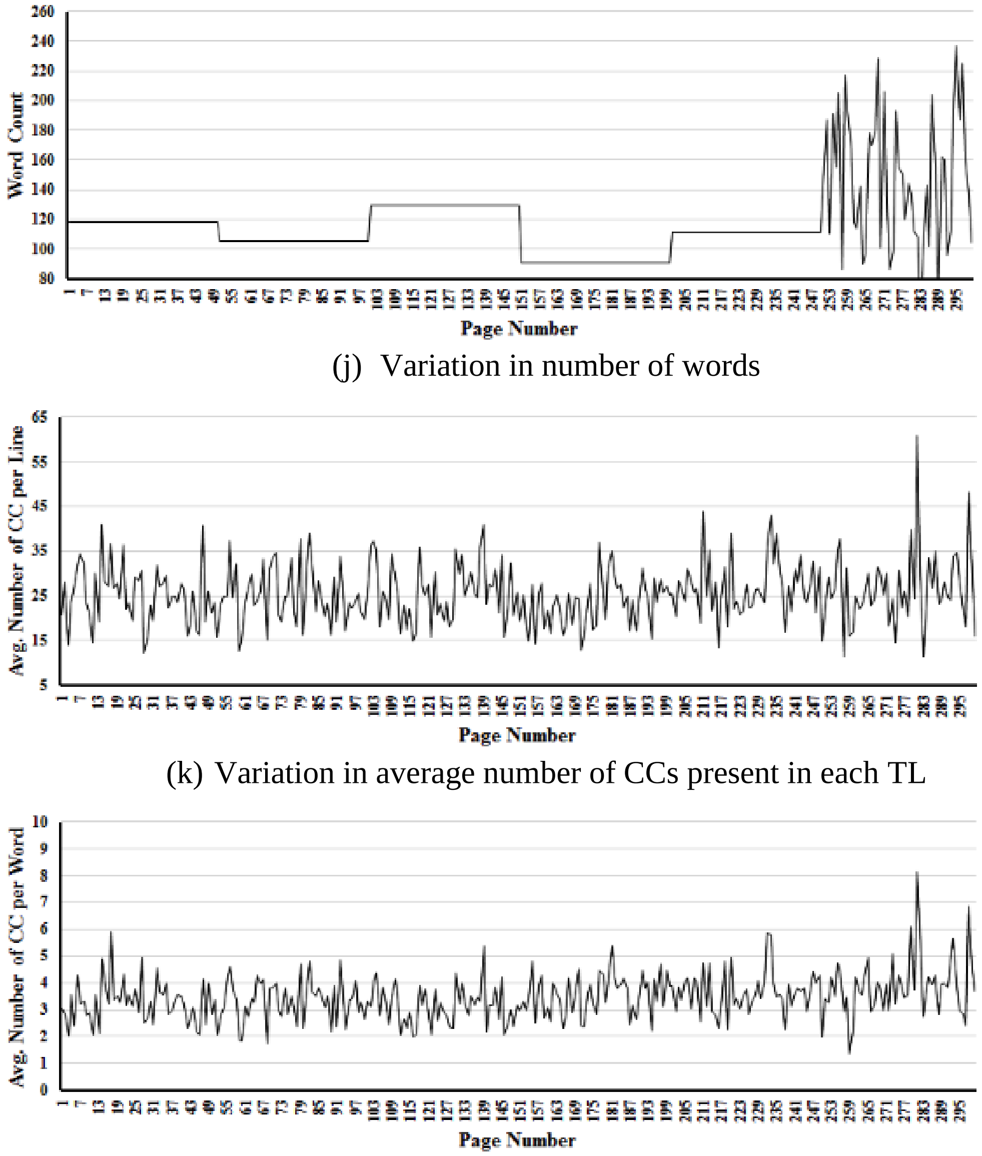
(g) Variation in average stroke height



(h) Variation in average stroke width



(i) Variation in number of TLs



(j) Variation in number of words

(k) Variation in average number of CCs present in each TL

(l) Variation in average number of CCs present in each word

Fig. 2.15 Depiction of variations of different page-level statistics, listed in Table 2.2, for entire database

2.5 Discussion

As already mentioned that to conduct any content analysis related research problem sufficient amount of data is required. Therefore in the present chapter, a database is prepared which contains 300 document pages. 80% of these document pages, containing any of the five different said categories, are collected from different individuals. These document pages are collected following the IAM data collection standard. Ample variations in terms of writer's group, sex and educational qualification are there. In addition to this, 20% of the entire data is taken from CMATERdb1.1.1, a publicly available repository for page level handwritten document pages.

Table 2.3 Description of statistical parameters used to define page images

Term	Minimum	Maximum	Average	SD	Q1	Q2	Q3	IQR
$\boldsymbol{H_c}$	915	3826	2123.42	559.58	1732	2024.5	2404	672
$\boldsymbol{W_c}$	1041	2799	2220.87	192.92	2146.5	2235.5	2312.5	166
$\boldsymbol{Area}$	1589980	10334026	4736263.63	1444747.12	3771670	4407656.5	5271135	1499465
$\boldsymbol{\#CC}$	148	1110	404.55	137.32	319.5	383	450.5	131
$\boldsymbol{AR}$	0.39	2.52	1.1137	0.2916	0.915	1.09	1.29	0.375
$\boldsymbol{pixelDen}$	0.04	0.14	0.09	0.017	0.07	0.08	0.1	0.03
$\boldsymbol{avgStrokeHt}$	5.16	13.01	8.7283	1.3013	8.02	8.71	9.395	1.375
$\boldsymbol{avgStrokeWd}$	6.41	14.5	9.39	1.27	8.55	9.30	10.08	1.53
$\boldsymbol{lineCount}$	10	30	15.99	3.44	14	15	17	3
$\boldsymbol{wordCount}$	54	237	116.71	25.45	105	111	129	24
$\boldsymbol{avgCCPerLine}$	11.38	60.66	25.46	6.81	21.26	24.92	29.36	8.10
$\boldsymbol{avgCCPerWord}$	1.33	8.12	3.47	0.89	2.88	3.41	3.97	1.09

The quality of the form images are first enhanced using MMC filtering and then these enhanced images are binarized using a ratio based binarization technique. Next, unwanted pixels present in binarized form document images are removed. A page-level skew correction is also applied on noise-free form images. Finally, the part having only handwritten content is cropped out and stored as binarized and gray-scale document images. In addition to these, the statistics related to height, width, aspect ratio, number of words/ lines etc. are computed from each of the document pages. The analysis reveals enough variety within the collected document pages. These collected handwritten document pages would be useful in conducting several experiments such as TL and word extraction, character segmentation, word searching and document categorization that will be described in the following chapters of the thesis.

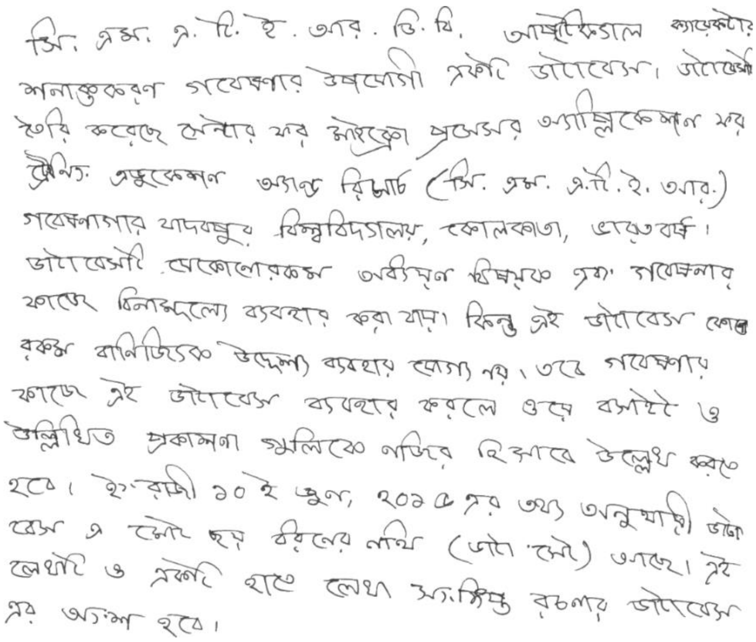
Chapter 3

TEXT LINE EXTRACTION FROM HANDWRITTEN *BANGLA* DOCUMENT IMAGES

3.1 Introduction

Text line (TL) extraction from document images is a process which isolates each of the TLs present in a document. It is one of the important and challenging tasks for textual content analysis in handwritten document images. Error in this stage will be carried forward to next stages of the said problem, which is definitely unwanted. TL identification for handwritten document pages is more difficult than that from the printed ones. Reason behind such comment is due to inherent characteristics of handwritten documents. To be specific, TLs in a handwritten document may be skewed with different angles of inclination with horizontal axis i.e., TLs may not be oriented in same direction (see Fig. 3.1).

Generally, TLs can easily be extracted from the document images by identifying only the valleys of horizontal pixel density histograms (see Fig. 3.2(a)) when skew of TLs are negligible and the document is written with ample inter-line spacing. But this scenario is very rare while considering handwritten document page images. One instance of complex document page with the corresponding horizontal pixel density histogram is shown in Fig. 3.2(b). Sometimes adjacent TLs may touch/overlap with another at single point/multiple points (see Fig. 3.3(a-d)). All these issues make the TL extraction from handwritten document images a challenging research problem.



সি. এম. এ. টি. ই. আর. ডি. বি. আঞ্চলিক ক্যারেক্টার
শনাক্তকরণ গবেষণার উপযোগী একটি ডাটাবেস। ডাটাবেসটি
তৈরি করেছে সেন্টার ফর মাইক্রো প্রসেসর অ্যাপ্লিকেশন ফর
ট্রেনিং এডুকেশন অ্যান্ড রিসার্চ (সি. এম. এ.টি.ই. আর.)
গবেষণাগার যাদবপুর বিশ্ববিদ্যালয়, কোলকাতা, ভারতবর্ষ।
ডাটাবেসটি যেকোনোরকম অবাণিজ্যিক বিষয়ক এবং গবেষণার
কাজে বিনামূল্যে ব্যবহার করা যায়। কিন্তু এই ডাটাবেস কোনো
রকম বাণিজ্যিক উদ্দেশ্য ব্যবহার যোগ্য নয়। তবে গবেষণার
কাজে এই ডাটাবেস ব্যবহার করলে ওয়েবসাইট ও
উল্লিখিত প্রকাশনা গুলিকে নজির হিসাবে উল্লেখ করতে
হবে। ইং রাজী ১০ই জুন, ২০১৫ এর তথ্য অনুযায়ী ডাটা
বেস এ মোট ছয় বিভাগের নথি (ডাটা সেট) আছে। এই
লেখাটি ও একটি হাতে লেখা সংক্ষিপ্ত রচনার ডাটাবেস
এর অংশ হবে।

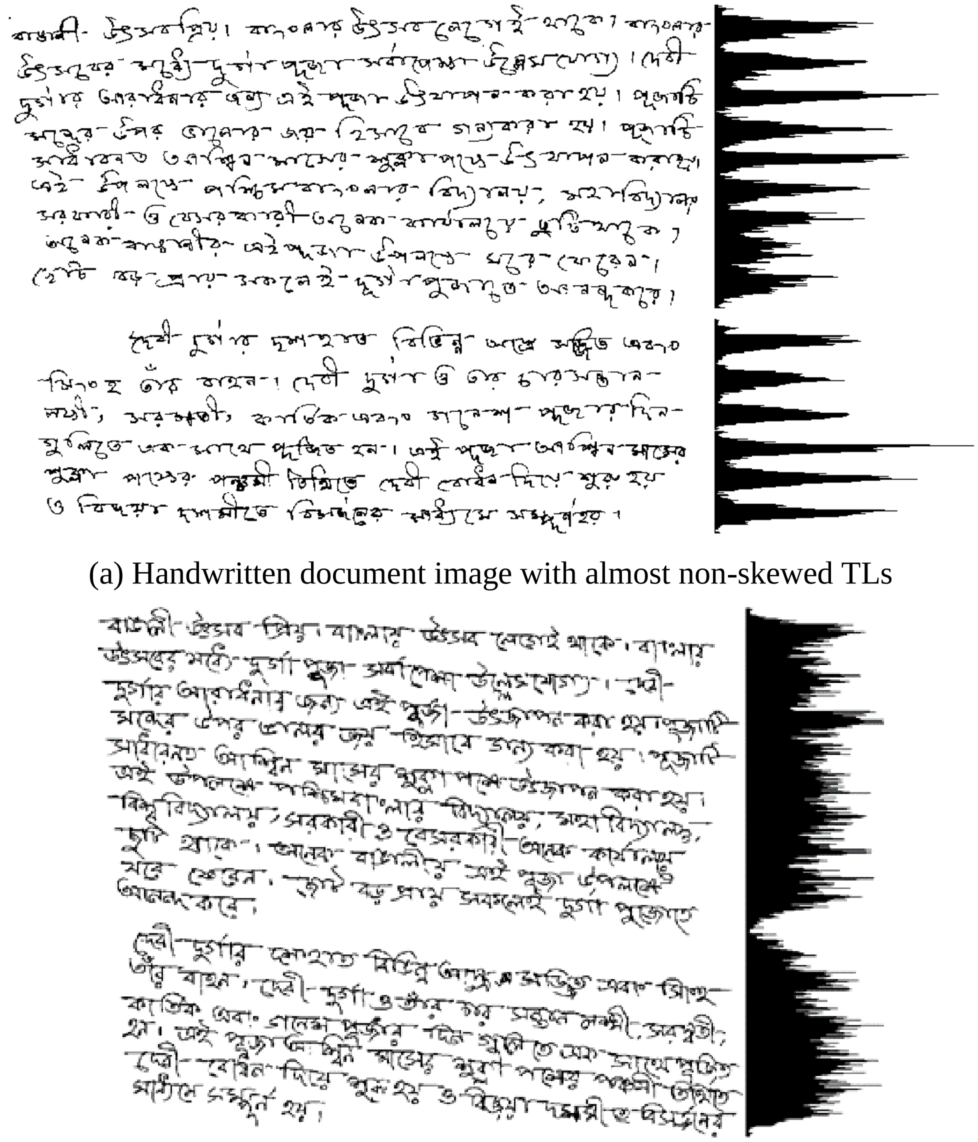
Fig. 3.1 An instance of handwritten document page where TLs are in different directions

3.1.1 Literature Survey

Many research articles on extraction of unconstrained handwritten TLs from digitized document pages are available in the literature [25, 47-51, 57, 78, 123-133]. These works can be classified broadly into 3 different categories, *viz.*, i) CC based approaches [47-49, 123-125], ii) Morphology based approaches [25, 126-132] and iii) Partitioning based approaches [50-51, 57, 78, 133]. The work presented in this thesis is hybridization of these categories.

3.1.1.1 CC based Approaches

In the first category of solutions, CCs of a document image are extracted first and subsequently they are analyzed based on various features of them. Finally, TLs are formed by joining the related CCs depending on some pre-defined hypotheses. An iterative hypothesis validation strategy using Hough Transform (HT) based alignment detection is used in [123]. Here, at each stage, a text-line hypothesis is generated by searching the best alignments of the CCs. An almost similar approach has been used in [124]. It, initially, measures the orientation angle of all CCs, present in a document page, with respect to page boundaries using HT and then uses natural learning method, similar to human learning procedure, to cluster the CCs. CCs belonging to same cluster are then joined to form TLs. A block based HT method is used in [134], which is performed in three steps. In the first step, CCs are extracted from binarized document image. Then, a block-based HT technique is employed for detection of TLs and finally, the TLs that are not separated in previous step are identified and then separated.



(a) Handwritten document image with almost non-skewed TLs

b) Handwritten document image with skewed TLs

Fig. 3.2 Horizontal pixel density histograms for two handwritten document images with varying skewness

The work described in [48] has extracted the CCs first and then analyzed them to form TL. In this work, the CCs are categorized into large and small CCs. Then the small CCs are joined to the larger one to construct the actual TLs. A similar work found in [49] where all the CCs of a document page are classified in three categories depending on average character height and average character width. In the second step, HT mapping is applied on a subset which is expected to contain major part of components that correspond to the characters. Finally, falsely detected TLs are separated by analyzing the skeletons of vertically connected components. In the work [135], first, the candidate CCs are extracted from document image using Maximally Stable Extremal Region (MSER) with the noises filtered by Adaboost and Convolution Neural Network (CNN). Then, the TLs are generated using hierarchical edge reconstruction model and cut by local linearity of TLs in the spanning tree which is formed for the document page image. Finally, for accurate TL extraction, the small components are re-connected based on TL energy minimization in terms of TL consistency and the fitting error. Please note that, handwritten document images contain large number of CCs. Therefore, processing a large number of CCs is time consuming which implies these methods are computationally expensive.

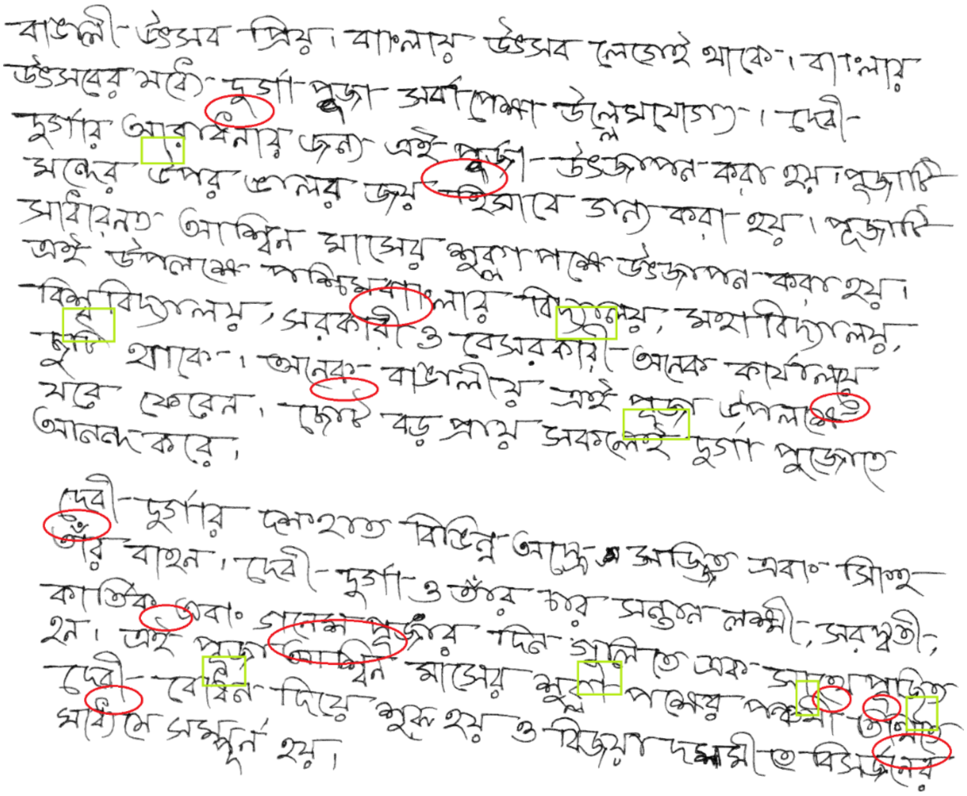


Fig. 3.3 Showing a document page where two TLs are overlapped (enclosed using elliptical shape) and /or touched (enclosed using rectangular shape)

3.1.1.2 Morphology based Approaches

A number of works [25, 126-132] are found in literature that have used morphological operator to obtain TLs from a document page. The authors in the works [126] and [127] have used Run length smoothing algorithm (RLSA) and the fuzzy RLSA respectively to extract TLs from handwritten document page images. An adaptive RLSA [25] evolves from classical RLSA and uses additional smoothing constraints in regard to the geometrical properties of neighboring CCs. The technique, used in [128], is based on morphological operations and RLSA. It segments individual TLs from unconstrained handwritten document images. A minimal spanning tree (MST) based clustering technique [129] with distance metric learning is used for TL segmentation from Chinese documents. Authors in [130] have relied on Mumford-Shah (MS) model to address some solution for the said problem. In this work, TL segmentation is achieved by minimizing the MS energy. In [131], density estimation and level-set methods are used for extraction of handwritten TLs from digitized document pages whereas the work [132] presents a novel technique for segmentation of multi-oriented handwritten TLs using water-flow method. It has used hypothetical water flow at a specific flow angle from both sides of the document image for the same. These methods are good for extracting TLs from handwritten documents while the TLs are not affected by touching/overlapped components and large line-level skewedness. Also, distinguishing the partially wet and completely wet region [132] is an overhead.

3.1.1.3 Partition based Approaches

In this category of works [50-51, 57, 78, 133] first the document page containing unconstrained handwritten samples is partitioned into a number of vertical fragments (VFs). Then the TL extraction hypothesis are applied in each VFs. Finally, the identified TLs belonging to each of the VFs are analyzed to form the final TL. In the work [133], a piece-wise painting algorithm (PPA) that enhances the separability between the foreground and background pixels is employed to smear the foreground portion of the document image and then from this smeared document image the TLs are extracted. The works [50-51] have used piece-wise water-flow technique for the said issue. Here, the basic water-flow technique has been employed in each of the VFs prior to the final detection of TLs present in the underlying document image. Then based on distance metric, all these generated line segments (LSs) in each of the fragments are joined with its nearby LSs form the adjacent VFs to obtain the final TLs. Whereas, the techniques mentioned in [57, 78] have used contour and Spiral Run Length Smearing Algorithm (SRLSA) based approach for finding the LSs in each of the VFs. Next, intra- and inter fragments LSs are joined to generate the final result.

Again, to find the TLs inside a fragment, an iterative approach is applied in [50-51, 133] which is time consuming. To address these issues, initially, a simple and effective line contour based algorithm has been introduced during this thesis work [78]. But the technique sometimes fails to detect TL segments in case of overlapping of the TLs. Therefore, another attempt has been made [57] which provide a partition based TL extraction technique by using SRLSA. It also faces ambiguity while considering inter-/intra- fragments joining since it uses LS boundary to measure several decision parameters. Therefore, a modified partition based TL extraction method is introduced. The proposed method heavily relies on the concept of the works [78] [57]. In addition to this, an automated evaluation technique is proposed to evaluate the present technique. To use this automated evaluation, 300 TL level ground truth (GT) images are prepared.

3.1.2 Objective of the Chapter

In sort the objectives that have been covered in this chapter are as follows:

- Designing of a new TL extraction method.
- Introduction of a new automated evaluation method.
- Preparing GT images that contain ideal TL marking.

3.2 Text Line Extraction Technique

A TL extraction technique from unconstrained handwritten digitized document pages have been designed here. To start with, a handwritten binarized document page (i.e., $\mathcal{P}_b$) is first partitioned vertically into n number of fragments of equal width. Next, all these fragments are passed through SRLSA based smearing technique. In each fragment, LSs are estimated by identifying upper and lower contours of them. After that, the identified TL segments of neighboring fragments are analyzed and merged in order to find out the correct boundary of each TL present in the document page. Fig 3.4 shows a schematic work flow diagram of the developed technique.

3.2.1 Partitioning the Document Page

Let, $y = f(x)$ is a real valued non-linear function defined over the interval $[a, b]$, where, $a, b \in \mathbb{R}$. Let $[a, b]$ is divided into n small partitions as $[a = x_0 < x_1 < x_2 < \cdots < x_n = b]$, where $n \in \mathbb{N}$. As $n \to \infty$, the function $y = f(x)$ would seem like a straight line in each of these partitions $[x_i, x_{i+1}]$, where $i = 0, 1, 2, \ldots, n$. Here, $\mathbb{N}$ and $\mathbb{R}$ are set of natural numbers and real numbers respectively. Moreover, if the function $y = f(x)$ is already a straight line then it will remain as a straight line in each of the said partitions.

Relying on the above concept, $\mathcal{P}_b$ is partitioned into N_F number of VFs since TLs that are contained therein may be skewed or curvy. Such partitioning ensures that the LSs in each of fragment appears to be straight. The number of fragments is predefined and the width of each

VF depends on width of the document page. The fragment width (F_W) in the present work is defined as $F_W = \frac{P_{\mathrm{W}}}{N_F}$.

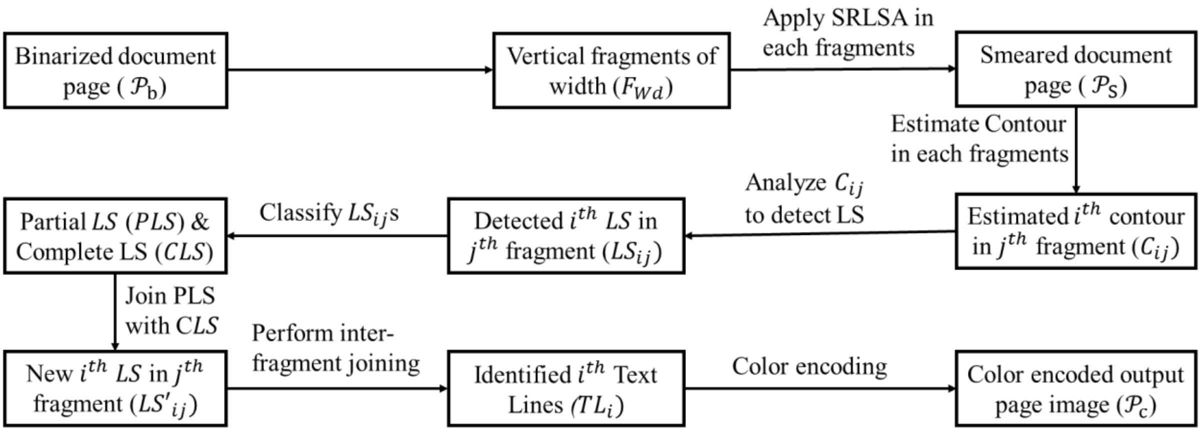

Binarized document page ($\mathcal{P}_{\mathrm{b}}$)
Vertical fragments of width (F_{Wd})
Apply SRLSA in each fragments
Smeared document page ($\mathcal{P}_{\mathrm{S}}$)
Estimate Contour in each fragments
Estimated i^{th} contour in j^{th} fragment (C_{ij})
Analyze C_{ij} to detect LS
Detected i^{th} LS in j^{th} fragment (LS_{ij})
Classify LS_{ij}s
Partial LS (PLS) & Complete LS (CLS)
Join PLS with CLS
New i^{th} LS in j^{th} fragment (LS'_{ij})
Perform inter-fragment joining
Identified i^{th} Text Lines (TL_i)
Color encoding
Color encoded output page image ($\mathcal{P}_c$)

Fig. 3.4 Schematic diagram of TL extraction from a document page

Now, maximum possible value of N_F is page width (say, P_{W}). But such choice of maximum possible value converts a $\mathcal{P}_b$ into vertical line of one pixel width and processing time required for such fragments will be high while reconstructing them into document page (reconstruction is described later in section 3.3.4). In this regard, an investigation has been conducted to find out how many words on an average may be present in a TL of a handwritten document image. The survey has revealed that a TL of an A4-sized handwritten document page contains around 5-9 words on an average. Based on this information, N_F is chosen as 8 in the developed technique. An instance of handwritten document page image with $N_F = 8$ is displayed in Fig. 3.5. However, to validate such choice, experiments have been conducted with varying N_F from 6 to 10. This validation is illustrated in result section.

3.2.2 SRLSA based Smearing

Converting the CCs (see Fig. 3.6) as a virtual CC (VCC) before performing actual TL extraction would be helpful. Frequency of such CCs in unconstrained handwriting is high. Such conversion combines the closely related components into a single component and thereby reduces the processing time. Therefore, such CCs present in a $\mathcal{P}_b$, are converted into possible VCCs using SRLSA. The reason behind choice of SRLSA over commonly used morphological operator RLSA [126] has been explained first and then the procedure has been described.

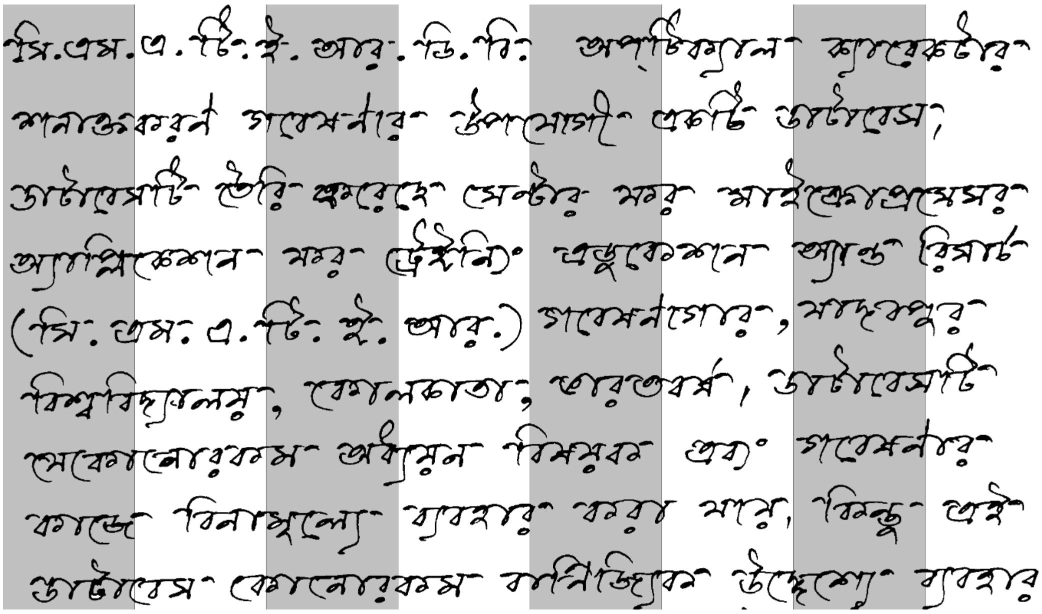


Fig. 3.5 Illustration of VFs considering N_F = 8. Alternative VFs are shaded

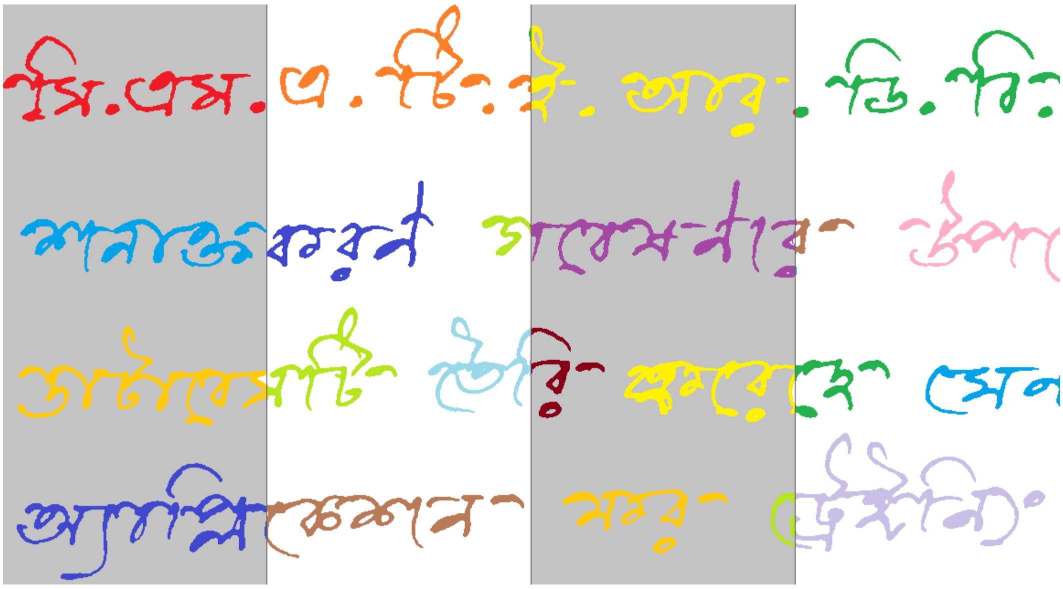


Fig. 3.6 Showing some CCs which need to be converted to VCCs for better result of TL extraction technique. The CCs with same color within a fragment should be converted to VCC

RLSA is a morphological operator which is used for block segmentation and text discrimination [126]. RLSA is usually applied to a sequence of binary values in which object and background pixels are represented as 1's and 0's respectively. The RLSA converts a binary sequence X into an output sequence Y according to the following rules:

- 1's in X are unchanged in Y.

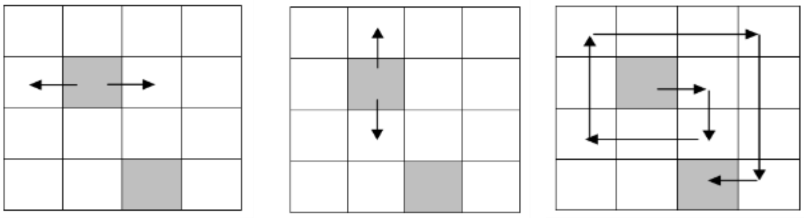
- 0’s in X are changed to 1’s in Y if the count of consecutive 0’s is less than or equal to a predefined limit, say, Th. This process of changing background pixels into foreground ones is often termed as smearing.

For example, with $Th = 4$ the sequence X is mapped into Y as follows (red colored pixels are the changed pixels):

X: 000100000101000010000000011000

Y: 111100000111111111000000011111

When applied to a binarized image, the RLSA has the effect of connecting together the adjacent object pixels that are separated by a distance less than or equal to a predefined threshold value. The technique can separately be applied in both horizontal and vertical directions to generate two different images. Sometimes, the resultant image are produced by logically OR-ing both these images. Irrespective of variations in RLSA, it fails to join two neighboring object pixels when they do not lie along a horizontal or vertical line. This scenario is very frequent while considering handwritten image. Fig. 3.7 depicts one such situation where two object pixels, lying very close to each other, are not smeared by basic horizontal RLSA (HRLSA) (see Fig. 3.7 (a)) or vertical RLSA (VRLSA) (see Fig. 3.7 (b)). To overcome the above limitation of RLSAs, a new smoothing technique, called SRLSA, (see Fig. 3.7 (c)) has been introduced in the work [46]. In this technique, pixels in an image are visited in spiral way to check spatial connectivity of two points in it which is illustrated in Fig. 3.8.



(a) HRLSA (b) VRLSA (c) SRLSA

Fig. 3.7 Smearing of two nearby pixels by different RLSA strategies (gray shade indicates foreground pixels)

Let, A and B be two arbitrary object pixels in an image (I) (see Fig. 3.8). Also let, ε is a relation to indicate spatial connectedness of these two points (i.e., A and B), which means $(A, \mathrm{B}) \in \varepsilon$, iff $\Omega(A, \mathrm{B}) \leq \tau$, where $\Omega(A, \mathrm{B})$ is the spiral distance between the two points A and B in I and τ is the spiral neighborhood distance threshold. The value of $\Omega(A, \mathrm{B})$ can be measured in two ways *viz.*, (i) $\Omega^{+}(A, \mathrm{B})$, the distance measured from B to A using anti-clock wise traversal and (ii) $\Omega^{-}(A, \mathrm{B})$, the distance estimated from A to B by traversing in clock wise manner. In the current work, $\Omega^{-}(A,B)$ has been used for confirming the spatial connectedness of the points A and B.

But, $\Omega^{+}(\mathrm{A},\mathrm{B})$ could be estimated by considering the initial direction of traversal in one of the four directions *namely*, East(E), South(S), West(W) and North(N). In this work, traversal starts towards E direction and then continued following S, W and N directions. The traversal in spiral way is depicted in Fig. 3.8. Then a hypothetical line($\mathcal{L}$) is drawn between A and B to connect them in I using Digital Differential Analyzer (DDA) line drawing algorithm [136]. $\mathcal{L}$ may pass over both object and background pixels. Only the background pixels along $\mathcal{L}$ are considered as smeared pixels for construction (shown in Fig. 3.8).

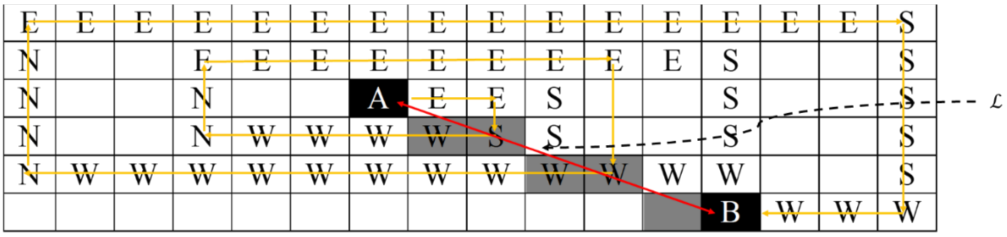
From Fig. 3.8, it is clear that for visiting the pixels spirally, eight different threshold values in terms of pixel count (four for directional movements at initial stage and rest for increment along these four directions in successive stages) could be set. However, for simplicity, here four different threshold values are considered which are

i. T_{HD}: Initial movement in horizontal direction (set as 3 pixels).

ii. T_{VD}: Initial movement in vertical direction (set as 2 pixels).

iii. Inc_{HD}: Increment in horizontal direction in successive traversal (set as 3 pixels).

iv. Inc_{VD}: Increment in vertical direction in successive traversal (set as 2 pixels).



E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	S
N			E	E	E	E	E	E	E	E	E	S			S
N			N			A	E	E	S			S			S
N			N	W	W	W	W	S	S			S			S
N	W	W	W	W	W	W	W	W	W	W	W	W			S
												B	W	W	W

Fig. 3.8 Illustration of SRLSA. Here, black indicates foreground pixels whereas light gray indicates smeared pixels that are obtained by drawing a straight line using DDA between two spatially connected pixels A and B

SRLSA technique, as described above, is applied on each VF of $\mathcal{P}_b$. This process forms a smeared document page (say, $\mathcal{P}_s$) which contains all object pixels of $\mathcal{P}_b$ and newly added smeared pixels (say, #). Therefore $\mathcal{P}_s$ can be defined as $\mathcal{P}_s = \{f(x,y): (x,y) \in [1, H_P] \times [1, W_P] \wedge f(x,y) \in \{0, 255, \#\}\}$. Here, '0', '255' and '#' represent object, background and smeared pixels as shown in Fig. 3.9 as light gray color.

3.2.3 Detection of LSs and Their Contours

Here, the objective is to detect all the LSs in each VF of $\mathcal{P}_s$. Let, i^{th} VF of fragmented $\mathcal{P}_s$ is represented by Γ_i, where $i = 1, 2, \ldots, N_F$ and defined by $\Gamma_i = \{f(x, y): (x, y) \in [1, P_H] \times [1 + (i - 1) \times F_W, i \times F_W] \wedge f(x, y) \in \{0, 1, \#\}\}$. The detection of LSs is carried out by estimating the upper and lower contours of the VCCs present in each VFs. The entire process of selecting LSs is described in Algorithm 3.1.

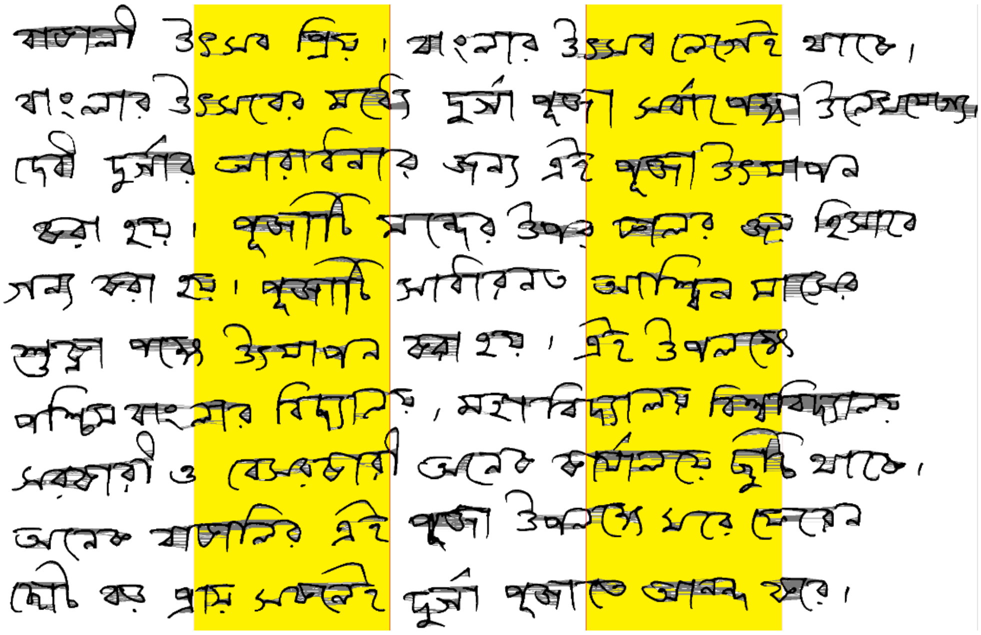


Fig. 3.9 An instance of $\mathcal{P}_s$ which is formed using $N_F = 5$ where light gray pixels are smeared pixels. The alternative VFs are shaded with yellow color

Algorithm 3.1 Detection of LSs in VFs

Input:

Fragmented $\mathcal{P}_s$: $\Gamma_j = \{f(x, y): (x, y) \in [1, P_H] \times [1 + (j - 1) \times F_W, j \times F_W] \wedge f(x, y) \in \{0, 255, \#\}\}$, where $j = 1, 2, \ldots, N_F$.

Output:

(i) i^{th} LS in Γ_j: LS_{ij}, $i = 1, 2, \ldots, N_{LS}^j$ and $j = 1, 2, \ldots, N_F$. Here N_{LS}^j represents number of LS in Γ_j which may differ from one VF to another.

(ii) Upper contour of LS_{ij}: UC_{ij}, where $i = 1, 2, \ldots, N_{LS}^j$ and $j = 1, 2, \ldots, N_F$.

(iii) Lower contour of LS_{ij}: LC_{ij}, where $i = 1, 2, \ldots, N_{LS}^j$ and $j = 1, 2, \ldots, N_F$.

(iv) Average LS's height in Γ_j : μ_{LSH}^j, where $j = 1, 2, \ldots, N_F$.

(v) Average line spacing in Γ_j: μ_{LSS}^j, where $j = 1, 2, \ldots, N_F$.

(vi) Average height of all LS_{ij}: μ_{LSH}

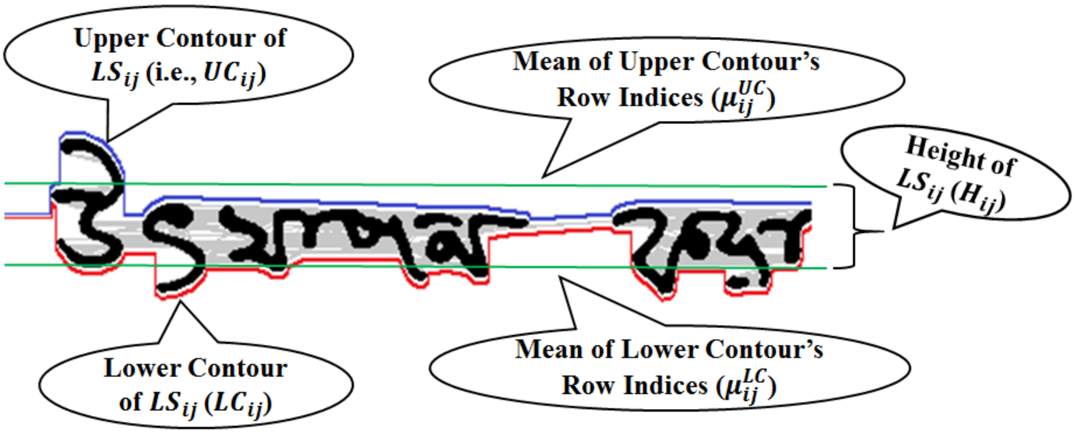
Start

```
 // Estimating LSs in $\Gamma_j$, j=1 to $N_F$
for $j = 1$ to $N_\mathrm{F}$
    Set $N_{LS}^{j} = 0$.
    Set $i = 1$;  // counter to indicate LS number
    Set $k = 1$; // index for spacing between two consecutive LS
    for $R_{No} = 1$ to $P_H$
        1. Scan $\Gamma_j$ in top-down and left-right manner.
        2. Find a row with at least one object pixel (i.e., $f(x,y) = 0$) / smeared pixel
           (i.e., $f(x,y) = \#$) and call it starting row of $LS_{ij}$ (say, $S_R^i$).
        3. Scan from $S_R^i$ to downward and find another row containing only background pixel
           (i.e., $f(x,y) = 255$) and call it as ending row of $LS_{ij}$ (say, $E_R^i$).
        4. Generate the upper contour ($UC_{ij}$) of $LS_{ij}$ by searching the first object/smeared
           pixel from $S_R^i$ till $E_R^i$, ∀column index in $\Gamma_j$. An estimated upper contour of a
           document image is shown in Fig. 3.10.
        5. Estimate the lower contour ($LC_{ij}$) of $LS_{ij}$ by searching for the first object/smeared
           pixel from $E_R^i$ till $S_R^i$ in upward direction, ∀column index in $\Gamma_j$. An estimated lower
           contour of a document image is shown in Fig. 3.10.
        6. Calculate average of row indices of upper contour (say, $\mu_{ij}^{UC}$) and lower contour
           ($\mu_{ij}^{LC}$) as $\mu_{ij}^{UC} = \left[\frac{\sum r_t \in UC_{ij}}{n(UC_{ij})}\right]$ and $\mu_{ij}^{LC} = \left[\frac{\sum r_t \in LC_{ij}}{n(LC_{ij})}\right]$, where $r_t$ indicates the row index
           of $t^{th}$ contour point $\in C$,  $n(C)$ is number of contour points  and the function $[x]$
           indicates nearest integer value of $x \in \mathbb{R}$. The position of upper and lower contour
           and their mean row indices are depicted in Fig. 3.10.
        7. Calculate $LS_{ij}$'s height ($H_{ij}$) in this fragment as $H_{ij} = \mu_{ij}^{LC} - \mu_{ij}^{UC} + 1$. The height
           of an LS belonging to some VF is shown in Fig. 3.10 and Fig. 3.11.
        8. if $i > 1$
               Calculate spacing ($\Delta_{kj}$) between any two consecutive LS's (see Fig. 3.11) as
               $\Delta_{kj} = \mu_{ij}^{UC} - \mu_{(i-1)j}^{LC} + 1$
               $k = k + 1$
           end if
           $R_{No} = E_R^i + 1$
        9. update $i = i + 1$
        10. $R_{No} = E_R^i + 1$
        11. Set $N_{LS}^{j} = N_{LS}^{j} + 1$
    End for $R_{No}$
    Calculate $\mu_{LSH}^{j} = \frac{1}{N_{LS}^{j}}\sum_{i=1}^{N_{LS}^{j}} H_{ij}$ and $\mu_{LS}^{j} = \frac{1}{N_{LS}^{j}-1}\sum_{k=1}^{N_{LS}^{j}-1} \Delta_{kj}$
End for j
Calculate $\mu_{LSH} = \frac{\sum_{x=1}^{N_F} \mu_{LSH}^{x}}{N_F}$
End
```

An instance of document page containing estimated upper and lower contours for each of the LSs is shown in Fig. 3.12. The LSs are generated using $N_\mathrm{F} = 5$.

3.2.4 Formation of Final TL

The above algorithm detects LSs of a TL in the different VFs which are needed to be merged to form appropriate TL boundaries for all the TLs in the document image. Therefore, the final TL formation is conducted in two steps, *namely*, a) merging the intra-fragment LSs and b) joining the inter-fragment LSs.



Upper Contour of LS_{ij} (i.e., UC_{ij})
Mean of Upper Contour's Row Indices (μ_{ij}^{UC})
Height of LS_{ij} (H_{ij})
Lower Contour of LS_{ij} (LC_{ij})
Mean of Lower Contour's Row Indices (μ_{ij}^{LC})

Fig. 3.10 A sample image showing estimated upper contour (UC_{ij}) (in blue color) and lower contour (LC_{ij}) (in red color) of a LS. Average row positions of upper (i.e., μ_{ij}^{UC}) and lower (i.e., μ_{ij}^{LC}) contours are also shown using green color rows. Meaning of the notations are found in Algorithm 3.1

3.2.4.1 Merging the Intra-fragment LSs

Sometimes, it is found that some of the VCCs, which are part of some characters, form individual LS inside a fragment (see Fig. 3.13). Therefore, a technique is required to put them in the proper TL. To do this, the entire LSs are first classified into two categories *viz.*, partial LS (PLS) and complete LS (CLS). Any LS (say, LS_{ij}) is identified as PLS or CLS based on the following rule:

$status(LS_{ij})\begin{cases} = PLS, if\ H_{ij} < \mu_{LSH} \\ = CLS, otherwise \end{cases}$, where H_{ij} is the height of LS_{ij} and μ_{LSH} is the mean height of all LSs in a document image.

Next, based on the position, a PLS is again classified into two categories *viz.*, enclosed PLS (EPLS) and adjacent PLS (APLS). If a PLS is entirely enclosed within the left and right boundaries of the corresponding VF then the PLS is taken as an EPLS, otherwise the PLS is an APLS. All these three categories of the LSs are illustrated in Fig. 3.13(a-b). The categorization process of an LS is described in Algorithm 3.2.

Algorithm 3.2 Classification of LS

Input:

(i) LS_{ij}, where $i = 1,2,\dots,N_{LS}^{j}$ and $j = 1,2,\dots,N_F$. Here, N_{LS}^{j} indicates the number of LSs in j^{th} VF.

(ii) H_{ij}, where $i = 1,2,\dots,N_{LS}^{j}$ and $j = 1,2,\dots,N_F$.

(iii) Average height of LS (i.e., μ_{LSH})

Output:

Categories: CLS, APLS, EPLS

```
Start
Step 1.  Estimate left ($LC_{ij}$) and right ($RC_{ij}$) column indices of $LS_{ij}$ using step 4 of Algorithm
         2.3.
Step 2.  If ($H_{ij} \geq \mu_{LSH}$) then
             Set $status(LC_{ij}) = CLS$
         else
             If ($LC_{ij} > (j-1) \times F_W \wedge RC_{ij} < j \times F_W$)   // $F_W$ is width of a VF
             then
                 Set $status(LC_{ij}) = EPLS$
             else
                 Set $status(LC_{ij}) = APLS$
             End if
         End if
End
```

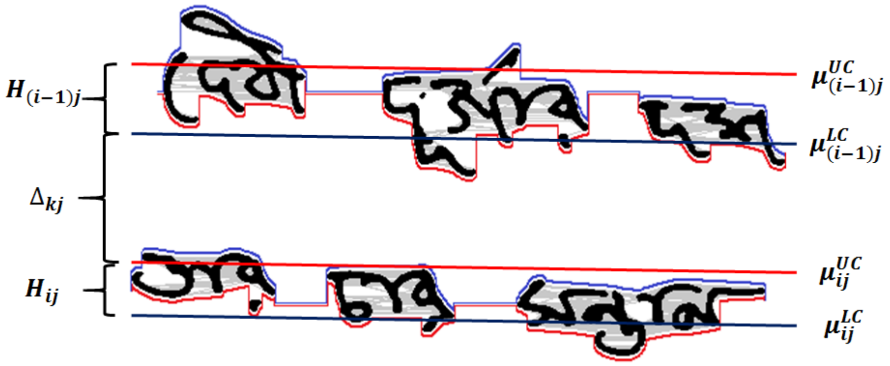


Fig. 3.11 A sample LS showing estimated spacing between two consecutive LS's i.e., Δ_{kj}. Meaning of the notations are found in Algorithm 3.1

From the above classification, it is clear that the component that forms an APLS (see Fig. 3.13(a)) is part of a component which belongs to left or right of the fragment in which the APLS belongs. Therefore, this type of LS are kept as it is. Therefore only the LSs of category $EPLS$ are considered for merging with their appropriate CLS. Let, i^{th} LS in j^{th} fragment (i.e., LS_{ij}) is an EPLS. Now, the appropriate LS (above or below LS_{ij}) to which LS_{ij} to be merged is decided depending upon the value of the decision parameter (say, r) which is calculated as $r = \frac{\Delta_{(i-1)j}}{\Delta_{ij}}$, where Δ_{ij} indicates the inter LS spacing between i^{th} and $(i+1)^{th}$ LSs in a particular fragment j. Now the merging is carried out by the following rule:

```
If ($r \leq 1$) then
        merge $LS_{ij}$ and $LS_{(i-1)j}$
else
        merge $LS_{ij}$ and $LS_{(i+1)j}$
```

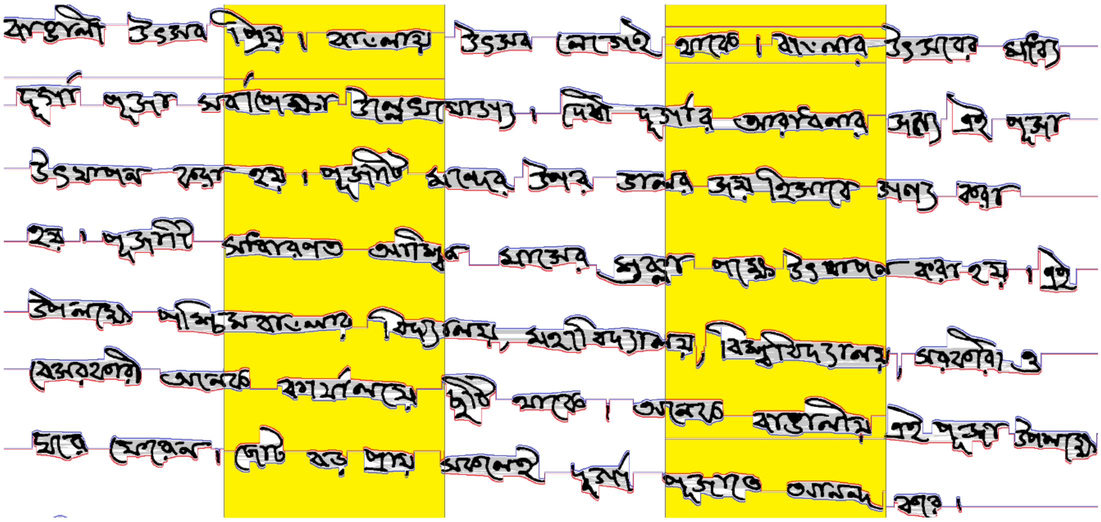
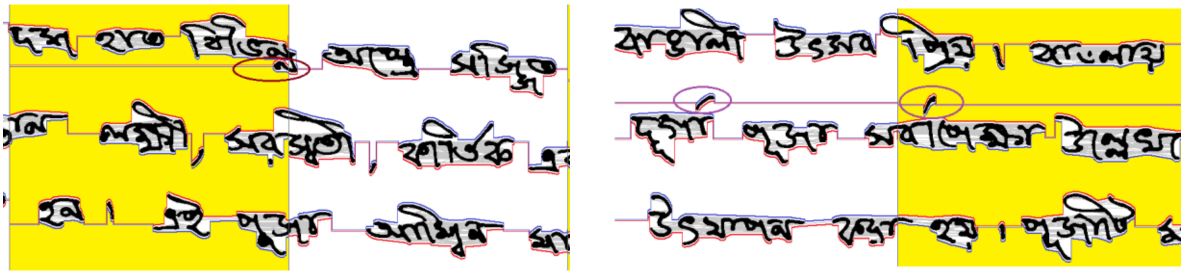


Fig. 3.12 A sample document image containing estimated upper (in blue color) and lower (in red color) contours in each of the VFs. Here $N_{\mathrm{F}} = 5$



(a) APLS

(b) EPLS

Fig. 3.13 Illustration of different types of PLS. (a) and (b) are showing APLS and EPLS respectively. These two categories of LS are marked therein within elliptical boundaries. Rest of the LSs are of CLS category

3.2.4.2 Joining the Inter-fragment LSs

It is already mentioned that the document pages are partitioned into a number of VFs and in each fragment different LSs are there. A sample output image of such partitioning is shown in Fig. 3.5. As this partitioning schema produces different LSs, therefore, it is now required to merge them to form the actual TLs present in the document image under consideration. The technique for joining two LSs of consecutive fragments (say, j^{th} and $(j+1)^{th}$ fragments) for i^{th} LS is described in Algorithm 3.3 which is iterative in nature.

3.3 Automated Evaluation System

An unbiased, automatic and human-error free evaluation protocol is very useful in image processing research. To build with an automated evaluation system, GT images play vital role. Keeping these facts in mind, in the thesis, an automated evaluation system along with GT images of ideally segmented TLs are introduced for assessment any TL extraction technique. In this section first the GT image preparation technique is described and then the evaluation protocol is defined.

Algorithm 3.3 Performing inter-fragment joining

Input:

(i) LS_{ij}, where $i = 1, 2, \dots, N_{LS}^{j}$ and $j = 1, 2, \dots, N_F$.
(ii) μ_{ij}^{UC}, where $i = 1, 2, \dots, N_{LS}^{j}$ and $j = 1, 2, \dots, N_F$. *//μ_{ij}^{UC} indicate mean row of upper contour of LS_{ij}*

Output:

Final document image with extracted line boundaries

```
Start
        for $j = 1$ to $N_F - 1$
          for $i = 1$ to $N_{LS}^{j} - 1$ // $N_{LS}^{j}$ indicates the number of LSs in $j^{th}$ fragment
            set $min_d = 0$.  // Minimum distance
            for $k = 1$ to $N_{LS}^{j+1}$  //  Number of LS in $(j+1)^{th}$ fragment
              Calculate taxicab distance ($d_K$) between $\mu_{ij}^{UC}$ and $\mu_{k(j+1)}^{UC}$
              i.e., $d_k = |\mu_{ij}^{UC} - \mu_{k(j+1)}^{UC}|$
            end for $k$
            Find minimum of all $d_k$ (say, $min_d$) i.e., $min_d = \min_{k=1,2,\dots,N_{LS}^{j+1}} \{d_k\}$
            Store the corresponding LS index $t$ for which $d_k = min_d$
            if ($min_d < \delta$) then  // $\delta$ is a threshold value
              join $LS_{ij}$ and $LS_{k(j+1)}$
            end if
          end for j
        end for i
End
```

3.3.1 Ground Truth Preparation

Availability of GT image makes an image-based database more useful as it helps in assessing one's algorithm. Generation of appropriate GT image is always a challenging and tiresome task, although, fast and accurate evaluation of an algorithm is possible using GT images and associated evaluation mechanism.

GT image of different TLS in a document page image is prepared here in a semi-automatic way. First, the TL boundaries are extracted using the method described in section 3.3 considering $N_F = 8$. Then, all the identified TLs of the document page are uniquely colored using color encoding. But, all the identified TLs are not ideal TLs as the present TL extraction process suffers from the following anomalies:

- A single TL is identified as two or more TLs (see Fig. 3.14(a)) or an isolated character(s) and/or character sub-part(s) extracted as individual TL (see Fig. 3.14 (b)). This scenario leads to over segmentation error.
- Two or more TLs are identified as single TL (see Fig. 3.14(c)) or part(s) of one TL is merged with other TL(s) (see Fig. 3.14(d)) thereby suffering from under segmentation error.

All these possible errors, produced during automated TL extraction process, are then manually corrected using the GTgen Tool [74]. Few examples of auto-generated erroneous samples and their corrected versions are displayed in Fig. 3.14(a-h). These corrected and error-free page samples are considered as final GT images and included in the current database.

3.3.2 Evaluation Methodology

To evaluate a TL extraction algorithm, three statistical measures have been considered *viz.*, recall, precision and F-measure [75]. The statistics True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN) are the pre-requisites for calculation of the said statistics. These statistics, excluding TN, are generated by incorporating position wise one-to-one mapping between the ideal (here GT image) and input image (here document image with extracted TLs). The evaluation of all the said statistics is based on count of object pixels. Please note that four major hypotheses are considered for evaluation of character segmentation process. These along with their respective contribution(s) in evaluation process are listed below.

Hypothesis 1: A TL is extracted ideally (see Fig. 3.15(a)). All the object pixels in this case are considered here as TP pixels.

Hypothesis 2: A TL is segmented into multiple TLs (see Fig. 3.15(b)). Object pixels of the largest TL (considering object pixel count) is considered as TP pixels and rest are as FN pixels.

Hypothesis 3: Multiple TLs stay wrongly connected after segmentation (see Fig. 3.15(c)). Object pixels of the largest TL are counted as TP pixels and the rest object pixels are considered as FP pixels.

Hypothesis 4: Part(s) of a TL is (are) attached with other TL (see Fig. 3.15(d)). Object pixels which appear wrongly with other TL(s) hold dual property. These pixels are FN pixels with respect to one TL or can be considered as FP pixels with respect to the other adjacent TL and vice versa. To get rid of such ambiguity these pixels are considered as FP pixels.

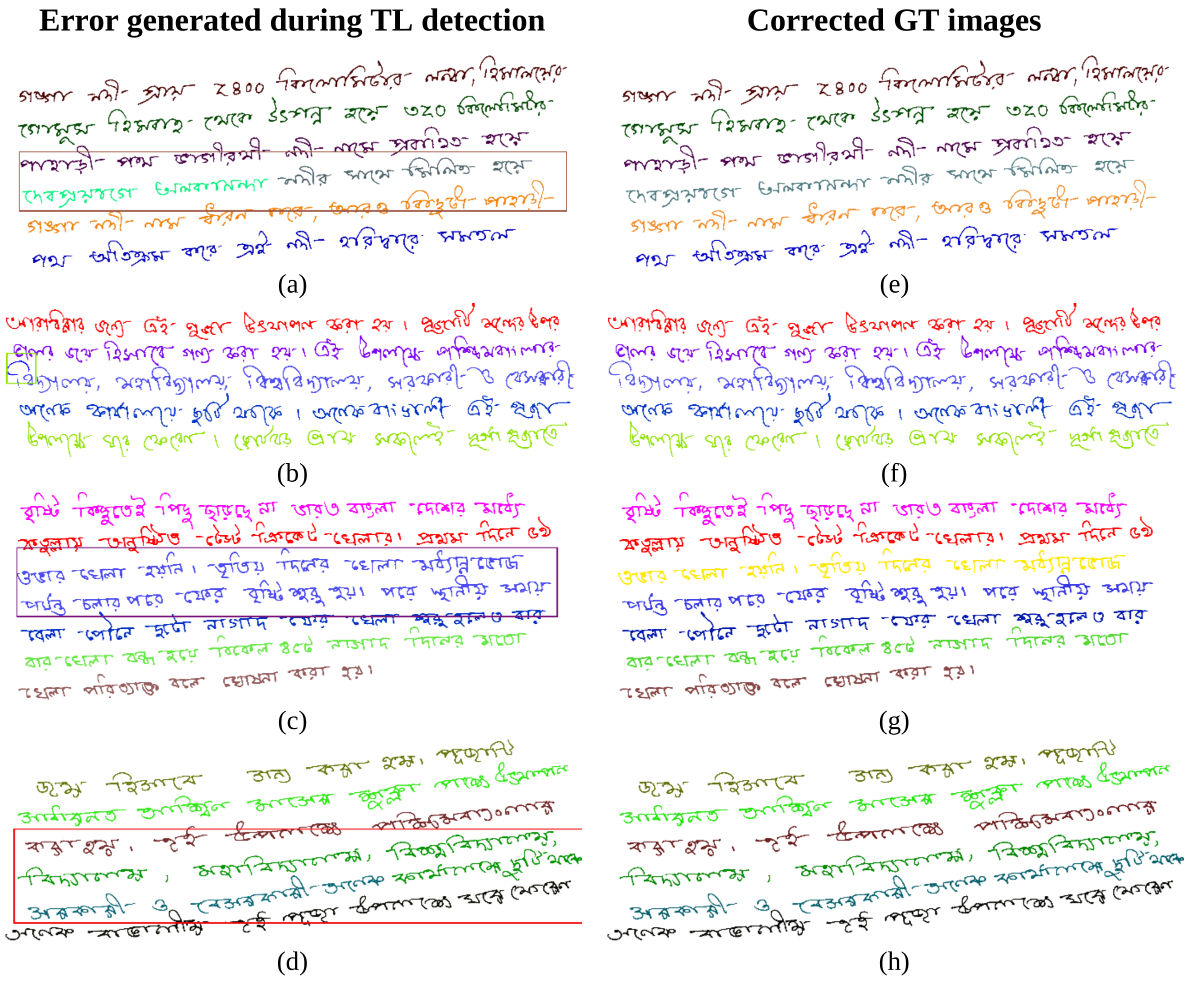
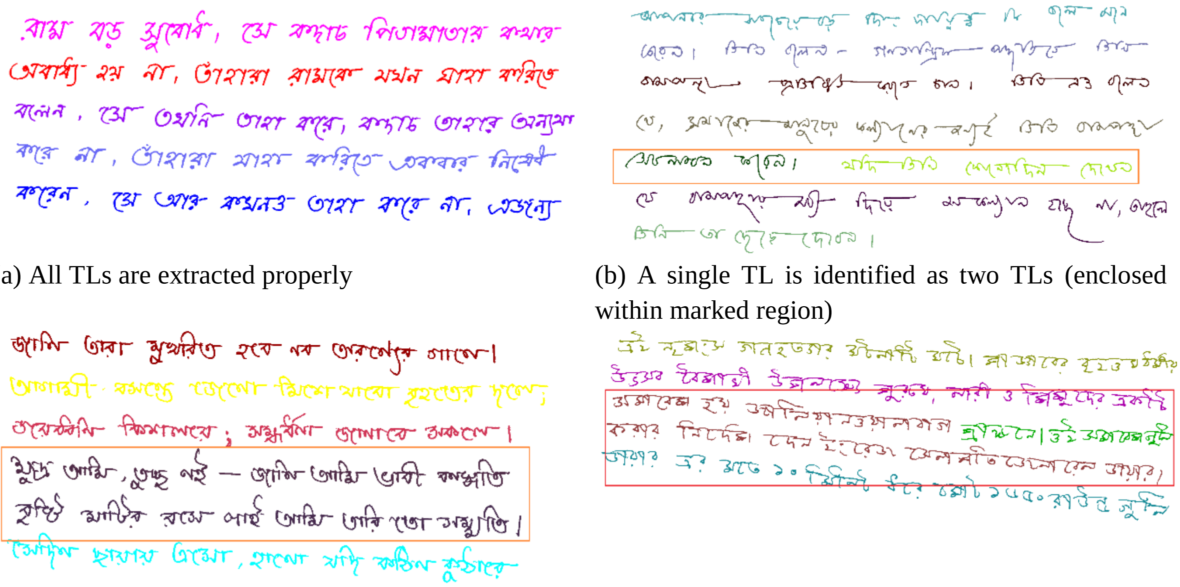


Fig. 3.14 Depiction of different types of errors that need manual correction during GT preparation and (e-h) manually corrected GT images

Please note that the same evaluation algorithm will be used in the later chapters (for automatic evaluation of word extraction (Chapter 4) and character segmentation (Chapter 5)). Therefore, some generic terminologies are used here for describing the said evaluation method, which are defined as follows.

Let, $F = \{f(x,y): (x,y) \in [1,Ht] \times [1,Wd]\}$ is an image. Here Ht and Wd are height and width of F respectively. Also let, $G = \{g(x,y): (x,y) \in [1,Ht] \times [1,Wd]\})$ and $R = \{r(x,y): (x,y) \in [1,Ht] \times [1,Wd]\}$ represent the corresponding GT image and resultant image respectively. G_C and R_C represent the set of components in G and R respectively. $X, Y\ and\ Z$ are the sets of TP, FN and FP pixels of F which are determined using Algorithm 3.4. The entire process is also described pictorially in Fig. 3.16 (a-f).



(a) All TLs are extracted properly

(b) A single TL is identified as two TLs (enclosed within marked region)

(c) Two TLs are identified as a single TL (enclosed within marked region)

(d) Part of a TL is merged with other TL to form a single TL (enclosed within marked region)

Fig. 3.15 Pictorial examples for the said hypotheses: with (a): Hypothesis 1, (b): Hypothesis 2, (c): Hypothesis 3 and (d): Hypothesis 4

Algorithm 3.4: Estimation of Recall, Precision and F-measure

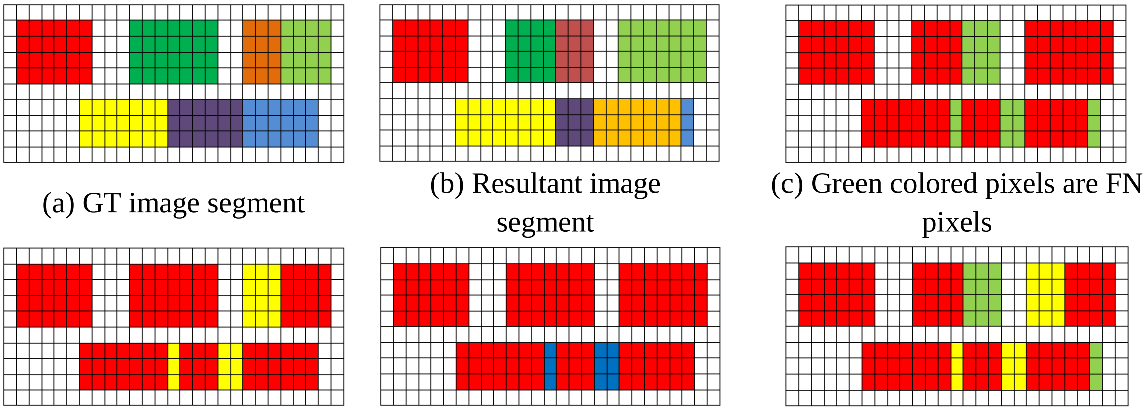
Input: $F, G\ and\ S$
Output: $X, Y\ and\ Z$

```
Start
Step 1.   Initialize $Y = \{\}$ and $Z = \{\}$ and $X = \{(x,y)|f(x,y) = '1' \wedge (x,y) \in [1,Ht] \times [1,Wd]\}$ i.e., X contains all object pixel co-ordinates of $F$.
Step 2.   // Initial selection of FN pixels
          $for\ i = 1,2,\dots,|G_C|$
              $S_i = \{\}$
              $S_i = \{j|(x,y) \in (G_C^i \cap X) \wedge (x,y) \in (R_C^j \cap X)\}$, for some $j = 1,2,\dots,|R_C|$. Here $G_C^i$ and $R_C^j$ represent $i^{th}$and $j^{th}$ component of G and R respectively.
              $if\ |S_i| > 1\ then$
                  Build $Q_k$ and $Q_k^{max}$, $k = 1,2,\dots,|S_i|$ as
```

```
                $Q_k = \{(x,y)|(x,y) \in R_C^j \wedge j \in S_i\}$ and $Q_k^{max} = \max_{k=1,2,\dots,|P_i|}\{|Q_k|\}$
            $end\ if$
        $end\ for$
        $Y = Y \cup \{(x,y)|(x,y) \in Q_k - Q_k^{max}\}$
Step 3. // Initial selection of FP pixels
        for $i = 1,2,\dots,|R_C|$
            $S_i = \{\}$.
            $S_i = \{j|(x,y) \in (R_C^i \cap X) \wedge (x,y) \in (G_C^j \cap X)\}$, for some $j = 1,2,\dots,|G_C|$.
            $if\ |S_i| > 1\ then$
                Build $Q_k, k = 1,2,\dots,|S_i|$ and $Q_k^{max}$ by
                $Q_k = \{(x,y): (x,y) \in G_C^j \wedge j \in S_i\}$ and $Q_k^{max} = \max_{k=1,2,\dots,|S_i|}\{|Q_k|\}$
            $end\ if$
        end for
        $Z = Z \cup \{\{x,y\}: (x,y) \in Q_k - Q_k^{max}\}$
Step 4. $X = X\backslash(Y \cup Z)$
Step 5. $Y = Y\backslash(Y \cap Z)$ // Considering Hypothesis 4
Step 6. Estimation of parameters $TPR, FNR, FPR,$ Recall, Precision and F-measure are
        defined by (note that $|D| = \sum|G_{TL}|$, i.e., number of object pixels in GT image)
        $TP = \frac{|X|}{|D|}$
        $FN = \frac{|Y|}{|D|}$
        $TN = \frac{|Z|}{|D|}$
        $Recall = \frac{|TP|}{|TP|+|FN|}$
        $Precision = \frac{|TP|}{|TP|+|FP|}$
        $F - measure = \frac{2 \times recall \times precision}{recall + precision}$
End
```

3.4 Experimental Results

For experimentation purpose, 300 handwritten document page images are used. The preparation of binarized form of these handwritten document pages is described in chapter 2. The present TL extraction method (described in section 3.3) is applied on them. Before generating result on the entire dataset, the optimized value of N_F is decided. For this, experimentations are performed with values of N_F as 6, 7, 8, 9 and 10 on 50 handwritten pages of Set-F of the present database. For evaluating the performances of the TL extraction algorithm for different values of N_F, the evaluation method along with the performances metrices, described in section 3.4.2, are used. As per that evaluation protocol, each TL in GT image or resultant image is considered as component. TPR, FNR, FPR, Recall, Precision and F-measure scores for the best case are highlighted. The results are shown in Table 3.1.



(a) GT image segment

(b) Resultant image segment

(c) Green colored pixels are FN pixels

(d) Yellow colored pixels are FP pixels

(e) Blue colored pixels are FN as well as FP pixels

(f) Red, Green and Yellow colored pixels are TP, FN and FP pixels resepectively

Fig. 3.16 Selection of TP, FN and FP pixels considering hypothetical image segments

Table 3.1 Average performances of TP, FN, FP, Recall, Precision and F-measure score on entire database with varying N_F values

N_F	**TP**	**FN**	**FP**	**Recall**	**Precision**	**F-measure**
6	0.9098	0.0671	0.0231	0.9719	0.9298	0.9460
7	0.9086	0.0680	0.0235	0.9717	0.9290	0.9455
8	**0.9100**	**0.0653**	0.0246	0.9711	**0.9316**	**0.9476**
9	0.9068	0.0704	**0.0228**	**0.9720**	0.9262	0.9445
10	0.9071	0.0685	0.0244	0.9709	0.9283	0.9454

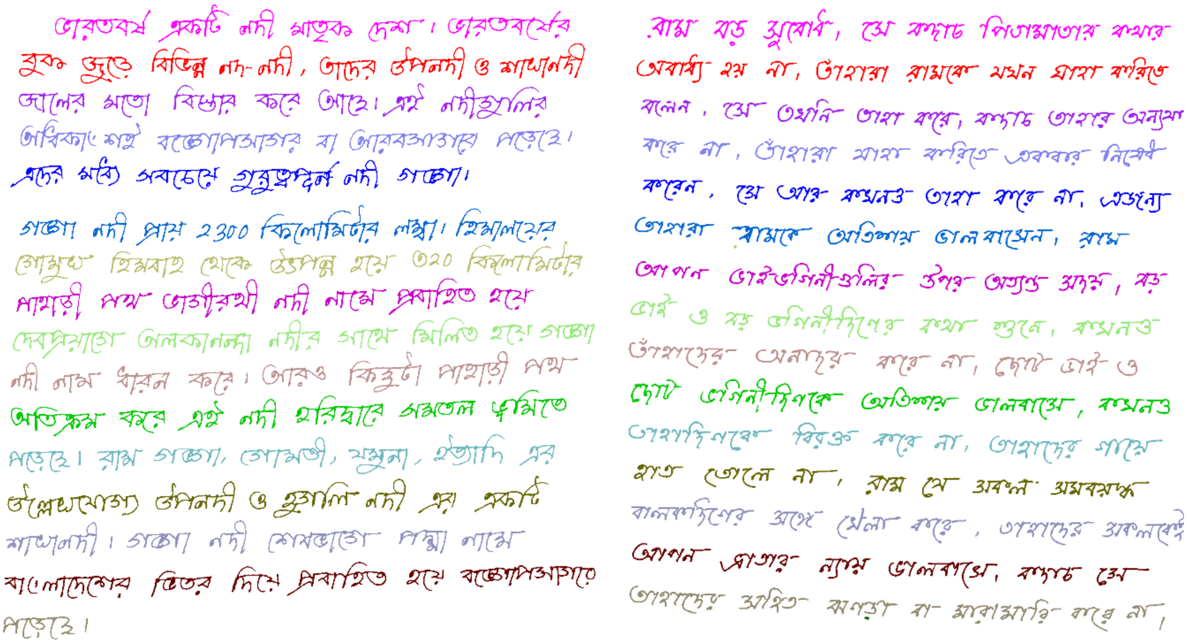
By investigating the results shown in Table 3.1, the following conclusions might be drawn.

- Experimentation with $N_F = 9$ produces best FP value (i.e., minimum FP) and recall. This result indicates that it generates less number of under segmented TLs, but generates more over segmented TLs while reducing the under segmented TLs.
- Experimentation with $N_F = 8$ provides minimum TP and FN. It also provides maximum precision and F-measure score. Better precision value indicates that it tries to optimize among truly extracted TLs and under segmented TLs. Finally, the better F-measure ensures the better trade-off among over, under and truly extracted TLs.

Hence, the best performing experiment is decided by average F-measure score which is the harmonic mean of recall and precision. The experimentation with $N_F = 8$ has provided best result on these 50 handwritten page samples. Therefore, this framework is applied on the entire database for the TL extraction purpose. Two output instances are shown in Fig. 3.17 where all the TLs, present in the document page, are extracted properly. All the above mentioned measures are also calculated on entire dataset. Such measures (page-wise) are plotted using line chart (see Fig. 3.18 (a-f)). To investigate the obtained result on entire dataset, statistics like maximum, minimum, mean, standard deviation (SD), first quartile (Q1), second quartile (Q2) i.e., median, third quartile (Q3) and inter quartile range (IQR) are introduced. These information are provided in Table 3.2.

Table 3.2 Provides experimental results of current TL extraction technique in terms different statistical measures

Term	TP	FN	FP	Recall	Precision	F-measure
Minimum	0.6786	0.0000	0.0000	0.7556	0.7014	0.8085
Maximum	1.0000	0.2323	0.2981	1.0000	1.0000	1.0000
Average	0.9322	0.0178	0.0500	0.9808	0.9487	0.9633
SD	0.0743	0.0319	0.0627	0.0345	0.0645	0.0421
Q1	0.9936	0.0196	0.0816	0.9978	0.9998	0.9969
Q2	0.9638	0.0053	0.0184	0.9945	0.9815	0.9816
Q3	0.8940	0.0021	0.0001	0.9794	0.9166	0.9441
IQR	0.0997	0.0175	0.0815	0.0185	0.0837	0.0528



ভারতবর্ষ একটি নদী মাতৃক দেশ। ভারতবর্ষের
বুক জুড়ে বিভিন্ন নদ-নদী, তাদের উপনদী ও শাখানদী
জালের মতো বিস্তার করে আছে। এই নদীগুলির
অধিকাংশই বঙ্গোপসাগর বা আরবসাগরে পড়েছে।
এদের মধ্যে সবচেয়ে গুরুত্বপূর্ণ নদী গঙ্গা।

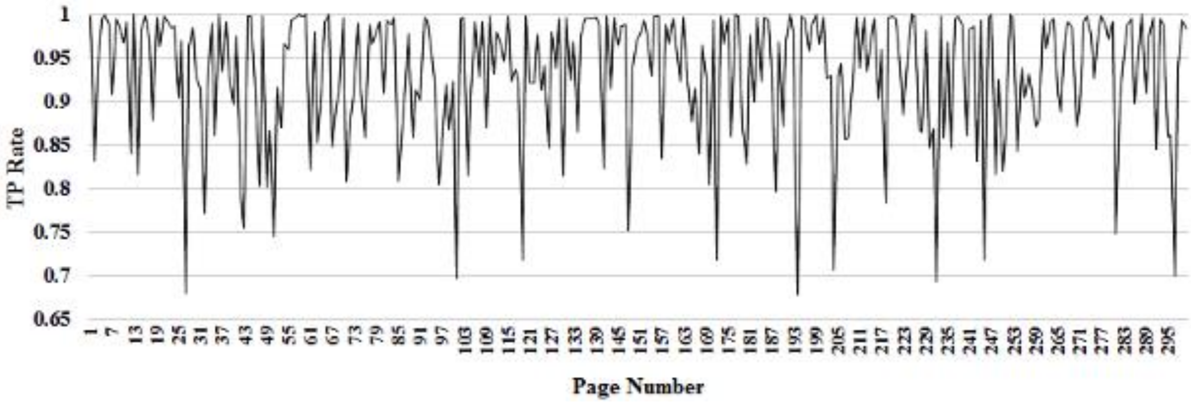
গঙ্গা নদী প্রায় 2300 কিলোমিটার লম্বা। হিমালয়ের
গোমুখ হিমবাহ থেকে উৎপন্ন হয়ে ৩20 কিলোমিটার
পাহাড়ী পথ ভাগীরথী নদী নামে প্রবাহিত হয়ে
দেবপ্রয়াগে অলকানন্দা নদীর সাথে মিলিত হয়ে গঙ্গা
নদী নাম ধারন করে। আরও কিছুটা পাহাড়ী পথ
অতিক্রম করে এই নদী হরিদ্বারে সমতল ভূমিতে
পড়েছে। রাম গঙ্গা, গোমতী, যমুনা, ইত্যাদি এর
উল্লেখযোগ্য উপনদী ও হুগলি নদী এর একটি
শাখানদী। গঙ্গা নদী শেষভাগে পদ্মা নামে
বাংলাদেশের ভিতর দিয়ে প্রবাহিত হয়ে বঙ্গোপসাগরে
পড়েছে।

(a)

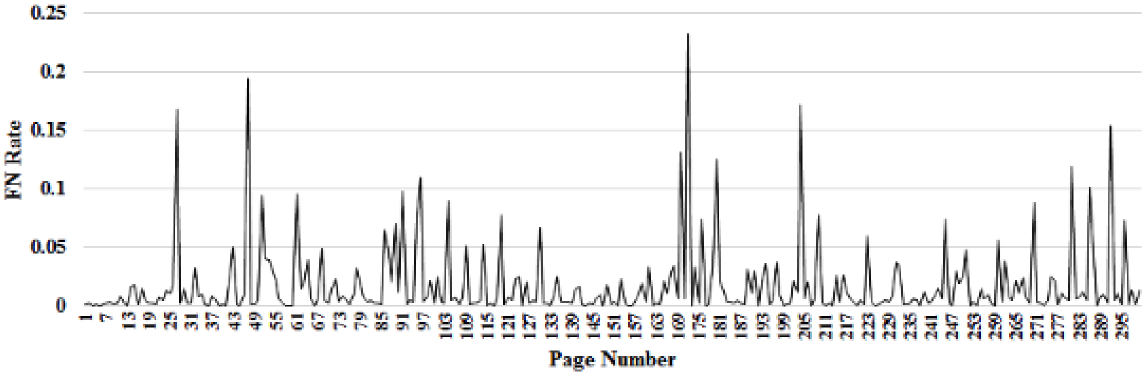
রাম বড় সুবোধ, সে কদাচ পিতামাতার কথার
অবাধ্য হয় না, তাঁহারা রামকে যখন যাহা করিতে
বলেন, সে তখনি তাহা করে, কদাচ তাহার অন্যথা
করে না, তাঁহারা যাহা করিতে বারবার নিষেধ
করেন, সে আর কখনও তাহা করে না, এজন্য
তাঁহারা রামকে অতিশয় ভালবাসেন, রাম
আপন ভাইভগিনীগুলির উপর অত্যন্ত সদয়, বড়
ভাই ও বড় ভগিনীদিগের কথা শুনে, কখনও
তাঁহাদের অনাদর করে না, ছোট ভাই ও
ছোট ভগিনীদিগকে অতিশয় ভালবাসে, কখনও
তাহাদিগকে বিরক্ত করে না, তাহাদের গায়ে
হাত তোলে না, রাম যে সকল সমবয়স্ক
বালকদিগের সঙ্গে খেলা করে, তাহাদের সকলকেই
আপন ভ্রাতার ন্যায় ভালবাসে, কদাচ সে
তাহাদের সহিত ঝগড়া বা মারামারি করে না,

(b)

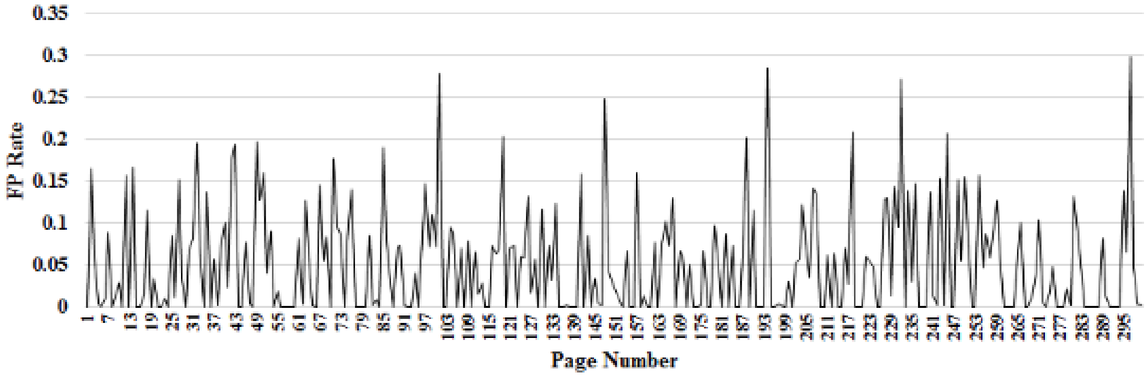
Fig. 3.17 Two instances of correctly extracted TLs. Page image of (a) Set-D and (b) Set-F



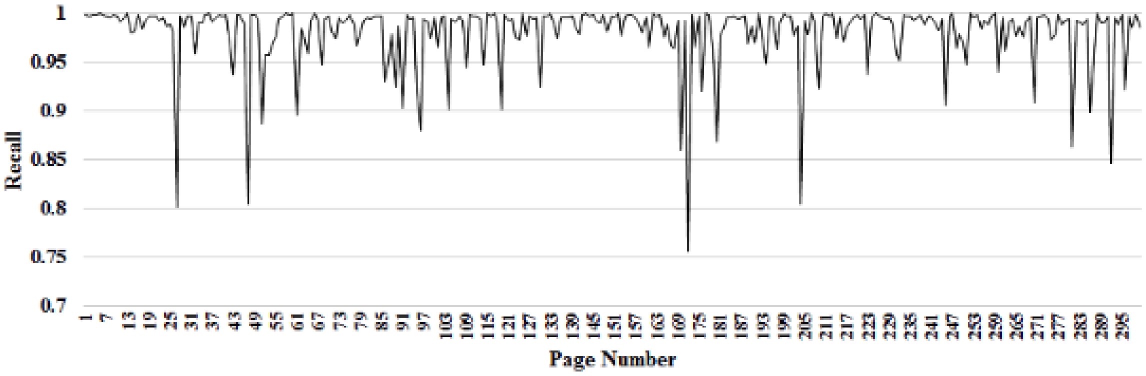
(a) Page-wise TP



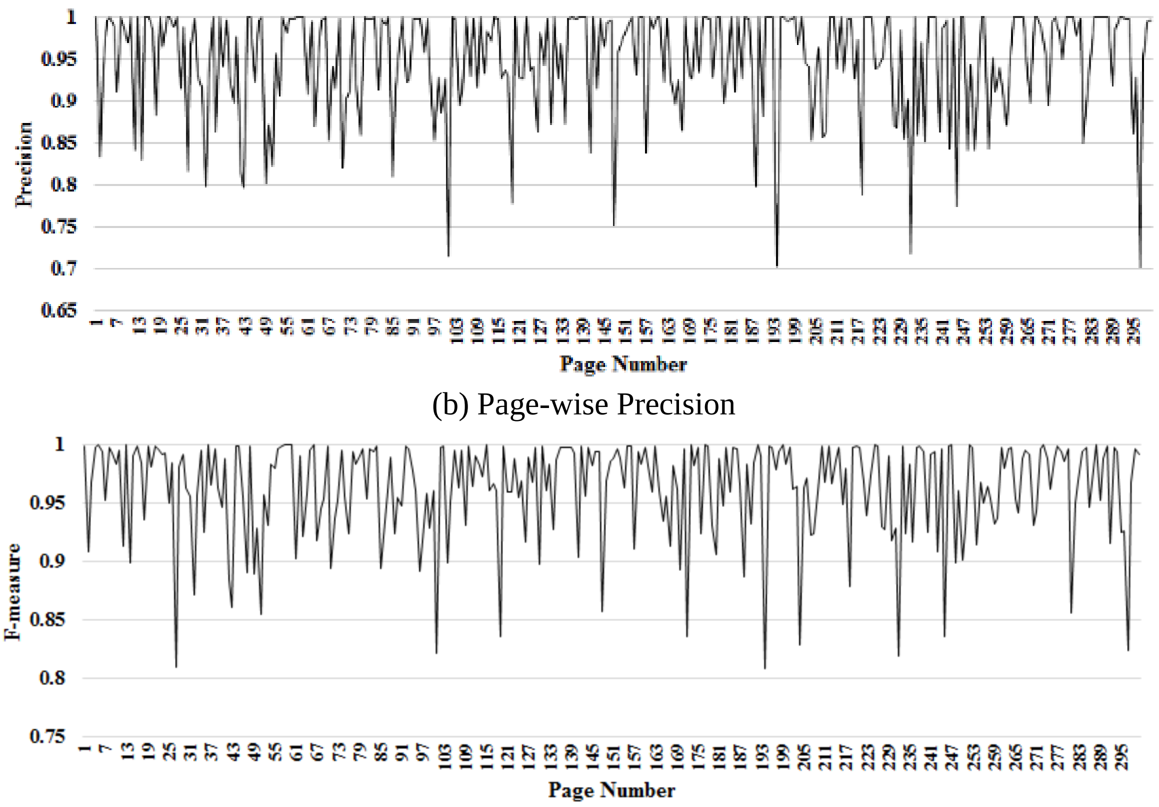
(b) Page-wise FN



(c) Page-wise FP



(d) Page-wise Recall



(b) Page-wise Precision

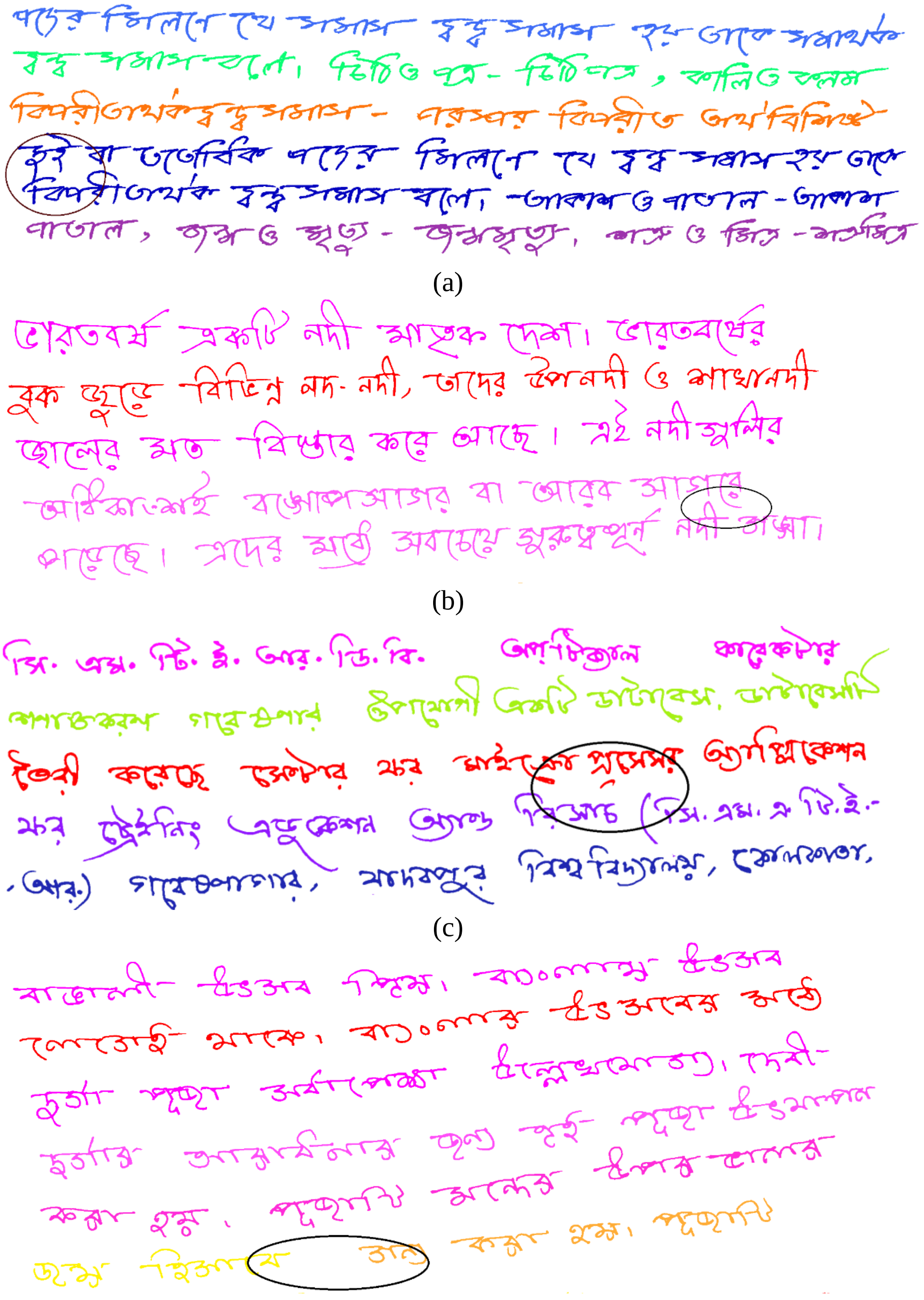
(b) Page-wise F-measure

Fig. 3.18 Page-wise performance measures for all the page samples.

3.4.1 Error Case Analysis

The results presented in the Table 3.2 also show that most of the error cases produced by the present technique are due to under segmentation of the TLs. Some of the erroneous results are shown in Fig. 3.19(a-f) where the erroneous region(s) is (are) enclosed by elliptical shape. Instances of under segmented TLs are shown in Fig. 3.19(a-c) while Fig. 3.19(d) displays a case of over segmentation. From the result observed in the Fig. 3.19(a), it can be said that present technique fails due to the presence of the touching components between the TLs under consideration while Fig. 3.19(b) reveals the presence of closely overlapping TLs. The closely overlapped components of these TLs have been converted to single VCC while applying SRLSA. Fig. 3.19(c) represents a situation where a part of a modified character of a TL gets connected to component of other TL and causing an under segmentation error. This fact happens while performing intra-fragment joining.

On the other hand, over segmentation has occurred mainly due to the choice of threshold for merging successive LSs in neighboring fragments as described in Algorithm 3.3 (see Fig. 3.19(c)). Some errors also have occurred because of some part(s) of a TL is wrongly grouped with another TL and the rest of the parts of it remain isolated to form another TL (see Fig. 3.19 (e-f)). These are fuzzy errors as considered in Hypothesis 4. Analyzing these images it can be said that such errors are happened due to presence of touching and/or overlapped components between the TLs.



পদের মিলনে যে সমাস দ্বন্দ্ব সমাস হয় তাকে সমার্থক
দ্বন্দ্ব সমাস বলে। চিঠি ও পত্র - চিঠিপত্র, কালি ও কলম
বিপরীতার্থক দ্বন্দ্ব সমাস - পরস্পর বিপরীত অর্থবিশিষ্ট
দুই বা ততোধিক পদের মিলনে যে দ্বন্দ্ব সমাস হয় তাকে
বিপরীতার্থক দ্বন্দ্ব সমাস বলে, আকাশ ও পাতাল - আকাশ
পাতাল, জন্ম ও মৃত্যু - জন্মমৃত্যু, শত্রু ও মিত্র - শত্রুমিত্র

(a)

ভারতবর্ষ একটি নদী মাতৃক দেশ। ভারতবর্ষের
বুক জুড়ে বিভিন্ন নদ-নদী, তাদের উপনদী ও শাখানদী
জালের মত বিস্তার করে আছে। এই নদীগুলির
অধিকাংশই বঙ্গোপসাগর বা আরব সাগরে
পড়েছে। এদের মধ্যে সবচেয়ে গুরুত্বপূর্ণ নদী গঙ্গা।

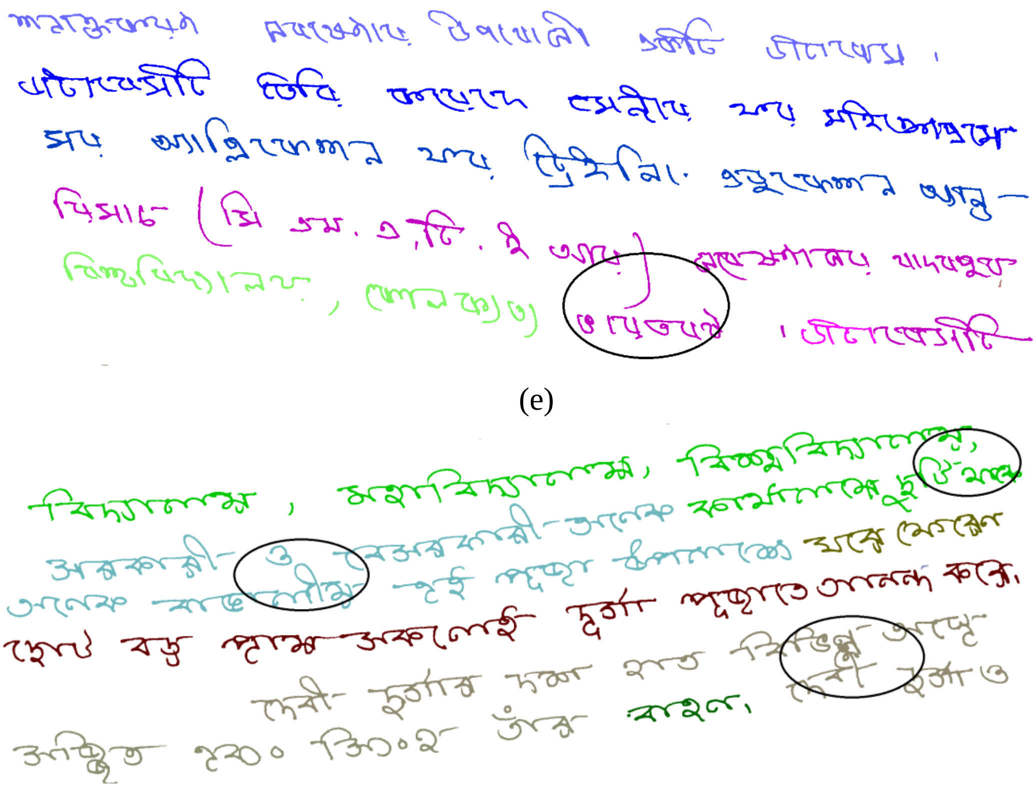
(b)

সি. এম. টি. ই. আর. ডি. বি. অপটিক্যাল ক্যারেক্টার
রেকগনিশন গবেষণার উপযোগী একটি ডাটাবেস. ডাটাবেসটি
তৈরী করেছে সেন্টার ফর মাইক্রোপ্রসেসর অ্যাপ্লিকেশন
ফর ট্রেনিং এডুকেশন অ্যান্ড রিসার্চ (সি.এম.এ.টি.ই.-
আর.) গবেষণাগার, যাদবপুর বিশ্ববিদ্যালয়, কোলকাতা.

(c)

বাঙালী উৎসব প্রিয়। বাংলায় উৎসব
লেগেই থাকে। বাংলার উৎসবের মধ্যে
দুর্গা পূজা সর্বশ্রেষ্ঠ উল্লেখযোগ্য। দেবী-
দুর্গার আরাধনার জন্য এই পূজা উদযাপন
করা হয়। পূজাটি অনেক উপাচারের
জন্য বিখ্যাত অত্র করা হয়। পূজাটি

(d)



(e)

(f)

Fig. 3.19 Example of error cases

3.5 Discussion

Extraction of TLs from the handwritten document images is one of the major challenges in an OCR system. Presence of skewed and/or touching TLs, which are obvious in handwritten documents, makes TL extraction job a difficult one to the researchers. In this context, the present chapter has described the development of a partitioning based TL extraction technique where first SRLSA is used to form VCC and then contour based method has been adopted. For experimentation, GT images containing ideal TLs is prepared. It also includes automated evaluation procedure for assessment of TL extraction method and the evaluation process is performed on 300 document images. The technique produces a reasonably good result.

Chapter 4

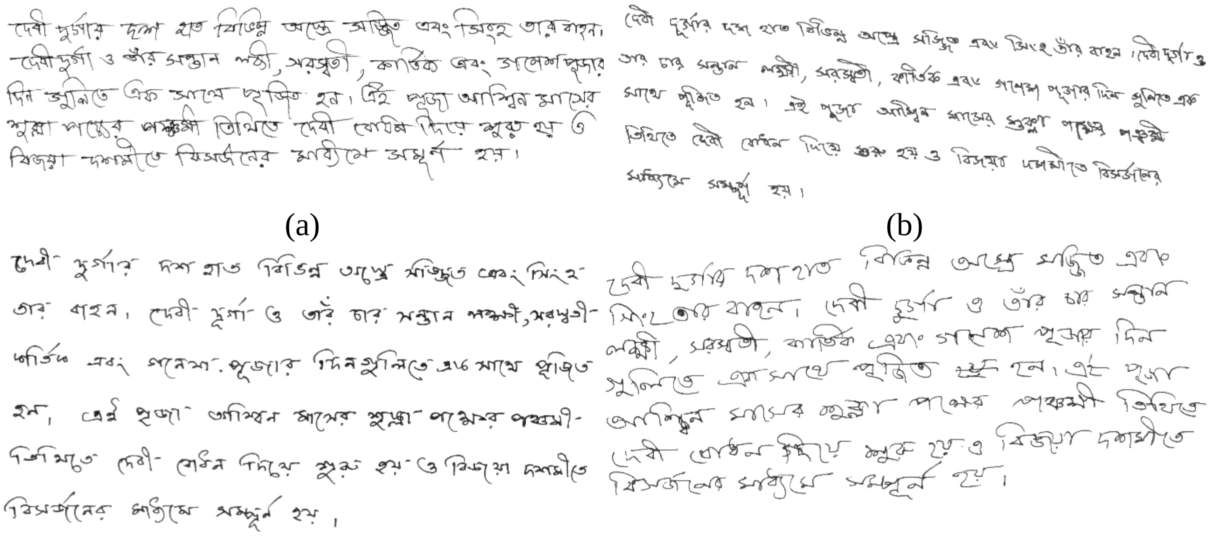
WORD EXTRACTION FROM HANDWRITTEN *BANGLA* DOCUMENT IMAGES

4.1 Introduction

The main objective of any document image segmentation technique is splitting a document image into smaller parts which are called segments. These segments can be produced depending on the requirement and segmentation technique can be applied at different levels of the document image. Each of these segments has its own importance. The segments can be labeled as word, text line (TL), phrase or any information unit depending on the purpose of the document analysis. Word extraction is a document image segmentation process which identifies all the words present therein. It is widely used in any document processing methods since word is considered as the primitive segment for most of document image analysis tasks.

Word extraction from unconstrained handwritten document images is more difficult in comparison to the printed ones. In a printed document, in general, all the characters have definite shapes and sizes. Also intra-word and inter-word gaps are mostly uniform in printed documents since these documents are typed through the keyboard or typewriter. For example, Fig. 2.2 (a, d) and Fig. 2.3(a-b) (refer to Chapter 2) illustrate the said facts. These issues make the process of word extraction from printed document much easier that its counterpart.

Apart from the above mentioned facts, the problems related to unconstrained handwritten document images are much more complicated due to the wide variations in handwriting patterns of the individuals (refer to Fig. 4.1(a-d)). Writing samples shown in these figures have same content written by four different writers which clearly specify that writing patterns vary from writer to writer. In addition to this, writing patterns differ for same writer from time to time. Such variations in writing styles add more complexity to the process of word extraction from handwritten documents.



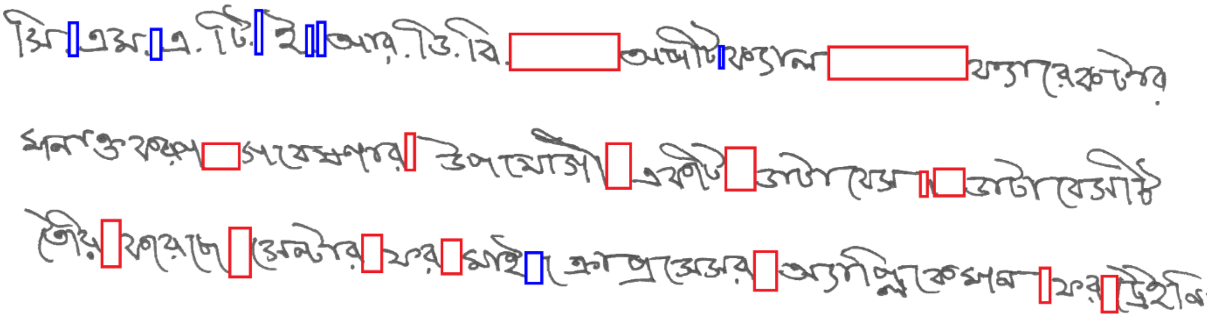
(a) (b)

(c) (d)

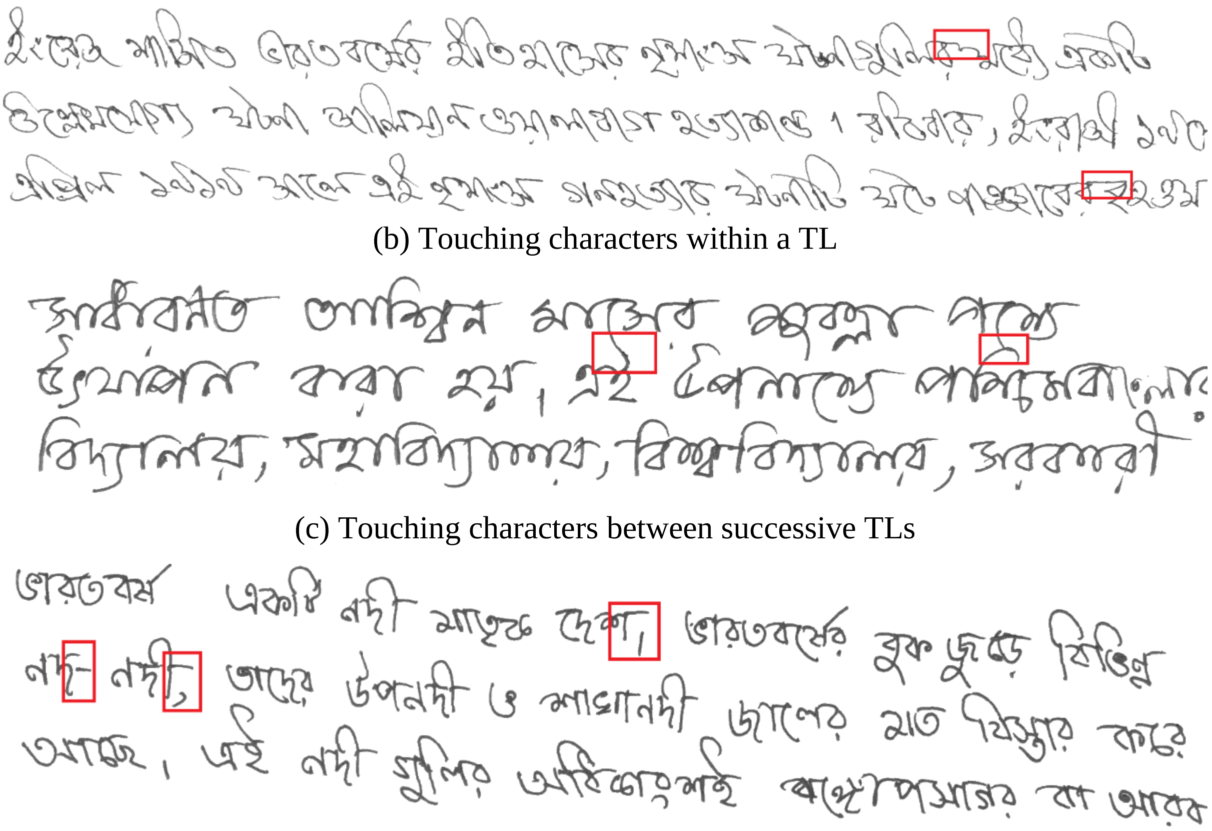
Fig. 4.1 Examples of variation in writing styles of different persons. Here, same content has been written by four different individuals

In addition to these, the following issues associated with handwritten document images add more challenges while extracting words from said document images.

- Non-uniform inter-word and inter-word spacing in a TL (refer to Fig. 4.2(a))
- Presence of touching characters within a TL (refer to Fig. 4.2(b)) or among successive TLs (refer to Fig. 4.2(c))
- Appearance of overlapping (refer to Fig. 4.2(d)) punctuation marks like comma (‘,’), period (‘|’ which is called as ‘DARI’ in *Bangla*), hyphen with characters within a TL.



(a) Non-uniform inter-word (blue rectangle) / inter-word (red rectangle) spacing



(b) Touching characters within a TL

(c) Touching characters between successive TLs

(d) Overlapping punctuation marks

Fig. 4.2 Examples illustrating various complex cases that researchers may have to face during word extraction from document images

4.1.1 Literature Review

Researchers have already tried to solve the problem of word extraction from handwritten documents but still the challenges exist. The techniques present in [46-47, 137] have used the conventional inter/intra word gap based analysis for extraction of words from handwritten TLs. The algorithm, described in [137], first divides a TL at all gaps that are larger than some initial threshold value. Then an iterative process which finds maximal of left and right gap of current CCs from its neighboring CCs at every iteration is followed. The CCs are further divided based on some estimated threshold depending on the estimated gap metrics. The work has been applied on isolated TLs taken from IAM online handwriting database [138]. In another work [139], authors have designed an algorithm based on a gap metric among successive CCs to isolate words from a TL. The estimated gap metrics are fed to linear SVM that uses soft-margin. For experimental purpose this work uses handwritten TLs collected from ICDAR database [140]. In [47], an adaptive gap metric is employed to perform word extraction from handwritten

Arabic TLs. This adaptive gap metric relies on deriving the gap values from the properties of each input TL.

In [141], a contour detection method is presented. In this work, initially the contour of the components present in a given TL are detected and then a threshold is chosen based on median and average of white run lengths present in the given TL. After that the words are extracted from the TLs based on the contour and the previously chosen threshold value. At last, these words are represented using bounded box enclosing all the object pixel present in it. Another method, described in [142], has used second order anisotropic Gaussian differential operator as a scale space technique to find the blobs present in a TL.

Run Length Smoothing Algorithm (RLSA) is a very common technique used in the field of image processing which is mainly used initially for block segmentation and text discrimination. It has also been used for extraction of words from TL [143]. Here, RLSA is applied on binarized TL image to convert each word present therein as CC. Spiral Run Length Smearing (SRLSA), a modified version of RLSA, is used in [46, 79] for extracting words present in a handwritten TL image. TLs are extracted from ground truth (GT) images taken from CMATERdb1.2.1 [24] (a standard document image database freely available to the research community).

Apart from these, a word extraction method, termed as page-to-word extraction technique, is introduced by the authors in the work [55]. In this work, words are directly extracted from document pages by bypassing the TL extraction step. In this work, at first, the authors have used Harris corner point detection algorithm [144] for detecting the key points on the entire document image. Then these key points are clustered using Density-based Spatial Clustering of Applications with Noise (DBSCAN) technique [145]. Finally, the boundary of the text words present in the document images are estimated based on the convex hull drawn for each of the clustered key points.

4.1.2 Motivation

The methods presented in [46-47, 79, 137, 139, 141-143] have considered individual TL as input for the word extraction purpose. However, obtaining such TLs from unconstrained handwritten documents needs a method for TL extraction from document image which in turns increases processing time. Also such segmentation of handwritten document page image may become error prone. For example, performance of TL extraction technique, described in previous chapter, is 0.9633 (F-measure score) which is evaluated on the document page images prepared during this thesis work. This fact ensures that applying a TL extraction method prior to word extraction from document images may decrease the performance of word extraction algorithm. Considering this very fact, in this chapter a page-to-word extraction method, similar to [55], is designed. However, the approach described in [55] is time consuming as it includes a number of operations like finding corner points using Harris corner point detection algorithm and clustering those detected corner points. Therefore, in this chapter a relatively faster method than [55] is introduced by evading the TL extraction overhead. This method is built using CC based approach.

It is to be noted that because of the lack of publicly available proper evaluation tool and standard large database containing documents and required GT images proper evaluation of the developed algorithm is not possible. An instance of such database containing document pages written in *Bangla* script and corresponding evaluation tool for the same is available through different ICDAR and ICFHR handwriting segmentation competitions. These database contains around 50 page-level writing samples which is very less in number for generalizing a word extraction technique. Hence, here a GT database, containing ideal word boundaries for all the 300 handwritten document pages (collected during this thesis work), has been prepared. Apart from the CC based word extraction technique and GT database, a comprehensive evaluation tool to evaluate any word extraction mechanism has been developed.

Additionally, the word extraction pipeline which is comprised of TL extraction from document page images first and then word extraction from these extracted TL images is also studied in this chapter. Here, TL extraction from handwritten document pages is performed by using the method described in Chapter 3 of this thesis. Next, words from each of the TLs are extracted using an existing technique which is described in the work [46].

4.1.3 Objective of the Chapter

Based on the above discussion the following objectives have been set and covered in this chapter:

- Design of an efficient page-to-word extraction mechanism.
- Preparing GT images of document pages showing ideal segmented words.
- Fitting the previously developed evaluation protocol, described in Chapter 3, for assessing the developed algorithm.

4.2 Page-to-word Extraction Technique

The primary objective of the present chapter is to describe a word extraction method which can extract words directly from a document page. This mechanism is defined as page-to-word extraction technique in this thesis work. This technique converts each word present a document page into virtual connected components (VCC) like blob in [142]. For this purpose, a CC based technique have been developed here. In this technique first CCs are extracted from binarized document images. Conversion of gray-scale image into binarized image is already described in Chapter 2. Next a two-stage joining protocol is followed to hypothetically connect the spatially close CCs. In the first stage of joining, each of the relatively smaller CCs are joined with corresponding larger CC. The decision to join two CCs are taken based on the distance between their centers of gravities (CGs). Whereas in the second stage, CCs that appear closely in horizontal direction are joined. In this case, distance between bounding boxes of any two CCs under consideration is used as decision parameter. Now, all these resulting CCs (i.e., VCC) are selected as word. The schematic diagram of the process is shown in Fig. 4.3 and the detail procedure is described in the following sections.

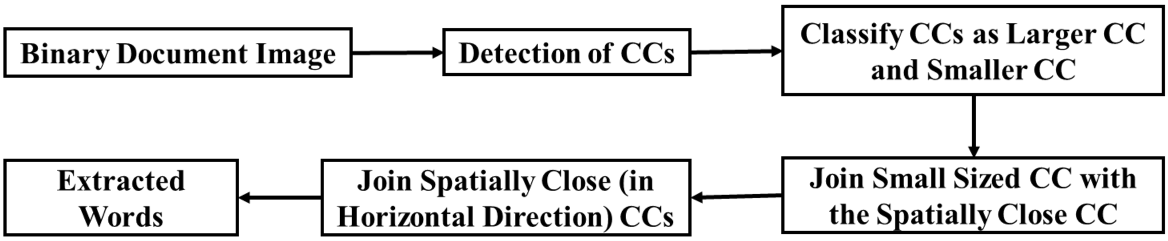

Binary Document Image
Detection of CCs
Classify CCs as Larger CC and Smaller CC
Join Small Sized CC with the Spatially Close CC
Join Spatially Close (in Horizontal Direction) CCs
Extracted Words

Fig. 4.3 Schematic diagram of page-to-word extraction method

4.2.1 Classification of CCs

In this approach, first, all the CCs present in a document image are extracted using 8-connected component labelling (CCL) algorithm [33]. Let, the CCs are $\mathfrak{C}_1, \mathfrak{C}_2, \dots, \mathfrak{C}_N$. Here, N is the number of CCs extracted from a document page. Now, the CCs are classified either as small sized or large sized using a rule-based algorithm. For this classification, height and width of the CCs are considered as feature values. Average height (μ_H) and average width (μ_W) of all $\mathfrak{C}_i$ are calculated as $\mu_\mathrm{H} = \frac{1}{\mathrm{N}}\sum_{i=1}^{N}\mathfrak{C}_i^h$ and $\mu_W = \frac{1}{\mathrm{N}}\sum_{i=1}^{N}\mathfrak{C}_i^w$. Here, $\mathfrak{C}_i^h$ and $\mathfrak{C}_i^w$ indicate the height and width of $\mathfrak{C}_i$. Now, a $\mathfrak{C}_i$ is classified into either small sized or large sized CC using the following rule:

$$size(\mathfrak{C}_i) = \begin{cases} smal, if\ \mathfrak{C}_i^h < \mu_H\ or\ \mathfrak{C}_i^w < \mu_W \\ large, otherwise \end{cases}$$

An example of document page containing both categories of CCs is shown in Fig. 4.4. In this figure red colored CCs are small sized CC and rest are large sized CC.

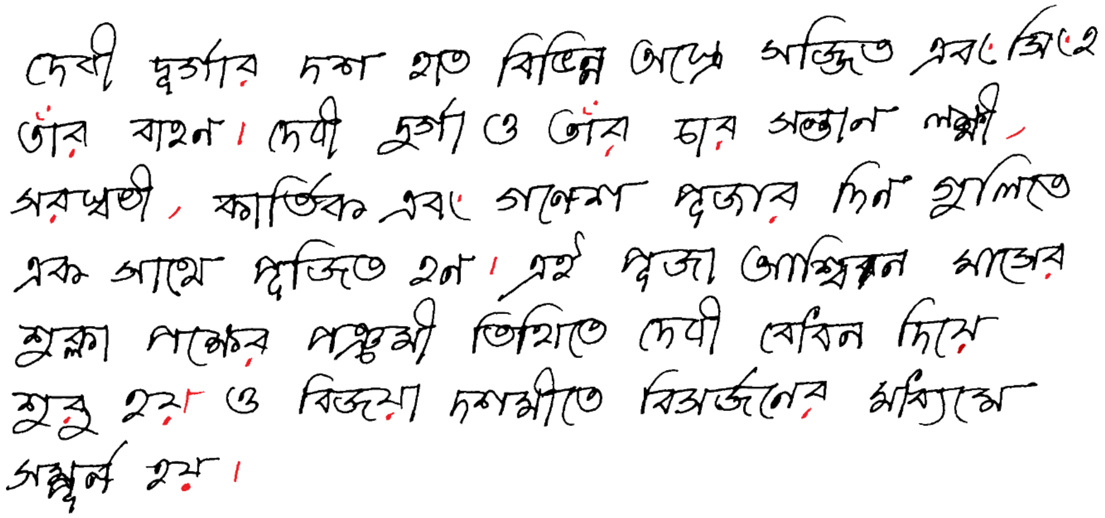


Fig. 4.4 An instance of document page where small sized CCs are displayed in red color

4.2.2 Joining of Small Sized CC to an Appropriate CC

In this section, the method for joining a small sized CC (say, $\mathfrak{C}_s$) with an appropriate CC (say, $\mathfrak{C}_l$) is described. An appropriate $\mathfrak{C}_l$ is selected using spatial closeness of the CCs under consideration. For this, all of the 8-Freeman directions [146] from the CG of $\mathfrak{C}_s$ (say, (C_x, C_y)) are visited. Let, the CG of first encountered $\mathfrak{C}_l$ in d^{th} direction is (C_x^d, C_y^d). Next, the Euclidean distance [17] (Δ_d) from (C_x, C_y) to (C_x^d, C_y^d) in d^{th}direction is calculated by

$$\Delta_d = \sqrt{(C_x - C_x^d)^2 + (C_y - C_y^d)^2}\ , \text{where } d = 1, 2, \ldots, 8$$

All such Δ_ds $(d = 1, 2, \ldots, 8)$ are calculated. Next, the minimum distance (say, min_{dist}) among all such distances is calculated i.e., $min_{dist} = \min_{d=1,2,\ldots,8}\{\Delta_d\}$ and the corresponding $\mathfrak{C}_l$ (say, $\mathfrak{C}_l^{min}$). Now, spatially closeness of these two CCs (i.e., $\mathfrak{C}_s$ and $\mathfrak{C}_l^{min}$) is defined using the following rule

$$spacially\ close(\mathfrak{C}_s, \mathfrak{C}_l^{min}) = \begin{cases} yes, if\ min_{dist} \leq \delta_1 \\ no, otherwise \end{cases}$$

Here, δ_1 is a threshold which is decided experimentally. Finally, for all $\mathfrak{C}_s$s present in a document image are hypothetically joined with their spatially close $\mathfrak{C}_l$s i.e., $\mathfrak{C}_l^{min}$. Such threading has been carried out by Digital Differential Analyzer (DDA) line drawing algorithm [136]. An instance of output image generated after this phase is shown in Fig. 4.5.

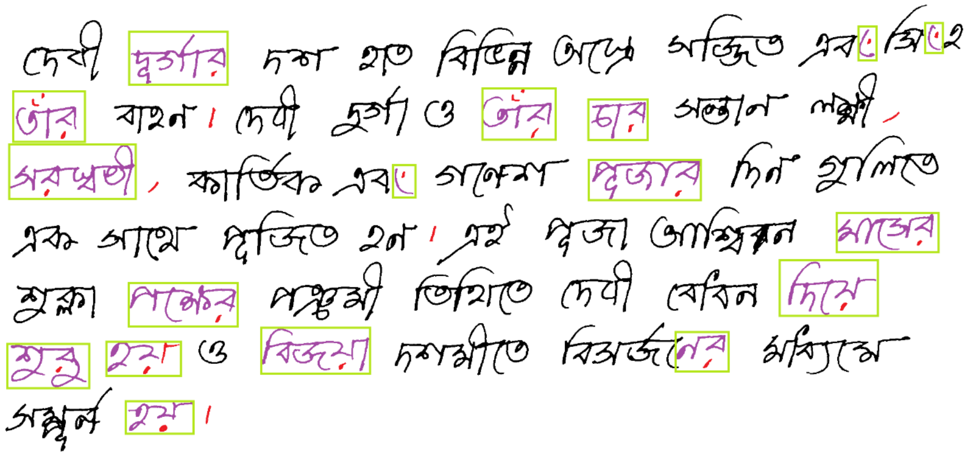

দেবী দুর্গার দশ হাত বিভিন্ন অস্ত্রে সজ্জিত এবং সিংহ
তাঁর বাহন। দেবী দুর্গা ও তাঁর চার সন্তান লক্ষ্মী,
সরস্বতী, কার্তিক এবং গণেশ পূজার দিন গুলিতে
এক সাথে পূজিত হন। এই পূজা আশ্বিন মাসের
শুক্লা পক্ষের পঞ্চমী তিথিতে দেবী বোধন দিয়ে
শুরু হয় ও বিজয়া দশমীতে বিসর্জনের মধ্যে
সম্পূর্ণ হয়।

Fig. 4.5 An instance of document page where small sized CCs (red color) and the corresponding CCs (purple color) to which the smaller ones would be connected. The CCs that have been formed after joining are marked with rectangular boxes

4.2.3 Joining CCs Horizontally

After completing the previous stage, the small sized CCs along with their corresponding large sized CC form new set of CCs. Let the new set of CCs are represented as $\mathfrak{C}'_1, \mathfrak{C}'_2, \dots, \mathfrak{C}'_{N'}$. Here, $N'(\leq N)$ is cardinality of new set of CCs. Next, left and right boundaries (i.e., columns) for all CCs are estimated. Let, for some $\mathfrak{C}'_i$ ($i \in [1, N']$ and $i \in \mathbb{N}$) the left and right columns are lc'_i and rc'_i respectively. Now, $\mathfrak{C}'_i$ and $\mathfrak{C}'_j$ ($i \neq j$) are joined hypothetically if the difference between rc'_i and lc'_j is less than some predefined threshold value (say, δ_2) i.e., $rc'_i - lc'_j \leq \delta_2$. Here, selection of $\mathfrak{C}'_j$ for some $\mathfrak{C}'_i$ is performed by visiting the components from CG of $\mathfrak{C}'_i$ to its left or right. Examples of merged CCs that are generated after joining CCs horizontally are shown in Fig. 4.6 and the final output of the current word extraction technique is shown in Fig. 4.7.

4.3 Preparation of Ground Truth Image

It has already been mentioned that the availability of GT information makes any database more useful since it helps enabling proper evaluation of any newly designed technique by comparing its output with the GT images. Considering this fact, in this thesis work, the GT images containing ideal word boundaries for all the document pages of present database are prepared. These GT images are prepared in a semi-automatic way. First, the page-to-word extraction mechanism as described in section 4.2 is applied on each of the binarized document pages and then all the identified words, present therein, are uniquely colored. The color encoding is made in such a way that neighboring words are painted with different colors. The present method sometimes generates over and /or under segmented words (see Fig. 4.8(a)). All such erroneous results are manually corrected using the GTgen Tool (refer to Fig. 4.8(b)). Such corrected and color encoded versions of automatically segmented document pages are considered here as GT images which are then added into the present database for future usages.

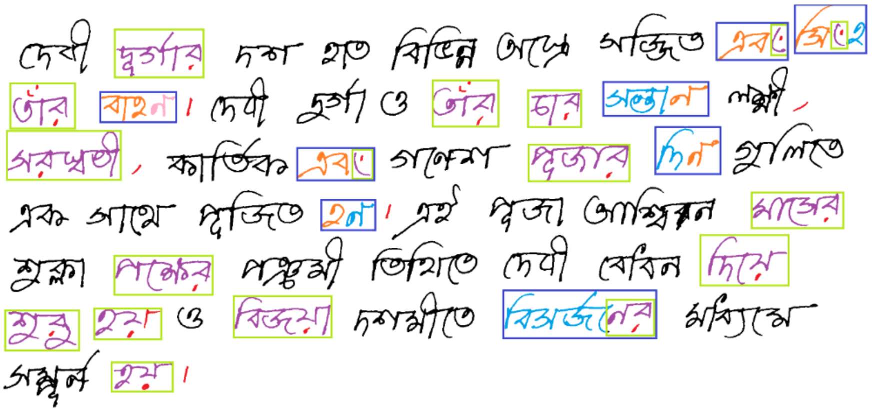

দেবী দুর্গার দশ হাত বিভিন্ন অস্ত্রে সজ্জিত এবং সিংহ
তাঁর বাহন। দেবী দুর্গা ও তাঁর চার সন্তান লক্ষ্মী,
সরস্বতী, কার্তিক এবং গণেশ পূজার দিন গুলিতে
এক সাথে পূজিত হন। এই পূজা আশ্বিন মাসের
শুক্লা পক্ষের পঞ্চমী তিথিতে দেবী বোধন দিয়ে
শুরু হয় ও বিজয়া দশমীতে বিসর্জনের মধ্যমে
সম্পূর্ণ হয়।

Fig. 4.6 Illustration of CCs that would be joined horizontally. VCCs, generated at previous step, are shown within light green colored rectangular boxes while the CCs (or VCCs) that would be joined during horizontal joining are enclosed within blue colored rectangular boxes

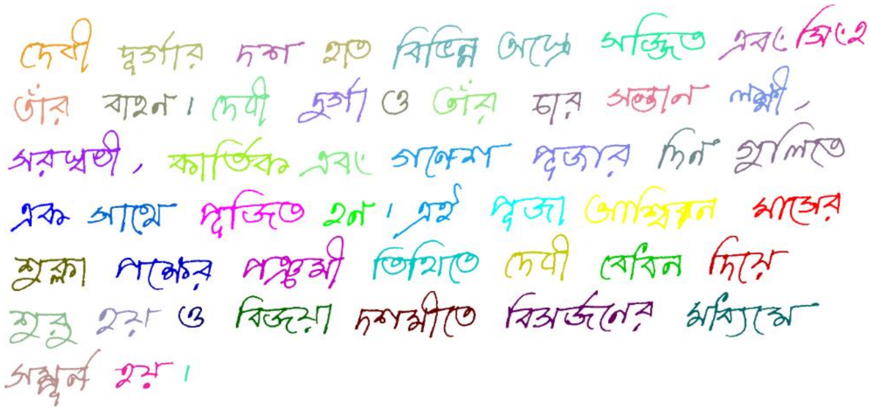
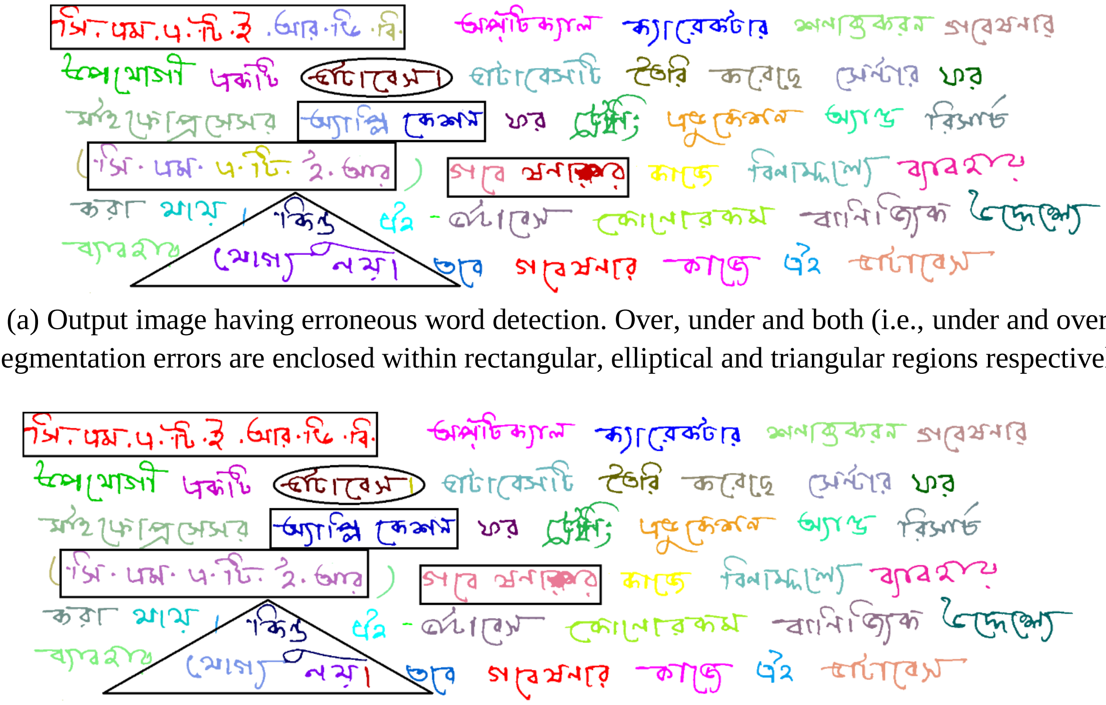

দেবী দুর্গার দশ হাত বিভিন্ন অস্ত্রে সজ্জিত এবং সিংহ
তাঁর বাহন। দেবী দুর্গা ও তাঁর চার সন্তান লক্ষ্মী,
সরস্বতী, কার্তিক এবং গণেশ পূজার দিন গুলিতে
এক সাথে পূজিত হন। এই পূজা আশ্বিন মাসের
শুক্লা পক্ষের পঞ্চমী তিথিতে দেবী বোধন দিয়ে
শুরু হয় ও বিজয়া দশমীতে বিসর্জনের মধ্যমে
সম্পূর্ণ হয়।

Fig. 4.7 Result after applying current page-to-word extraction technique. Here all the words are uniquely colored

4.4 Experimental Results

For experimental need, 300 handwritten document page images, which are in binarized form, are used. The present word extraction technique is applied on these handwritten document pages and the performance of the said technique is measured automatically. For this, the performance evaluation protocol as described in section 3.3.2 is used. To fit the said evaluation scheme for the performance measure of a word extraction method, extracted words in a document page and words in the corresponding GT image are considered as components. It has already been mentioned that the threshold values (i.e., δ_1 and δ_2) are optimized experimentally. To do this the present page-to-word extraction technique is applied on all the 50 handwritten document pages of Set-F of present database (refer to Chapter 2). The results with varying values of δ_1 and δ_2 are shown in Table 3.1 where bold style numbers indicate best scores.

Table 4.1 Average performances of TP, FN, FP, Recall, Precision and F-measure scores for all document pages belonging to Set-F with varying parameters (i.e., δ_1 and δ_2) of CC based page-to-word extraction.



(a) Output image having erroneous word detection. Over, under and both (i.e., under and over) segmentation errors are enclosed within rectangular, elliptical and triangular regions respectively.

(b) The manually corrected GT image

Fig. 4.8 Illustration of GT image preparation

By investigating the results shown in Table 4.1, the following conclusions might be drawn.

- Experimentation with $\delta_1 = 20$ and $\delta_2 = 60$ generates the best recall value. The best recall value indicates that it generates less number of under segmented words. Whereas, the precision value for this case is higher than the best precision value. So these threshold values (i.e., $\delta_1 = 20$ and $\delta_2 = 60$) generate more over segmented words while trying to reduce under segmented words.

- Experimentation with $\delta_1 = 10$ and $\delta_2 = 60$ provides the best precision score. Better precision value indicates that it tries to optimize among truly segmented words and under segmented words.

Table 4.1 Results of present word extraction technique while varying with the parameter values on document page images of Set-F

δ_1	δ_2	TP	FN	FP	Recall	Precision	F-measure
10	30	0.8632	0.1061	0.0308	0.8878	0.9618	0.9220
10	40	0.8805	0.0901	0.0294	0.9041	0.9640	0.9322
10	50	0.8936	0.0790	**0.0273**	0.9167	0.9678	0.9407
10	60	0.9001	0.0723	0.0275	0.9240	**0.9685**	0.9450
10	70	0.9007	0.0694	0.0299	0.9270	0.9661	0.9454
15	30	0.8964	0.0699	0.0337	0.9255	0.9614	0.9424
15	40	0.9072	0.0593	0.0335	0.9369	0.9624	0.9487
15	50	0.9125	0.0537	0.0338	0.9428	0.9626	0.9519
15	60	**0.9144**	0.0505	0.0351	0.9462	0.9615	**0.9530**
15	70	0.9124	0.0492	0.0385	0.9474	0.9579	0.9518
20	30	0.9080	0.0502	0.0419	0.9463	0.9545	0.9495
20	40	0.9141	0.0426	0.0430	0.9542	0.9537	0.9530
20	50	0.9142	0.0406	0.0446	0.9561	0.9521	0.9531
20	60	0.9136	0.0392	0.0471	**0.9573**	0.9494	0.9524
20	70	0.9105	**0.0387**	0.0507	0.9574	0.9456	0.9504

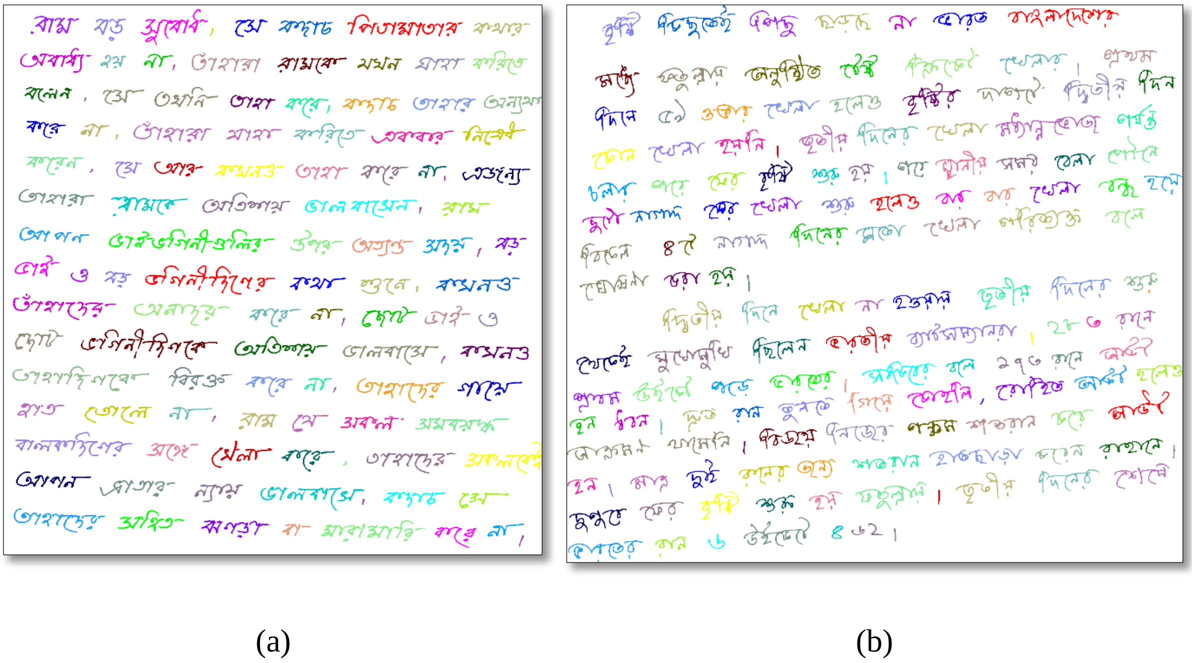
Apart from these, the better F-measure score ensures the better trade-off among over, under and truly identified words. Hence the best performing experiment is decided by average F-measure score which is harmonic mean between recall and precision. The experimentation with $\delta_1 = 15$ and $\delta_2 = 60$ has provided best average F-measure score on these 50 handwritten page samples. Therefore, this framework is applied on the entire database for the page-to-word extraction purpose. Two output instances of the present page-to-word extraction technique are shown in Fig. 4.9 where all the words, present in those document pages, are extracted properly. All the above mentioned measures are also calculated on the entire dataset. Such measures (page-wise) are plotted using line chart (see Fig. 4.10 (a-f)). To access the word extraction result on entire dataset, statistics like maximum, minimum, mean, standard deviation (SD), first quartile (Q1), second quartile (Q2) i.e., median, third quartile (Q3) and inter quartile range (IQR) are introduced. These information are given in Table 4.2.

4.4.1 Comparison with the Other Word Extraction Model

It has already been mentioned that in this work a page-to-word extraction technique is devised. To compare it with the traditional word extraction i.e., word extraction is applied on extracted the TLs. For this, first the TLs are extracted from handwritten document page images using the TL extraction algorithm described in the previous chapter and then words are extracted from each of the extracted TLs using the SRLSA based TL to word extraction technique as described in [46]. The comparative results are shown in Table 4.3. In this table only the average results are shown for comparison purpose.

Table 4.2 Experimental results in terms of different statistical measures of present page-to-word extraction technique

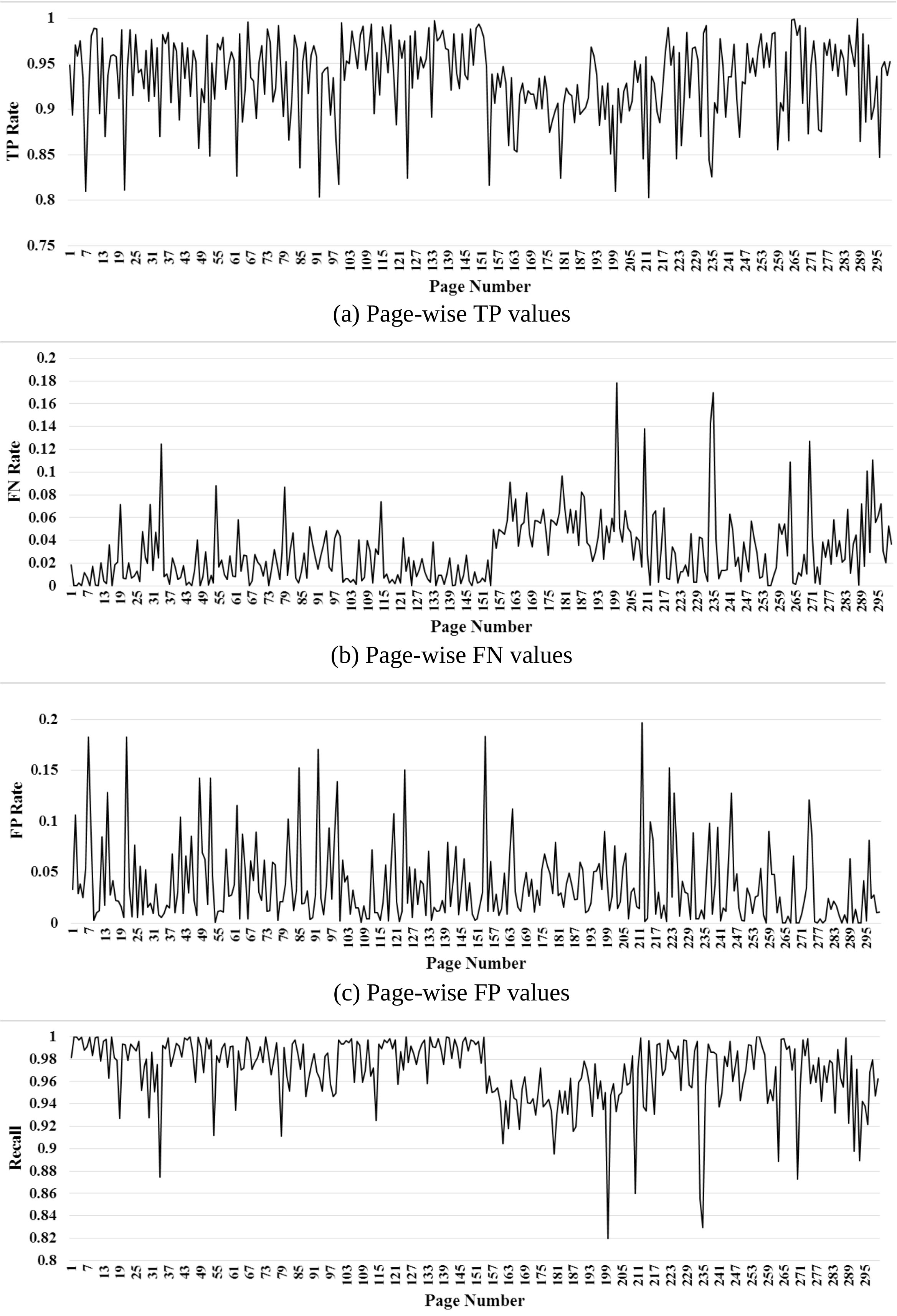
Term	TP	FN	FP	Recall	Precision	F-measure
Minimum	0.8024	0.0000	0.0000	0.8193	0.8029	0.8904
Maximum	1.0000	0.1785	0.1970	1.0000	1.0000	1.0000
Average	0.9336	0.0294	0.0370	0.9696	0.9620	0.9651
SD	0.0443	0.0290	0.0381	0.0297	0.0387	0.0242
Q1	0.9075	0.0070	0.0103	0.9537	0.9459	0.9515
Q2	0.9403	0.0222	0.0246	0.9768	0.9746	0.9692
Q3	0.9693	0.0448	0.0525	0.9928	0.9896	0.9844
IQR	0.0618	0.0378	0.0422	0.0391	0.0437	0.0329



(a)

(b)

Fig. 4.9 Two instances of document page image where all the words are extracted correctly by the present technique

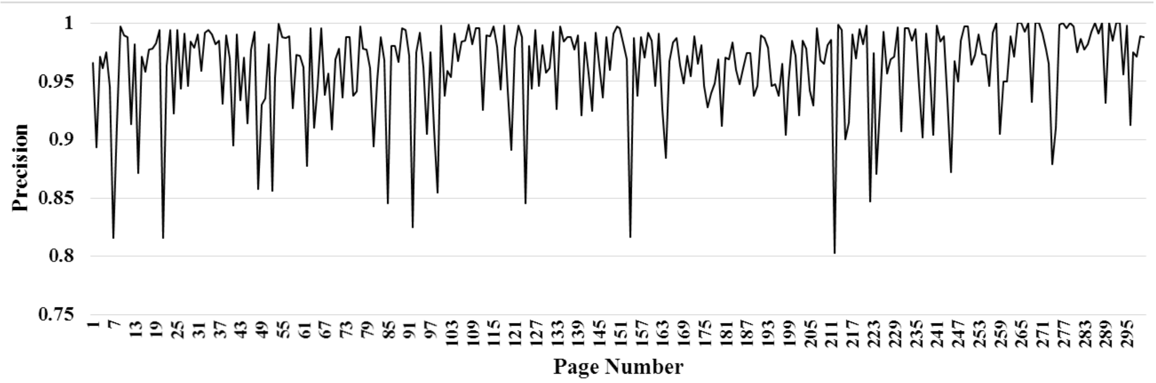


(a) Page-wise TP values

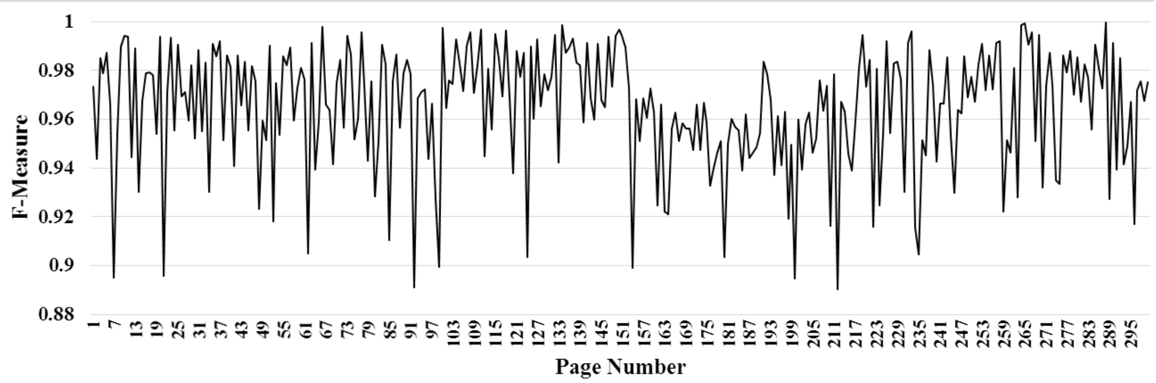
(b) Page-wise FN values

(c) Page-wise FP values

(d) Page-wise recall values



(e) Page-wise precision values



(f) Page-wise F-measure values

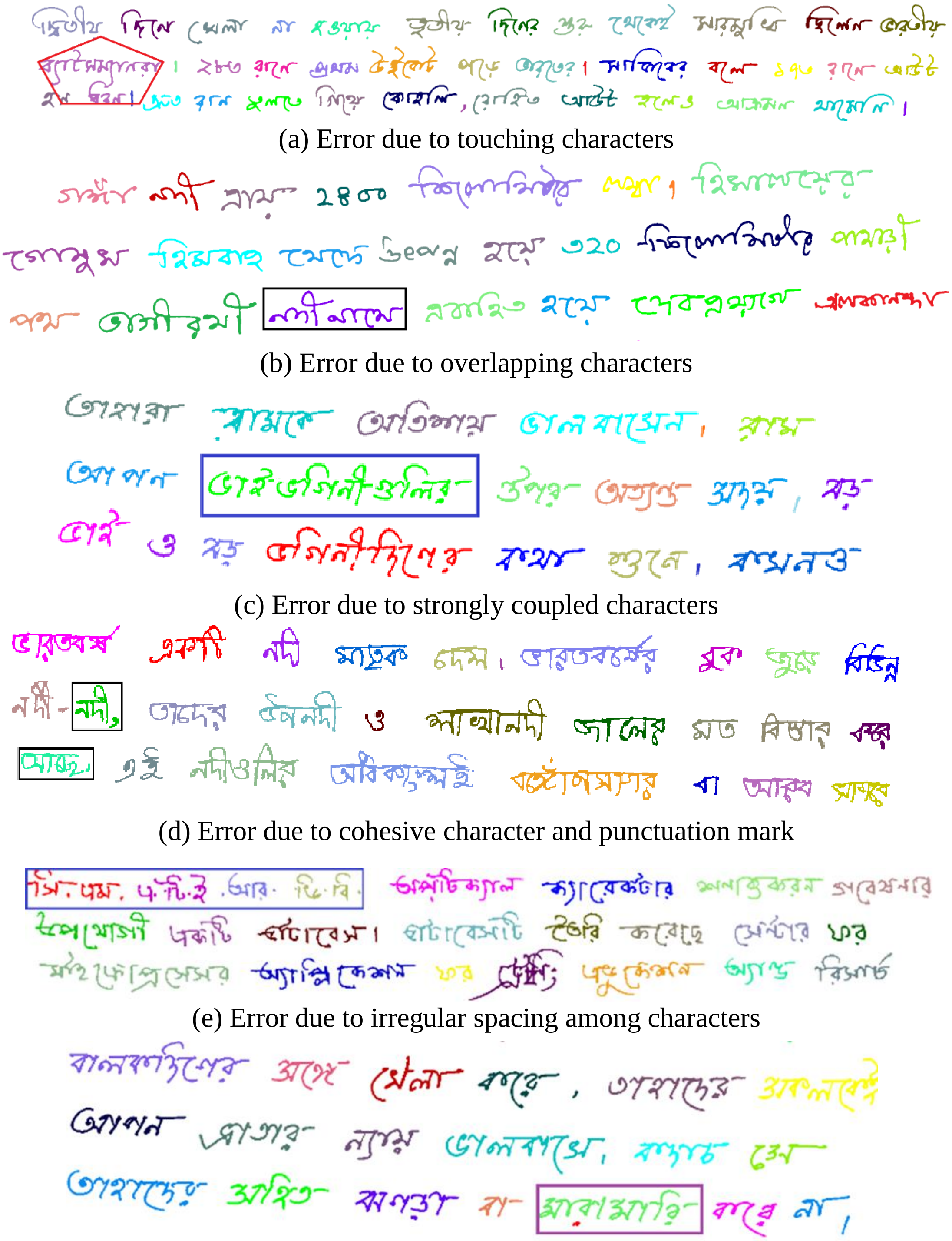
Fig. 4.10 Different page-wise performance measures of the present page-to-word extraction technique on the entire database

Table 4.3 Comparison of performances of the current word extraction techniques

Method	TP	FN	FP	Recall	Precision	F-measure
TL to word [46]	0.8936	0.0726	0.0338	0.9636	0.9249	0.9439
Page-to-word	0.9336	0.0294	0.0370	0.9696	0.9620	0.9651

4.4.2 Error Case Analysis

From the results shown in the Table 4.2, it is seen that recall value is higher than precision value. This scenario indicates that most of the errors produced by the present technique is due to under segmentation of words. Such under segmentation occurs mostly due to presence of inter-word touching characters (refer to Fig. 4.11(a))/ overlapping characters (refer to Fig. 4.11(b)) and strongly coupled characters (refer to Fig. 4.11(c)). These scenarios may occur within same TL or within neighboring TLs. In addition to these, occurrence of cohesive punctuation mark and characters (refer to Fig. 4.11(d)) is another reason for the said case. Whereas unusual distance within two characters of a word or between a character and part of character/modified shape leads to under segmentation error. These two cases of are depicted in Fig. 4.11 (e-f).



(a) Error due to touching characters

(b) Error due to overlapping characters

(c) Error due to strongly coupled characters

(d) Error due to cohesive character and punctuation mark

(e) Error due to irregular spacing among characters

(f) Error due to presence of distant character subpart

Fig. 4.11 Illustration of error cases generated by present word extraction method

4.5 Discussion

Extraction of words from unconstrained handwritten document images is one of the fundamental requirements in an OCR system. It is a difficult assignment for the researchers due to a presence of non-uniform inter/intra-word spacing, touching characters and close appearance of punctuation marks, which are obvious in handwritten documents. In this context, the present chapter has described a possible solution to the said research problem which is a Euclidean distance based component joining method. For assessing the technique, GT images containing ideal word boundaries for 300 document pages are prepared. It also uses an automated evaluation procedure for assessing the present page-to-word extraction method. This evaluation technique can easily be applied to other word extraction method. The present page-to-word extraction system is experimented on entire the database and the outcomes are satisfactory.

Chapter 5

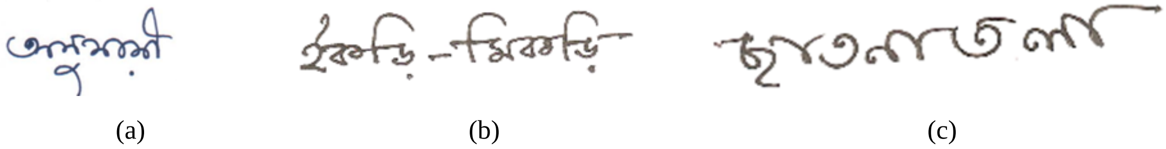
CHARACTER EXTRACTION FROM HANDWRITTEN *BANGLA* WORD IMAGES

5.1 Introduction

It has already been mentioned in Chapter 1 that Optical Character Recognition (OCR) system [15] converts optically scanned documents into machine encoded form. One of the major steps of OCR system is word recognition which aims to enable understanding of human readable printed and/or handwritten words to machines. For word recognition researchers have either followed analytical [15, 63] or holistic approach [63]. For the former one, segmentation of a word into constitute character(s) or character subpart(s) is a pre-requisite, whereas the later one is performed in segmentation-free way. Indeed the former one is better approach for recognizing words a lexicon independent way. Even, it has already been mentioned in Chapter 1 that number of middle zone characters or character like shapes, ascendant and descendant would help in reducing the search space which is formed by target words while searching a keyword in a recognition based way. Considering this facts character extraction from word image can be considered as one of the important stage for analyzing textual contents in a handwritten document image.

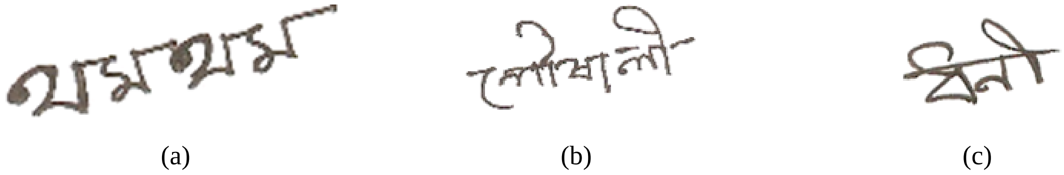
Character extraction aims at splitting the word image into characters or character like shapes since characters are the primitive segments of a word image. It is worthy to mention that character extraction from unconstrained handwritten word images [46] is more difficult while compared to the printed ones [81, 147]. The reason being that in a printed word, in general, all the characters have definite shapes and sizes since these are typed through the keyboard or typewriter. In addition, the problems related to unconstrained handwritten documents are much more complicated due to the wide variations in handwriting patterns of the individuals which also vary for same writer from time to time. These issues make the process of character extraction from handwritten word more difficult than its counterpart.

It is to be noted that character segmentation mechanisms are mostly script dependent [46]. The reason being the use of certain script specific features while designing any segmentation hypothesis. The present character segmentation technique is solely designed for *Bangla* script which is a *Matra* based script. Such initiative would help in designing better OCR system in other *Matra* based scripts like Devanagari, Gurumukhi and Assamese. Characteristics of *Bangla* script is described in detail in Chapter 1. However, the character extraction from handwritten *Bangla* word images is not an easy tasks. It is mainly due to the presence of non-uniform *Matra* (refer to Fig. 5.1(a-c)), skewed word images (refer to Fig. 5.2(a-c)), joining of consecutive characters at region(s) other than *Matra* region (refer to Fig. 5.3(a-c)), extended part(s) of middle zone character (refer to Fig. 5.4(a-c)) and multiple parts of characters that may be fall in *Matra* region (refer to Fig. 5.5(a-c)).



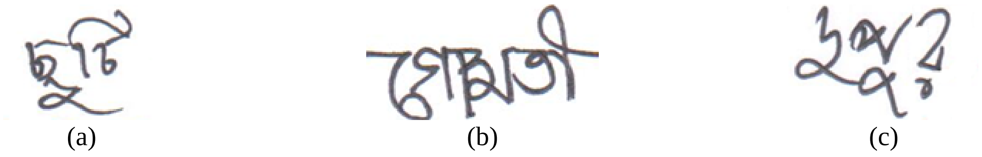
(a) (b) (c)

Fig. 5.1 Sample images showing presence of non-uniform Matra



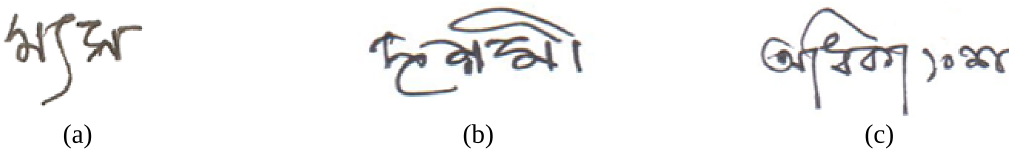
(a) (b) (c)

Fig. 5.2 Skewed word images



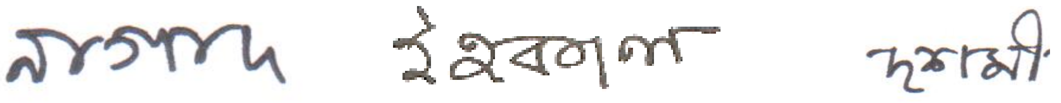
(a) (b) (c)

Fig. 5.3 Sample images that show joining of two characters except *Matra* region



(a) (b) (c)

Fig. 5.4 Sample word images having elongated part of middle zone character



(a) (b) (c)

Fig. 5.5 Word images having multiple parts of certain characters in *Matra* region

5.1.1 Literature Review

A number of research initiatives [46, 60-62, 82-83, 148-159] have been made by researchers for character extraction from handwritten words. In this section, first the researches that employ technique for character extraction from isolated handwritten *Bangla* word image [46, 60-62, 82-83, 148-150] are pronounced and then some contemporary works [46, 60-62, 82-83, 148-150] performed on words written in other scripts for the said task are surveyed. While considering researches on handwritten *Bangla* script two major research approaches are found in literature. The first category of techniques [60, 62, 148-150] has considered a word image as an indivisible component. Whereas in the other category of works [46, 61, 82-83], connected components (CC) containing two or more characters are only considered for extracting character from them.

In the work [148], the authors have proposed a method of recursive contour following to detect the segmentation points within a dynamically set region where the common *Matra* of the characters of a word lies. The region's upper boundary is set to *Matra* line whereas the bottom boundary is set using thickness value and a dynamically set weighted measure of contour pixels. Finally, the lower contour pixels of a word image that lie in the above mentioned region are considered as segmentation points. A nature inspired concept called water reservoir has been introduced in [149] for extraction of characters from handwritten *Bangla* words. In this concept it is assumed that water is poured from top and bottom of the word images which in turn generates water reservoir. Next, the reservoir base points and structural feature of the component are analyzed to detect the true segmentation point(s).

In another work [62], Multilayer Perceptron (MLP) based classification technique has been employed. In this work first the lower contour of a word image is estimated and then feature values are extracted for each object pixel on the estimated lower contour. Features that have been extracted in this work are 8-Freeman direction values of nearby pixels (excluding the current pixel) up to certain limit. In addition, three positional measures of the current pixel *viz.*, distance of pixel from top of the word boundary, number of object pixel along vertical direction containing the underlying object pixels and value of horizontal object pixel run containing the pixel, are also included in the feature set. Finally, these feature values are used to classify a contour pixel either as segmenting point or non-segmenting point. A triangular fuzzy membership function along with horizontalness feature in terms of object pixel has been used in [60] for detecting *Matra* region. The same fuzzy membership function is further used to calculate membership value of each of the object pixels that lie in the estimated *Matra* region. Number of object pixels and distance of furthest object pixel from *Matra* region along each column are estimated as feature values. A similar work is found in [150]. In this work instead of using triangular fuzzy membership function, bell-shaped fuzzy membership function has been used. Words written in four scripts *namely, Bangla*, Devanagari, Gurumukhi and Syloti have been considered as inputs.

The aforesaid methods provide good result while the characters present in the word image are connected though *Matra*. But, this scenario is very unusual since presence of disconnected components inside a word image are found very frequently while considering unconstrained handwriting. In addition to this, isolated characters or character sub-parts, very often, appears in unconstrained word images. Therefore researchers in the works [46, 61, 82-83] have followed alternative approach where already isolated characters, in general, are not passed through the actual segmentation process i.e., only CCs that contain multiple characters / character shapes are considered for actual segmentation process. In this category of works (refer to [46, 61, 82-83]), generally, CCs that are only considered for segmentation process are identified prior to actual segmentation process. For example, in the work [61], the authors have first extracted the CCs of a word image and then each CC is passed through segmentation process. During segmentation of CCs, first the lower contour of a CC is estimated and then the lower contour is traced anti-clockwise manner to detect the segmentation points on it using three predefined patterns. The final set of segmentation points is confirmed considering their distance from *Matra* region. Such confirmation process reduces segmentation of isolated character or character like shape that appears as individual CC in upper zone or lower zone of a word image. But, still this segmentation method suffers from segmentation of isolated character that appears in the middle zone of a word image (see Fig. 5.5(a-c)).

To get rid of the above mentioned issues, in the work [82], first the CCs of a word image is extracted and then these CCs are classified into either of the two classes, namely, ‘Segment further’ and ‘Do not Segment’ using a MLP based classifier. A 7-element feature vector has been designed for the classification purpose. Finally a CC, which has been classified as ‘Segment further’ is segmented using the segmentation logic used in [150]. This technique sometime suffers from classification error, failure in detecting appropriate segmentation points or extracting characters though detected segmentation points. In addition to these, it also suffers from loss of object pixels but which is not much alarming.

Considering the loopholes, the authors in [46, 83] have provided some alternative solutions. In these works, a 12-element feature vector has been used for better classification. In addition, segmentation lines having minimum number of object pixels for each detected segmentation point clusters have been drawn to get rid of loss of object pixels. This mechanism also helps in segmenting connected characters if at least one segmentation point is detected in the *Matra* zone. Finally, the scanning region while extracting features as mentioned in [60, 150] has been shortened for estimating better *Matra* region and segmentation points.

Beside the work on handwritten *Bangla* script, some research attempts are found in literature for other *Matra* based scripts like Devanagari [150-152] and Gurumukhi [150, 153-154]. A fuzzy bell-shaped membership function, used in [150], has been used for both *Matra* region detection and confirming *Matra* pixels as segmentation points in [151]. Whereas in the work [152], the authors have presented Pixel Plot and Trace and Re-plot and Retrace (PPTRPRT) model for the same purpose. In this work a feed-forward MLP has been used to recognize the segmented character. Whereas in the work [153], the authors have first generated CCs of a handwritten words written in Gurumukhi script and then selected CCs are considered for further segmentation. This selection is performed using intuitively selected threshold value which represents aspect ratio of a CC. These authors have also designed a merging strategy which solely depends of inter-CC distance. However in the work [154], the authors have recommended a water reservoir based model for extraction of characters from a word written in Gurumukhi script.

Apart from these above mentioned research attempts, some contemporary works [155-157] related to character extraction on non-*Matra* based scripts are also found in the literature. A recognition based character extraction method for word images written in Roman script is found in [155]. In this work, first a word image is segmented vertically through valleys of vertical histogram of object pixel count and then passed through pre-trained character model. Genetic Algorithm (GA) has been used for selecting optimal set of segmentation points. A similar approach to [154] has been described in [156] for extracting characters from handwritten English word images. In the work [157], a recognition based character extraction from handwritten Japanese words has been proposed which relies on over-segmentation strategy. It first estimates candidate separating points through projection analysis and graph generation, and then generates candidate patterns. Next it evaluates each candidate pattern by analyzing their overlapping and connectedness with other candidate patterns. Finally, optimal separating line(s) is (are) generated using the score returned from previous step. Another recognition based approach is found in [158] where main focus has been given on separation of non-touching characters which are not linearly separable from handwritten Japanese words. Instead of finding segmentation line like the work reported in [157], it generates segmentation path to separate two characters. However, prior to the actual segmentation process, it clusters the vertically separable patterns into isolated character, non-linearly separable and touching using a non-linear clustering method. The method has been experimented on handwritten English and Chinese words. More works can also be found in [159-160] for the words written in Arabic [159], Roman [160] and Chinese [160] scripts.

5.1.2 Motivation

Research initiatives [46, 60- 62, 82-83, 148-160], discussed in the previous section, have dealt with the problem of character extraction from handwritten word images. These works are either applied on *Matra* based scripts [46, 60- 62, 82-83, 148-154] or *non-Matra* based scripts [155-160]. However, from technical point of view, these works can be considered as (i) direct word segmentation approach (consider the entire word as input and segment them by some means [60, 62, 148-154]), (ii) CC segmentation based approach (consider only the CCs that actually need segmentation [46, 61, 82-83]) and (iii) recognition based approach (detect the characters, present in a word image, based on a classifier feedback [157-160]). The techniques that followed second or third approach need large number of samples, collected in offline mode, for training the classifier under consideration which require huge manual effort. Not only this, these techniques are heavily dependent on classifier performance and the nature of collected train samples. In addition, they are also time consuming. Considering these it can be safely claimed that the methods that follow the first approach are faster and more appropriate for real world applications. Therefore, in this thesis, a technique is designed following first approach. It is to be noted that despite being a challenging research problem, this field faces following research gaps:

- **Absence of open access database for character segmentation:** In most of the cases, word images those have been used for assessing character segmentation algorithms are not made publicly available [46, 60- 62, 82-83, 148-160]. Not only this, the ground truth (GT) images those represent ideally segmented characters or character subpart(s) are also missing. Such GT information of any database enhances its usability as researchers, even without any knowledge of the script in which the words are written, can work effortlessly.
- **Common evaluation methodology:** It is also seen that there is an absence of common and globally accepted evaluation protocol for qualitative assessment of character segmentation algorithms. It is worthy to mention that quantitative estimations are mostly performed manually, which is a tiresome process and also suffer from human errors. For example, the works [46, 60- 62, 81-83, 148-154, 158-160] have made the assessment of their algorithm in manual way.

Relying on the above discussion, in this thesis, a character extraction technique is designed by hybridizing two fuzzy membership functions that are known as trapezoidal [81] and bell-shaped [82] fuzzy member functions. Also, a database is prepared primarily for quantitative evaluation of character segmentation algorithm. This database contains GT images for each word image which is explained later. The performance of this character segmentation algorithm on the prepared database has been evaluated using the said GT information.

5.1.3 Objective of the Chapter

The work described in this chapter mainly aims to achieve the following objectives:

- Designing a competent character segmentation technique.
- Preparing GT images showing ideal segmented character and /or character sub-parts, of isolated word images.
- Fitting the previously developed evaluation protocol, described in Chapter 3, for assessing the present algorithm.

5.2 Character Extraction Technique

In this chapter, a new character segmentation mechanism for isolated handwritten *Bangla* word images has been presented. This method considers an entire word image and generates the segmented character(s) or character like shape(s). The entire segmentation methodology is divided into 5 different steps (see Fig. 5.6). First a mask based technique has been adopted to estimate the horizontal zone boundaries (zone and zone boundaries of a *Bangla* word image is shown in Fig. 1.2 in Chapter 1). Next, a fuzzy method is designed to detect *Matra* pixels and segmentation points. From these segmentation points segmentation lines with minimum loss of object pixels are estimated. Some segmentation lines are then nullified using a rule based method. In addition to these, to isolate the modified shapes belonging to lower zone, a separate method for segmenting lower zone component(s), if present, is adopted here. Before going into detail segmentation methodology, data (isolated handwritten word images) preparation process is described first.

5.2.1 Database Description

It has already been mentioned that handwritten word recognition (HWR), core of any handwritten OCR system, is a challenging job till date. The development of an analytical approach for HWR, which is an age-old and philosophically most sound mechanism, will get a boost if a comprehensive database of handwritten words is available. Keeping these facts in mind, in the present work, a database of isolated handwritten *Bangla* words is prepared. This database contains both dictionary and non-dictionary words of *Bangla* language that contain only basic characters and modified shapes (i.e., the words do not contain any compound character).

5.2.1.1 Data collection and analysis

To collect the word images two different types of document images which are i) document pages containing standard textual content i.e., document pages prepared during the present thesis work and ii) datasheets containing isolated words (see Fig. 5.7), have been considered. In both cases, texts/words are written freely on a white A4 size paper using black/blue ink and then those pages are scanned using a flatbed scanner as a gray-scale image with 300 dpi resolution. Next, the words are cropped manually from these scanned documents.

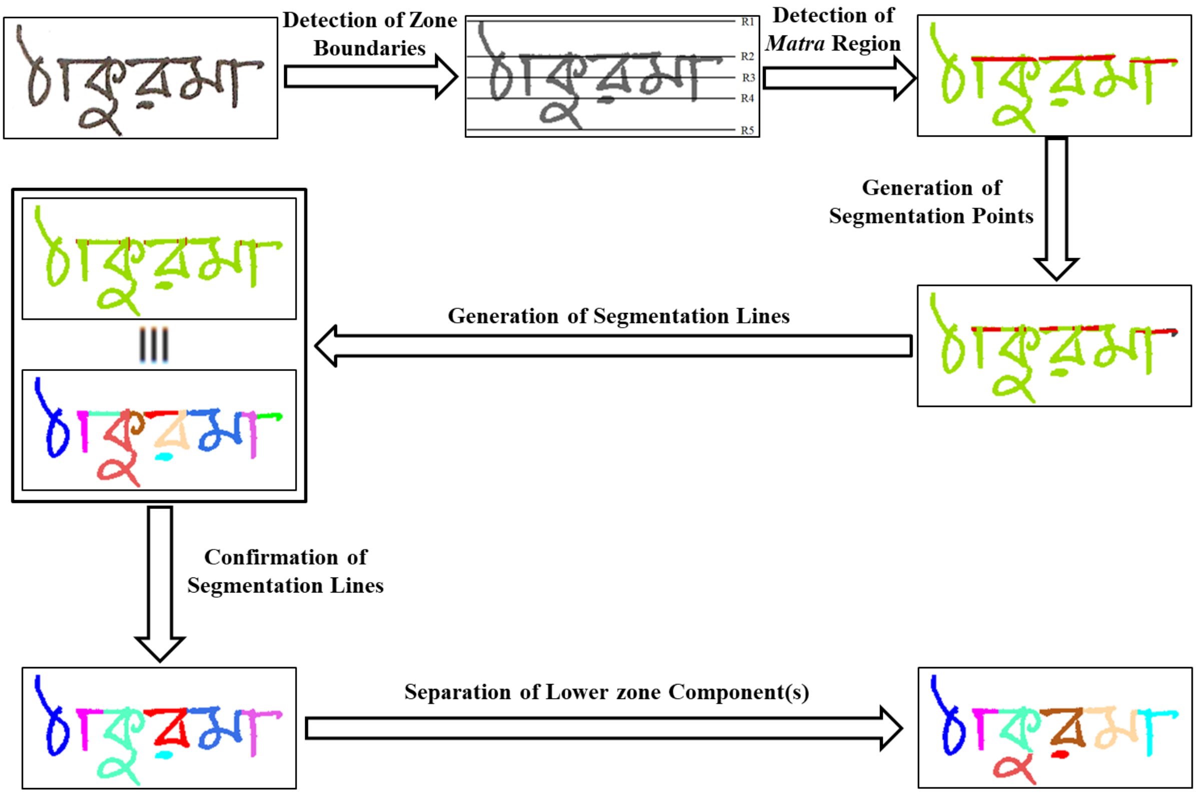

Detection of Zone Boundaries
R1
R2
R3
R4
R5
Detection of *Matra* Region
Generation of Segmentation Points
Generation of Segmentation Lines
Confirmation of Segmentation Lines
Separation of Lower zone Component(s)

Fig. 5.6 Block diagram of the character segmentation mechanism

It is worth in mentioning that selection of the words is carried out to cover all the commonly used basic characters and modified shapes of *Bangla* script. Finally wrongly spelled dictionary words are filtered out using the *Bangla* word vocabulary book entitled "*Akademi Banan Abhidhan*", published by *Paschimbanga Bangla Akademi* [161]. The database also contains few non-dictionary words. The entire database contains in total 5000 handwritten isolated word images. Some statistics related to word samples are shown in Table 5.1. The statistics and parameters, used here for indicating characteristics of collected word images of the present database, have been described in Chapter 2 (see section 2.4).

5.2.1.2 Naming of word images

All the collected word images are named as word####. Here #### represents index number of the word images ranging from "0001" to "5000". The index number ($Index_w$) is determined using the convention described below.

A lexicon $\mathcal{L}(=\{\ell_i : i = 1, 2, \dots, n_L\})$ has been prepared containing minimum set of *Bangla* words corresponding to each of the collected word images of $\mathcal{W} = \{w_i : i = 1, 2, \dots, n_W\}$ i.e.,

$\mathcal{W}$ is the word image database. Here, $n_{\mathcal{L}}$ and $n_{\mathcal{W}}$ are number of unique words in $\mathcal{L}$ and $\mathcal{W}$ respectively. Let, $\mathcal{A} = \{a_i : i = 1, 2, \ldots, n_{\mathcal{A}}\}$, where $n_{\mathcal{A}}$ is the number of basic characters in *Bangla* script, be the set of all basic *Bangla* characters appearing in the first position of the words in $\mathcal{L}$ and index i is in chronological order of the basic character in *Bangla* alphabet. Now, all the words in $\mathcal{L}$ are indexed based on position of first character in $\mathcal{A}$. Here, $|\mathcal{L}| < |W|$ as some $\ell \in \mathcal{L}$ have multiple copies of $w \in \mathcal{W}$. Here, for the said database, $n_{\mathcal{W}} = 5000$, $n_{\mathcal{A}} = 41$ and $n_{\mathcal{L}} = 1298$. Next, a relation $\rho \subseteq \mathcal{L} \times \mathcal{W}$ ($x \rho y$ if x is "machine encoded form" of y, $x \in \mathcal{L}$ and $y \in W$) is defined to name all the word samples $w \in \mathcal{W}$. Next, $\mathcal{W}$ is partitioned into $n_{\mathcal{L}}$ number of subsets defined as $\mathcal{P}_i = \{(\ell, w_j) : \ell = \ell_i \wedge w_j \in \mathcal{W}\}$ where, $i = 1, 2, \ldots, n_{\mathcal{L}}$ and $j = 1, 2, \ldots, k$. Finally, the index number ($Index_w$) of a word w is defined as $Index_w = \sum_{i=1}^{n-1} |P_i| + j$, where, $n = 1, 2, \ldots, n_{\mathcal{L}}$and j is the index of $w \in \mathcal{W}_i'$.

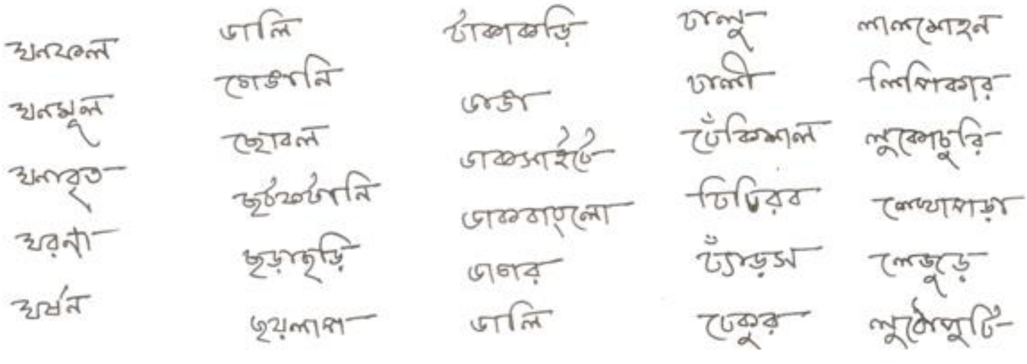


Fig. 5.7 Sample handwritten written document page containing isolated word images written by different individuals

Table 5.1 Description of statistical parameters used to define page images. The meaning of statistical measures and parameters of word image are described section 2.4

Term	Minimum	Maximum	Average	SD	Q1	Q2	Q3	IQR
Ht	19	185	66.41	28.18	41	63	86	45
Wd	41	897	182.20	75.33	129	167	221	92
Area	902	105846	13235.50	9769.39	5510	10800	18424	12914
#CC	1	16	3.37	1.70	2	3	4	2
AR	0.55	9.49	2.98	1.11	2.18	2.85	3.57	1.39
pixelDen	0.04	0.25	0.13	0.03	0.10	0.12	0.14	0.04
avgStrokeHt	2.65	16.06	6.62	2.27	4.23	7.08	8.36	4.13
avgStrokeWd	2.38	17.39	7.28	2.41	4.88	7.57	8.95	4.07

5.2.1.3 Pre-processing of the word images

In this stage, all the isolated word images are first passed through MMC based filtering technique (discussed in section 2.3.1 of Chapter 2) and then binarized using ratio based global thresholding method (explained in section of 2.3.2 in Chapter 2). Effect of this binarization technique over commonly used Otsu's method is illustrated in Table 5.2. Then all the connected components (CCs) of a binarized word image are extracted using 8-way connected component labelling (CCL) algorithm [115]. Finally, small and unwanted CC(s) (containing 5 or less number of object pixels) are removed. Some examples of word images after removing the unwanted CCs are shown in Table 5.3. These binarized word image samples, at this stage, have been considered for preparing GT images. Five sample images of the current database and their corresponding binarized images have been shown in Table 5.4. More samples of collected word samples are shown in Table A1 of appendix.

Table 5.2 Pictorial illustration of ratio based binarization technique over Otsu's technique

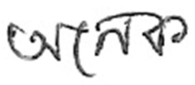
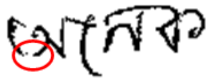
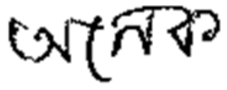
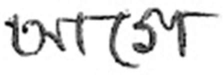
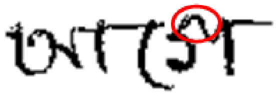
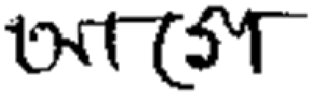
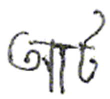
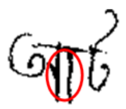
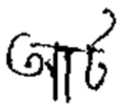
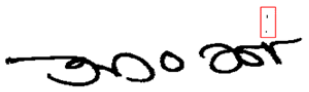
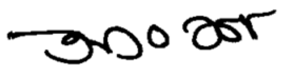
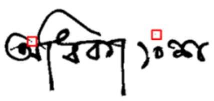
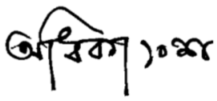
Original image	**Binarized image using**	
	Otsu's threshold	**Ratio based threshold**
		
		
		

Table 5.3 Outcomes of noise removal method

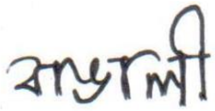
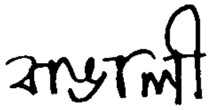
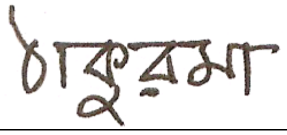
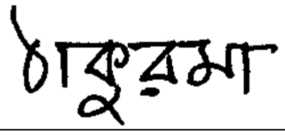
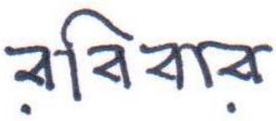
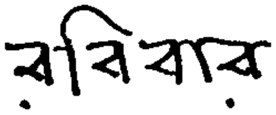
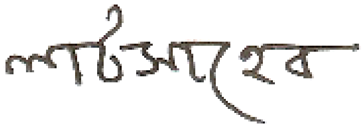
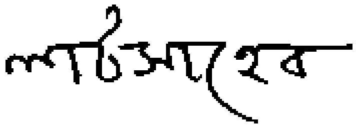
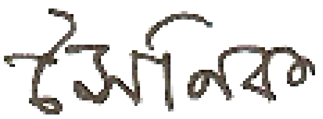
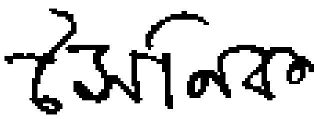
Sample Binarized image with unwanted pixels	**Image after removing unwanted pixels**
	
	

5.2.1.4 Analysis of the Word Images Database

In this section, association between $\mathcal{L}$ and $\mathcal{W}$ has been drawn. First, the distribution of ℓ ($\in \mathcal{L}$) with respect to varying number of $w (\in \mathcal{W})$ (i.e.,$|\mathcal{P}_i|$) is prepared and is depicted in Fig. 5.8. From this figure, it is found that the majority of the words (68% of $n_{\mathcal{L}}$) in $\mathcal{L}$ have two or more $w (\in \mathcal{W})$ (i.e.,$|\mathcal{P}_i| \geq 2$, $i = 1, 2, \ldots, n_{\mathcal{L}}$) and out of which 193, approximately, 15% of $n_{\mathcal{L}}$, have 8 or more sample variations in $\mathcal{W}$ (i.e.,$|\mathcal{P}_i| \geq 8$, $i = 1, 2, \ldots, n_{\mathcal{L}}$).

Comparison in number of basic characters (see Fig 5.9 (a)) and modified shapes (see Fig. 5.9(b)) in a word in $\mathcal{L}$ and $\mathcal{W}$ is provided here. The figures reveal that more than 70% of the words have at least 3 basic characters. It also shows that no word containing single character, rarely found in any language, is present in the database. On the other hand, from Fig. 5.9(b) it is clear that around 87% of the words contain modified shapes which in turn increase complexities during segmentation process. Variations with respect to presence of basic characters and modified shapes are presented in Fig 5.10(c). It indicates that most of the word images contain 4 or more such counts.

Table 5.4 Five samples from the database with their binarized version

Sl. #	**Word Sample**	**Preprocessed Binarized Image**
1		
2		
3		
4		
5		

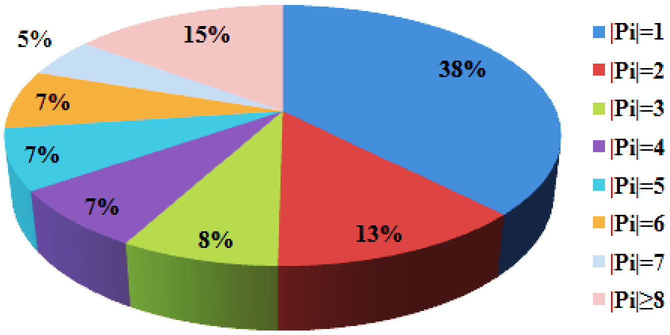
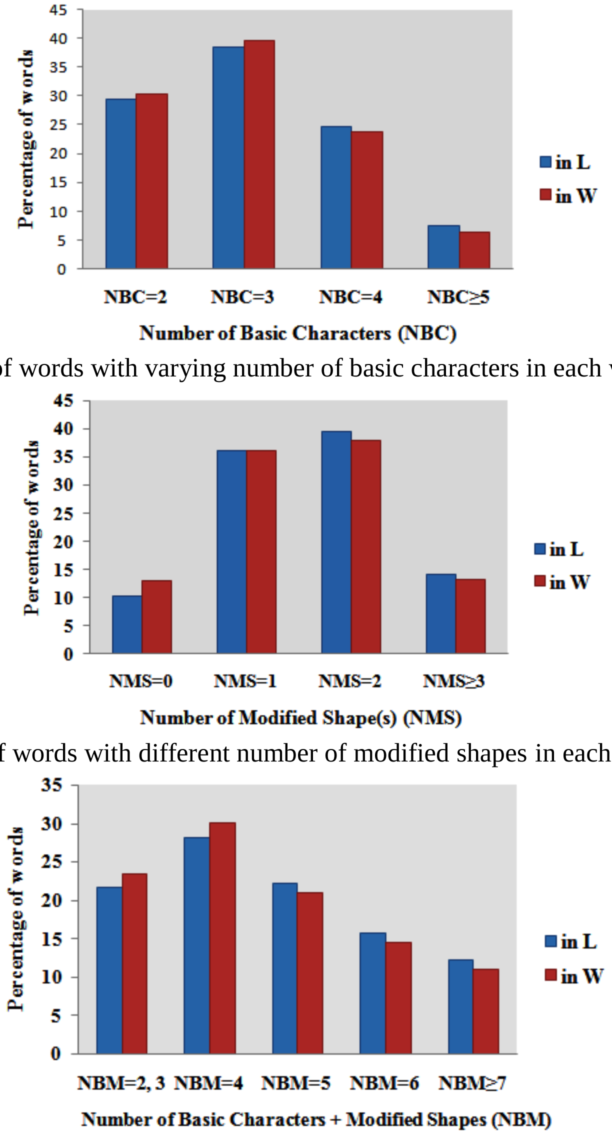


Fig. 5.8 Distribution of ℓ ($\in \mathcal{L}$) with respect to varying number of w ($\in \mathcal{W}$)



(a) Distribution of words with varying number of basic characters in each word in $\mathcal{L}$ and $\mathcal{W}$

(b) Distribution of words with different number of modified shapes in each word in $\mathcal{L}$ and $\mathcal{W}$

(c) Distribution of words with varying number of basic characters with modified shapes in each word in $\mathcal{L}$ and $\mathcal{W}$

Fig. 5.9 Comparison of different statistics in $\mathcal{L}$ and $\mathcal{W}$

5.2.2 Detection of Horizontal Zone Boundaries

Any binarized *Bangla* word image (say, B) can be represented as a set of pixels as $B = \{f(i,j): 1 \leq i \leq Ht \wedge 1 \leq j \leq Wd\}$, where Ht and Wd are height and width of B respectively. Here $f(i,j)$ assumes the value either $'0'$ or $'1'$ (where '0' represents background and '1' represents object pixels). Any *Bangla* word image can be divided into three non-overlapping zones *viz.*, upper zone, middle zone and lower zone [46] (see Fig. 5.10). Estimation of the zone separation lines, i.e., starting row of upper zone ($R1$), ending row upper zone or starting row of middle zone ($R2$), ending row of middle zone / starting row of lower zone ($R4$), ending row of lower zone ($R5$) and middle row of $R2$ and $R4$ (i.e., $R3$) are carried out here. The positions of Ri (i=1, 2, 3, 4, 5) are shown in Fig. 5.10.

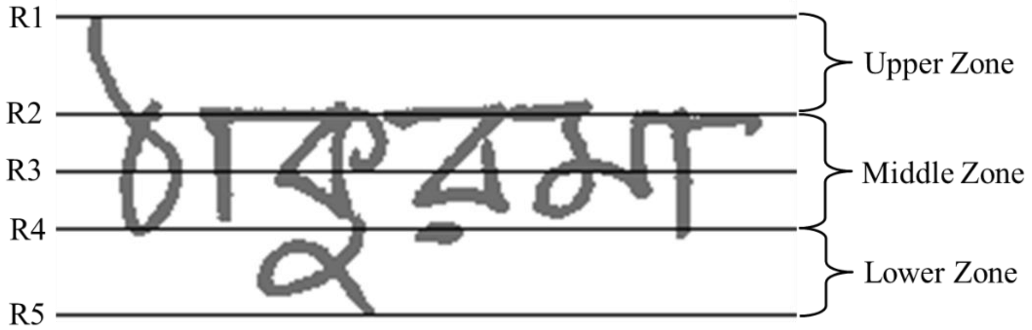


Fig. 5.10 Positions of three zones of a *Bangla* word. It also shows the zonal boundaries i.e., R1, R2, R4 and R5 and middle of R2 and R4, i.e., R3

Detecting Ri's ($i = 1, 3, 5$) are straight forward. But, estimation of $R2$ and $R4$ is the main challenge for the researchers due to unconstrained, skewed and varying writing styles. To get proper zonal information, a mask based approach has been considered here where mask size is $h \times w$. For implementation, w is set as Wd and h is varied based on average vertical run length of object (say, RLD) pixels and background (say, $RLND$) pixels of B. The entire process is described in Algorithm 5.1. Estimated mask along with the zone boundaries are depicted in Fig. 5.11 for the sample word images of Table 5.4.

Algorithm 5.1 Estimation of $\mathrm{R2}$ and $R4$
Input: Input image, here B
Output: $\mathrm{R2}$ and $R4$

Step 1. *//Estimation of mask height h*

Say, $\mathrm{i^{th}}$ $(1 \leq \mathrm{i} \leq \mathrm{Wd})$ column contains $\mathrm{N_i}(\geq 0)$ occurrences of continuous object pixels *with length* $L_{i1}, L_{i2}, \dots, L_{iN_i}$ in B.

Estimate $RLD = \frac{1}{Wd}\sum_{i=1}^{Wd} \max_{j=1,2,\dots,N_j}\{L_{ij}\}$

Let i^{th} ($1 \leq i \leq Wd$*)* column contains $\mathrm{N'_i}(\geq 0)$ occurrences of continuous background pixels with length $\mathrm{L'_{i1}, L'_{i2}, \dots, L'_{iN_i}}$.

Measure $RLND = \frac{1}{Wd}\sum_{i=1}^{Wd} \max_{j=1,2,\dots,N_j}\{L'_{ji}\}$

$h = \lfloor \lfloor RLD + 0.5 \rfloor + \lfloor RLND + 0.5 \rfloor + 0.5 \rfloor$.

Step 2. *// Estimation of $R2$*

Let Γ_i, $i = 1, 2, \dots, Ht - h$, is the maximum length of all horizontal runs of object pixels within mask i.

$\Gamma_i = \max_{j=1,2,\dots,h}\{\max_{k=1,2,\dots,\mathcal{C}_j}\{HL_{jk}\}\}$, where HL_{jk} and $\mathcal{C}_j$ are the length of k^{th} horizontal run and number of such runs in j^{th} row inside i^{th} mask ($i = 1, 2, \dots, Ht - h$) respectively.
$\delta = \frac{1}{Ht-h}\sum_{i=1}^{Ht-h}\Gamma_i$, where δ is mean value ofΓ_i's inside i^{th}mask ($i = 1, 2, \dots, Ht - h$) and is considered here as criteria parameter for selecting mask containing $R2$.

$\eta = \min_{\mathrm{i}}\{\mathrm{i}: \Gamma_\mathrm{i} \geq \delta\}$, where η is the starting row index of selected mask.
$R2 = \left\lfloor \frac{(\eta + h)}{2} + 0.5 \right\rfloor$.

Step 3. *//Estimation of $R4$*

Let Δ_i, $i = 1, 2, \dots, Ht - h$, is the maximum of all horizontal transition counts within mask i.

Now, $\Delta_i = \max_{j=1,2,\dots,h}\{\mathcal{TC}_j\}$, where $\mathcal{TC}_j$ is the transition (i.e., changeover between object and background pixels and vice versa) point counts ($\mathcal{TC}$s) in horizontal direction in j^{th} row inside i^{th} mask ($i \in [1, Ht - h]$).

$\delta' = \frac{1}{Ht-h}\sum_{i=1}^{Ht-h}\Delta_i$, where δ' is mean value of Δ_i's inside i^{th} mask ($i \in [1, Ht - h]$) and is considered here as parameter for selecting mask containing $R4$.
$\eta' = \max_{\mathrm{i}}\{\mathrm{i}: \Delta_\mathrm{i} \leq \delta'\}$, where η' is the starting row index of selected mask.

$$\mathrm{R4} = \left\lfloor \frac{(\eta' + \mathrm{h})}{2} + 0.5 \right\rfloor$$

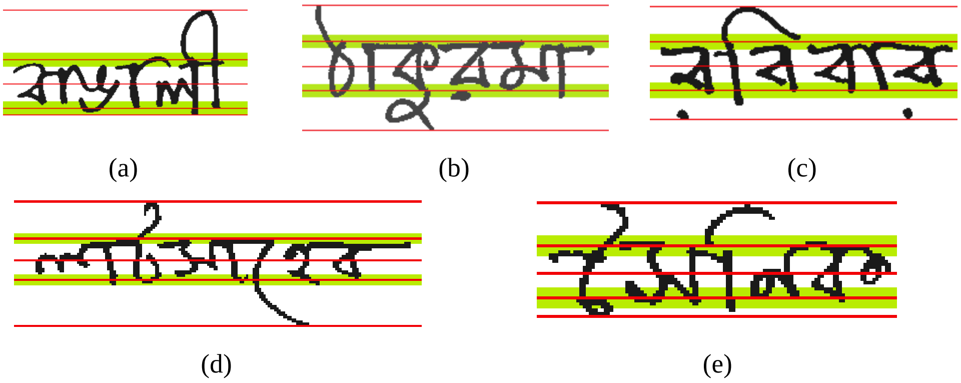


Fig. 5.11. Illustration of zone boundaries detection of the input images shown in Table 5.4. In these images red colored straight lines from top to bottom are R1, R2, R3, R4 and R5 respectively. Whereas, green colored rectangular shapes indicate the mask positions that contain R2 and R4 that are considered in this work as zone separating lines.

5.2.3 Determination of *Matra* Pixels

The common headline or *Matra* of a connected word segment may be identified as the continuous horizontal stripe of black pixels appearing at the top of most of the characters and some of modified shapes in the word segment. In a cursive handwriting, the appearance of a *Matra* is often disjoint and wavy. In addition to this, generally, the boundary between the sets of *Matra* pixels and non-*Matra* pixels in the region R1-R3 is not distinct in handwritten words. These issues make the identification of potential *Matra* pixels a challenging task. Considering the problems related to determining *Matra* pixels, here, a fuzzy measure of a pixel is obtained by multiplying its horizontalness value and fuzzy membership value. The horizontalness value of a pixel is determined by number of object pixels that appear to its left and/or right in continuous way. For estimation of membership value of a pixel that points to its potential belongingness to *Matra* zone a trapezoidal fuzzy membership function ($\mu_{\mathcal{TZ}}$) [81] is chosen.

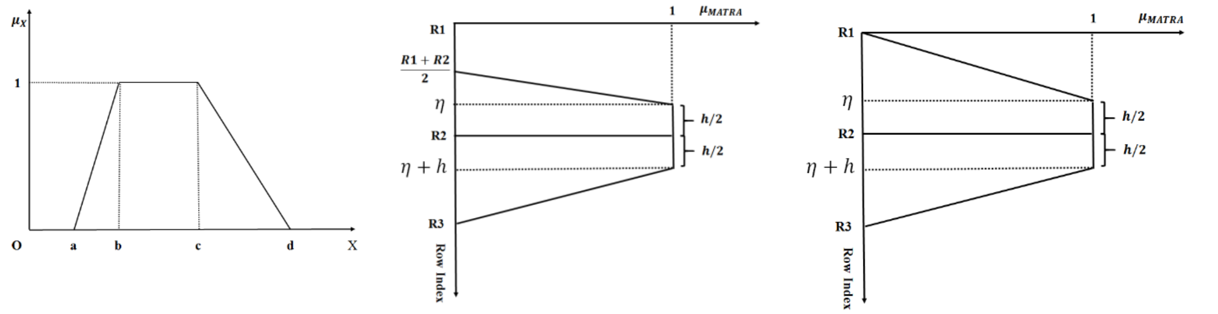
The exact expression of $\mu_{\mathcal{TZ}}$ is shown in eq. (5.1) while Fig. 5.12(a) depicts its geometrical representation. In this equation a, b, c and d ($a < b < c < d$) represent lower limit, lower support limit, upper support limit and upper limit of the membership function respectively. However, to fit this fuzzy membership function for the said problem, value of b, c and d are set as η, η + h and $R3$ respectively. Description of η and h are provided in Algorithm 5.1. Value of the parameter a is set following rule which is mentioned in eq. (5.2). Such choice of parameter values is made to assign maximum fuzzy membership value (i.e., 1) to all black pixel that lie within a certain distance (here, $h/2$) from the line R2. As pixels move away on both sides from boundaries of the region (i.e., [η, η + h]), their degree of belongingness to the set of *Matra* pixels should diminish. Notation of $\mu_{\mathcal{T}\mathcal{Z}}$ which is designed following the said specification is termed as μ_{MATRA} and the geometrical representation this fuzzy membership functions is shown in Fig. 5.12(b-c).

$$\mu_{\mathcal{T}\mathcal{A}}(x) = \begin{cases} 0, & x < a \\ \dfrac{x-a}{b-a}, & a \le x < b \\ 1, & b \le x \le c \\ \dfrac{d-x}{d-c}, & c < x \le b \\ 0, & x > d \end{cases} \tag{5.1}$$

$$a = \begin{cases} \dfrac{R1+R2}{2}, & \dfrac{R2-R1}{R4-R2} > \Delta \\ R1, & otherwise \end{cases} \tag{5.2}$$

Here Δ is predefined threshold value which is set here as 0.2.

Finally to determine *Matra* pixels in the region R1-R3, the product of the horizontalness value and fuzzy belongingness value of all object pixels within the said region are computed. Then mean of all such product values (say, avg) is estimated. Finally, an object pixel is considered as *Matra* pixel if its fuzzy measure exceeds avg value. All such *Matra* pixels constitute the *Matra* region. Sample images containing detected *Matra* pixels using the present method are shown in Fig. 5.13 while the original word images are shown in Table 5.4.



(a) (b) (c)

Fig. 5.12 The different membership functions (i.e., $\mu_{\mathcal{T}\mathcal{Z}}$) for detecting the set of *Matra* pixels: (a) represents the $\mu_{\mathcal{T}\mathcal{Z}}$ as defined in eq. (5.1), (b) and (c) variations in $\mu_{\mathcal{T}\mathcal{Z}}$ depending on alternative values of 'a' in eq. 5.2.

5.2.4 Determination of Potential Segmentation Points

Potential segmentation points are *Matra* pixels across which a word image is to be fragmented vertically in order to generate character/character sub-parts. Ideally these points should lie on the column positions along which the number of object pixels within the rows R2 and R3 and distance of the farthest object pixel (within the rows R2 and R3) from R2 are less. To earmark the degree of belongingness of a *Matra* pixel to the class of potential segmentation points, here a bell-shaped fuzzy membership function ($\mu_{\mathcal{BS}}$) [46] is chosen. The exact expression of generalized bell-shaped fuzzy membership function is shown in eq. (5.3) and its graphical representation is depicted in Fig. 5.14(a).

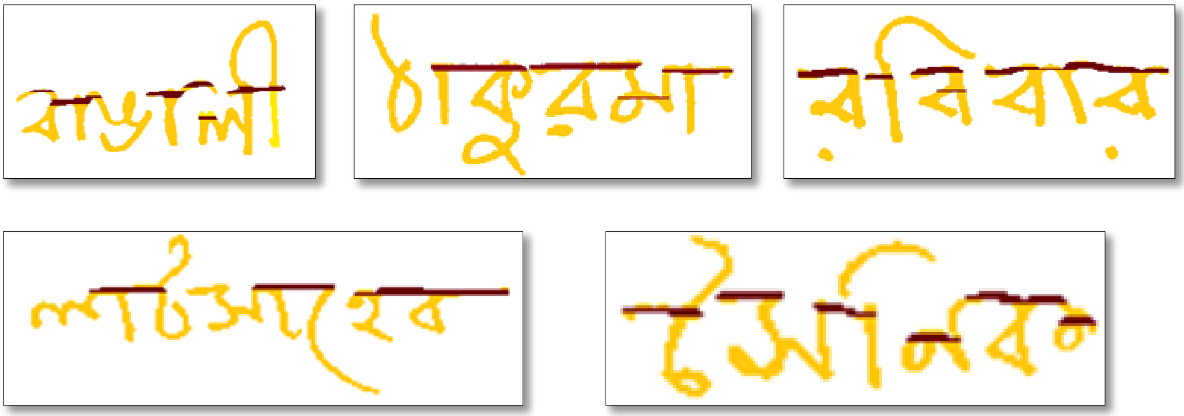


Fig. 5.13 Detected *Matra* pixels (red color pixels) of the input images shown in Table 5.4 are depicted.

$$\mu_{\mathcal{BS}}(x) = \frac{1}{1 + \left|\frac{x - c}{a}\right|^{2b}} \tag{5.3}$$

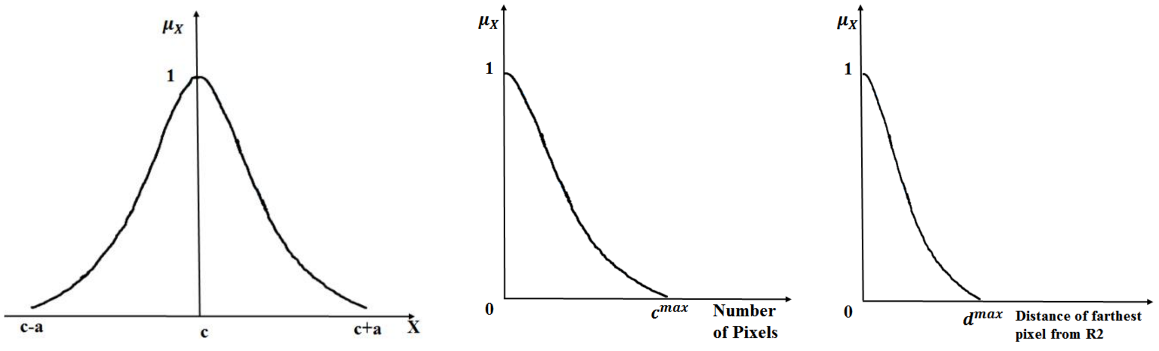
In this equation (i.e., eq. (5.3)), a, b and c are controlling parameters for deciding nature of $\mu_{\mathcal{BS}}$ which attains maximum value at $x = c$. To determine the fuzzy membership value of each of the columns in a word image this function is used here by suitably setting the parameter values. The values b are c are set as 1 and 0 respectively. Whereas the value of a is set as either maximum number of column-wise object pixel (say, c^{max}) or maximum of distances of farthest object pixel in each of the columns from R2 (say, d^{max}). Based on the choices of the parameter a, two bell-shaped fuzzy membership functions are defined. These fuzzy membership functions are termed as $\mu_1(x)$ and $\mu_2(x)$ which are formulated (considering pre-set values of b and c) in eqs. 5.4 and 5.5 respectively. Also, their graphical representations are shown in Fig. 5.14(b-c).

$$\mu_1(pc(x)) = \frac{1}{1 + \left|\frac{pc(x)}{c^{max}}\right|^{2}} \tag{5.4}$$

$$\mu_2(dfp(x)) = \frac{1}{1 + \left|\frac{dfp(x)}{d^{max}}\right|^{2}} \tag{5.5}$$

In eqs. 5.4 and 5.5, $pc(x)$ and $dfp(x)$ represent pixel count and distance of farthest object pixel from R2 respectively and x represents index of column under consideration.

To decide about whether a *Matra* pixel would be a potential segmentation point, the average of all $\mu_1(pc(x))$ and $\mu_2(dfp(x))$ values for all the columns containing at least one *Matra* pixel is computed. Let the average value is avg. If the average fuzzy membership value (i.e., $(\mu_1(pc(x)) + \mu_2(dfp(x)) * 0.5)$) of some column containing *Matra* pixel exceeds avg then all the *Matra* pixels belong to the column under consideration is marked as potential segmentation point(s). Sample outcomes containing estimated potential segmentation points, marked using red color, are shown in Fig. 5.15.



(a) (b) (c)

Fig. 5.14 The membership functions for determination of potential segmentation points (a) represents the $\mu_{\mathcal{BS}}$ as defined in eq. (5.3), (b) represents $\mu_1(pc(x))$ as defined in eq. 5.4 and (c) represents $\mu_2(dfp(x))$ as defined in eq. 5.5

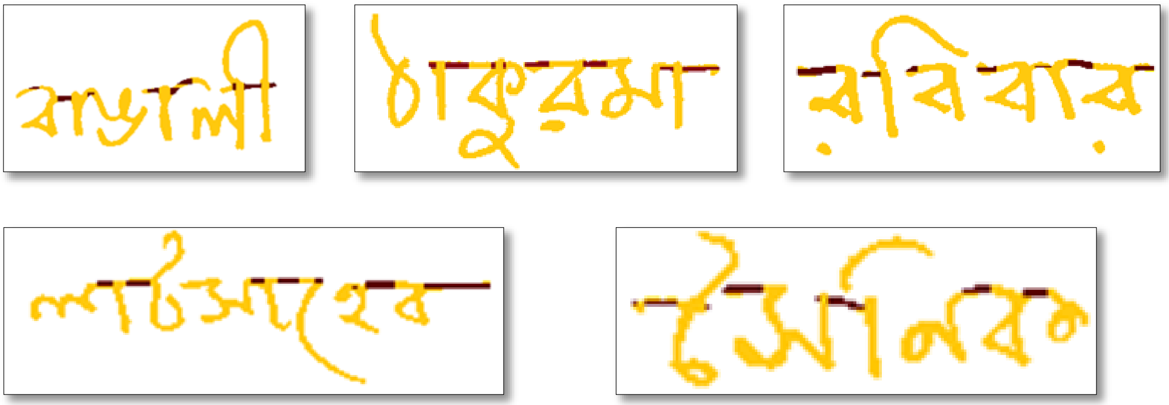


Fig. 5.15 Depiction of detected segmentation points (red color pixels), which are generated after confirmation of Matra pixels as segmentation pixels, of the images shown in Fig. 5.13.

5.2.5 Identification of Actual Segmentation Columns

Selection of actual segmentation points which accurately segment the word images into their constituent characters or character sub-parts is a challenging issue. Poor selection of these points leads to over and/or under segmentation during the segmentation process. As a result of these, characters of their sub-parts may get broken/segmented, leading to loss of information [46]. Therefore, for determination of actual segmentation points, there is always a trade-off between under/over segmentation of word images. In this thesis, an attempt is made to optimize between under and over segmentation, with minimum loss of object pixels, which is carried out by selecting a single column for segmentation on the *Matra* region. The methodology is described as follows.

To do this, cluster of segmentation points is identified using 8-neighbors CCL algorithm [115]. In Fig. 5.16(a-b) two samples of identified clusters are shown. Next, on each of these segmentation clusters two primary decisions have been taken. These decisions are related to selection of horizontal region for segmentation along specific columns on the *Matra* region and identification of the segmentation columns in each segmentation-cluster. This process is carried out using the following steps.

1. Check whether there is any ascendant in the word image under consideration for which height of upper zone of the word image is estimated. If the height of the said zone is exceeding threshold value $(0.2 * (R4 - R2))$, then it can be said that the word image has ascendant part(s) in the upper zone.
2. For each cluster, the following technique is applied to determine the segmentation column along which we can segment the word component under consideration:
   A. If there is no ascendant in the word component under consideration; calculate the sum of number of object pixels, *Matra* pixels and segmentation-point pixels for each column in the region from R1 to $\frac{(R3-R2)}{2}$. Otherwise, calculate the same for each column in the region from $\frac{(R2-R1)}{2}$ to $(R2 + \frac{(R3-R2)}{2})$.
   B. Consider the column for segmentation within the estimated region (row boundaries), which has the minimum sum, as calculated in step A.

Sample outcomes containing estimated segmentation columns, marked using red color, are shown in Fig. 5.17. Also the extracted components after segmentation through estimated segmentation columns are shown in Fig. 5.18.

5.2.6 Confirmation of Segmentation Lines

The segmentation mechanism, described above, still suffers from over segmentation error (see Fig. 5.18). In this stage, some over segmented parts of character or *Matra* component(s) are joined through cancelation of segmentation column(s) and redefining new ones. For this, histogram of object pixel count ($h(i)$) from $R2$ to $R3$ is calculated in vertical direction i.e., $h(i) = |\{i: f(j,i) =' 1'\}|$ where, $1 \le i \le Wd$ and $R2 \le j \le R3$. Then 1st order derivative ($dh(i)$) of $h(i)$ is calculated to get peak and valley of the histogram. $dh(i) = h(i+1) - h(i)$, $1 \le i \le Wd - 1$. Finally, following two restrictions are introduced to handle over segmentation issues.

i) A segmentation line is nullified if it does not enclosed by two peaks in $h(i)$.

ii) If multiple segmentation lines(say, $L_1, L_2, \ldots, L_N$) have appeared between two consecutive peaks, then a new segmentation line (SL_{New}) is defined as$SL_{New} = \frac{1}{N}\sum_{i=1}^{N} L_i$

Result after this correction is depicted in Fig. 5.19.

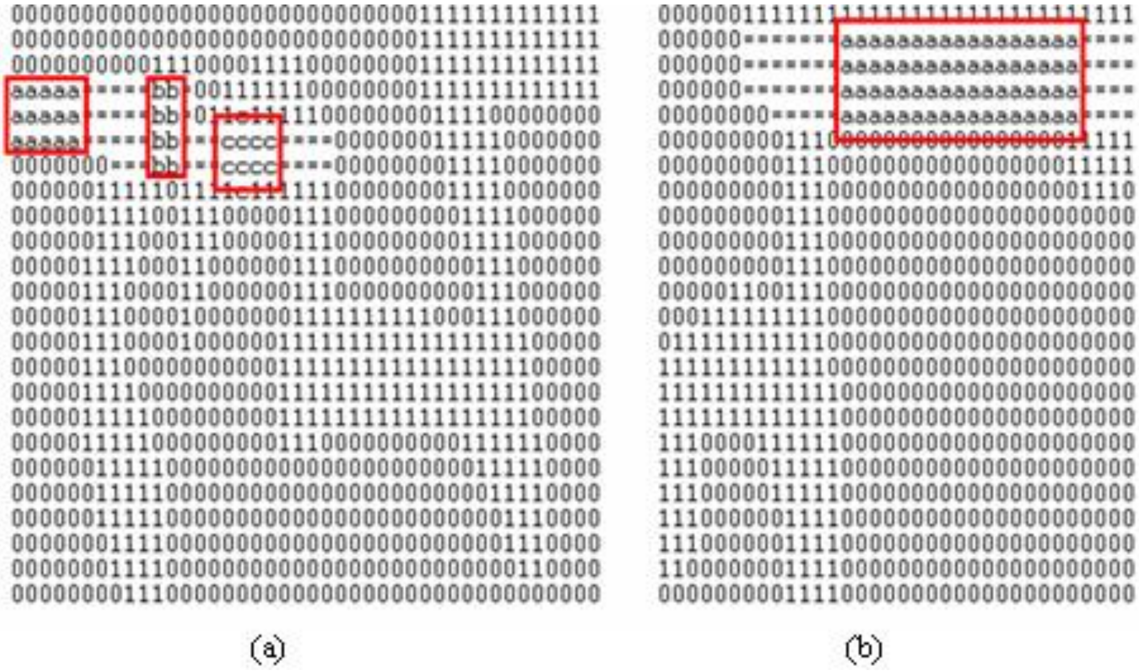


Fig. 5.16 Examples of segmentation point clusters. Here three segmentation clusters in (a) but only one segmentation cluster in (b). The characters ‘a’, ‘b’ and ‘c’ indicate segmentation point label while the symbol ‘=’ indicate *Matra* pixels [83].

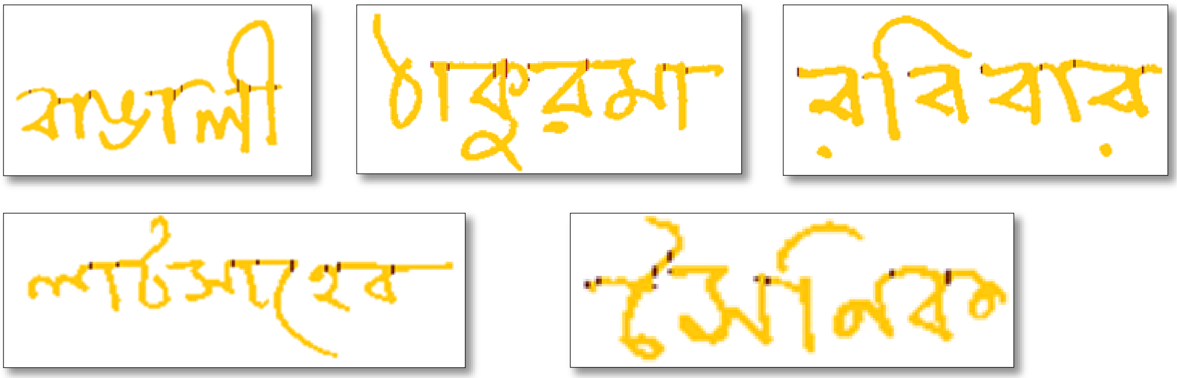


Fig. 5.17 Showing detected segmentation columns (red color pixels) of the input images shown in Table 5.4

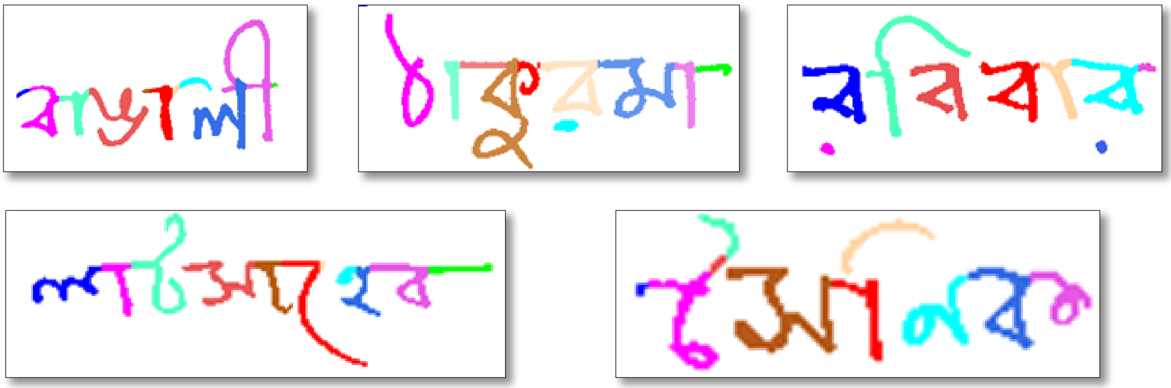


Fig. 5.18 Depiction of extracted characters or character like shapes, which are generated after segmenting through segmentation lines, of the word images shown in Fig. 5.17. Here, each different color indicates the extracted character or character like shape

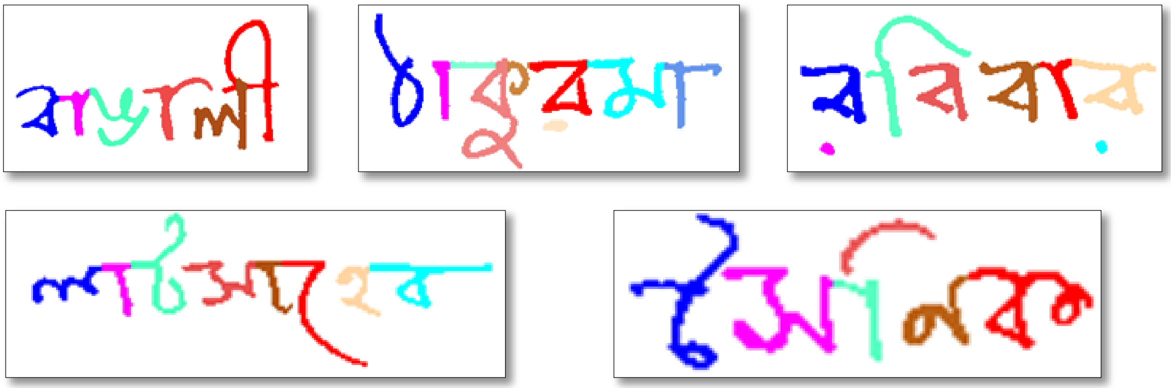


Fig. 5.19 Showing characters or character like shapes, which are generated after confirmation of segmentation lines, of the word images shown in Fig. 5.18. Here, each different color indicates the extracted character or character like shape

5.2.7 Separation of Lower Zone Components

All the components, appearing below $R4$ (ideally), are considered as lower zone components. But, it is frequently found that lower zone components are written along with their corresponding middle zone component and form a single component. Therefore, it is essential to separate these lower zone components from middle zone component. To accomplish this, first, middle zone CCs containing lower zone component or extended middle zone component (EMZC) are selected and then lower zone component is separated out from middle zone component.

To select such component a procedure called selection of word with dominant lower zone (DLZ) is performed based on vertical spread ($VS \in (0,1]$) of lower zone in the word.

$$VS = \frac{R5 - R4}{R5 - R1}$$

A word contains a DLZ if $VS > dP$, where dP is a decision parameter (here, it is experimentally set to 0.2).

Let $C_1, C_2, \ldots, C_N$ are N numbers of CCs generated after segmenting a word image (B) with DLZ i.e., $\bigcup_{i=1,2,\ldots,N} C_i = B$and $C_i \cap C_j = \emptyset, i \neq j$. These CCs are grouped into three categories *viz.*, (i) entirely a middle and/or upper zone CC, (ii) entirely a lower zone CC and (iii) EMZC or lower zone component connected with middle zone component. These grouping of CCs are carried out by checking vertical belongingness of CCs into different zones. The first two categories of CCs are skipped from lower zone separation.

Let SR_i and ER_i are the start row and end row of C_i, $i = 1, 2, \ldots, N$. Information about rows is estimated by scanning the components in raster order from top to bottom and bottom to top. The classification rule is defined as

$$C_i \begin{cases} \in category\ (i), if\ ER_i \leq [(R4 + R5)/2] \\ \in category\ (ii), if\ SR_i \geq [(R3 + R4)/2] \\ \in category\ (iii), Otehrwise \end{cases}$$

It is worth mentioning that EMZC (CC belonging to category (iii)) should be skipped from separation through R4. The detection of EMZC is performed by calculating horizontal average transition point within $R4$ to $R5$. Final results of the entire segmentation process are shown in Fig. 5.20.

5.3 Ground Truth Preparation

It has already been mentioned that availability of GT image makes an image-based database more useful as it helps in assessing one's algorithm. Generation of appropriate GT image, however, is always a challenging and tiresome task. Moreover, a fast and accurate evaluation of an algorithm is not possible without proper GT images and associated evaluation methodology. This evaluation is unbiased and human-error free which may found during assessment in manual mode.

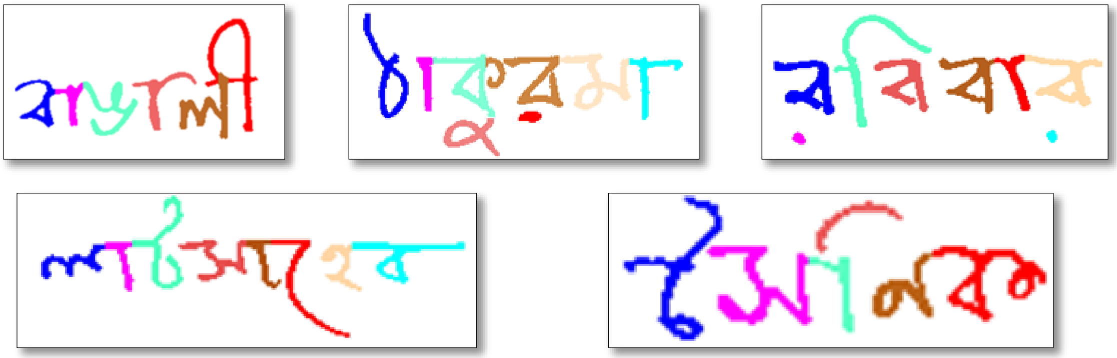


Fig. 5.20 Depiction of extracted characters or character like shapes, which are generated after applying lower zone separation technique on the input images shown in Table 5.4. Here, each different color indicates the extracted character or character like shape

Like previous chapters, GT image of a word sample is prepared here in a semi-automatic way. First, word images are segmented into constituent character(s) and modified shape(s) using the segmentation mechanism described in above section. This segmentation method is hereafter called as *core segmentation* methodology. Finally, all the CCs, obtained from a segmented word using 8-way CCL algorithm, are uniquely colored. Ideally all these CCs should be either a complete character or character shape, but this may not be true while segmenting handwritten word. Hence, it can be stated that the said handwritten word segmentation process suffers from anomalies due to following reasons:

- Failure in generating true segmentation points along *Matra* region (see Fig. 5.21(a)) or separating lower zone components (see Fig. 5.21(b)),
- Presence of touching characters within word image (see Fig. 5.21(c))
- Generation of segmentation points on isolated character and/or character sub-part (see Fig. 5.21(d)) or multiple segmentation lines on *Matra* region between two consecutive character shapes (see Fig. 5.21(e))

- Erroneous *Matra* zone detection (see Fig. 5.21(f)).

All these possible errors, produced during automated character segmentation process are then manually corrected using the Microsoft paint tool. Few examples of auto-generated erroneous samples and their corrected versions are displayed in Fig. 5.21(g-l). These corrected and error-free word samples are considered as GT images and included in the present database. It is noteworthy that inherently segmented CCs (part of character / modified shape) are kept isolated (see Fig. 5.22(a-c)) therein. Fig. 5.22 (a-b) shows some inherently segmented part(s) of a character and Fig. 5.22(c) shows the parts of a modified shape in GT images. The GT images of the sample words shown in Table 5.4 are shown in Table 5.5. This table also includes the final outcome of the current segmentation process (see Fig. 5.20).

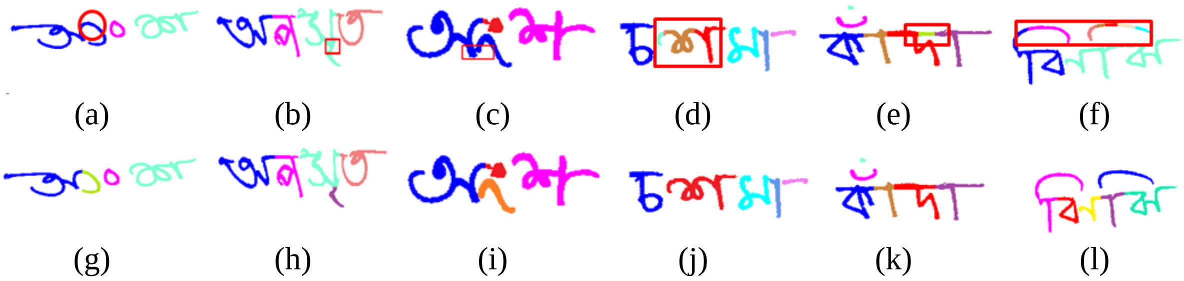


Fig. 5.21 Examples of different errors (a-f) those need manual correction for GT preparation and final GT images after manual correction (g-l)

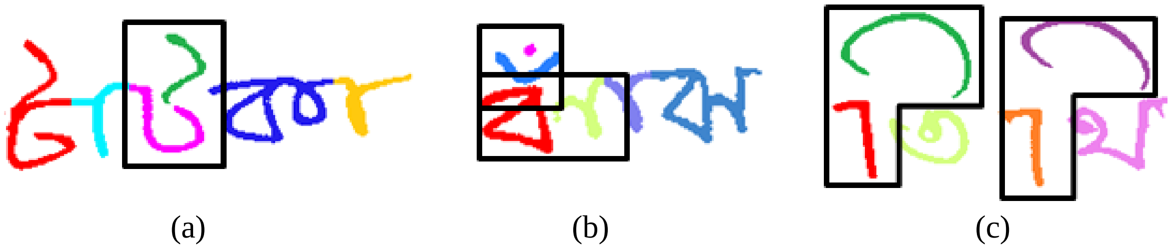
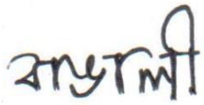
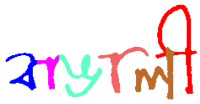
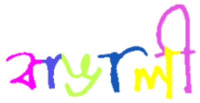
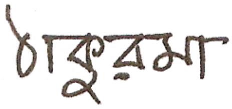
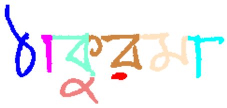
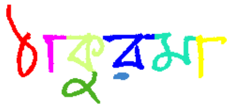
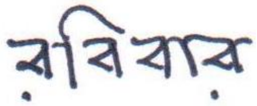
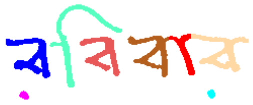
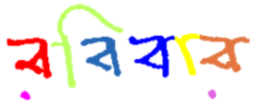
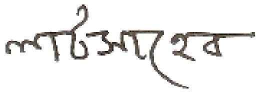
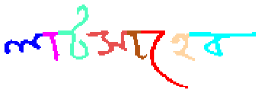
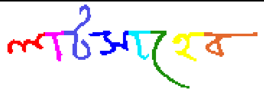
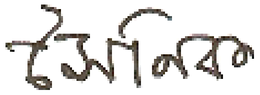
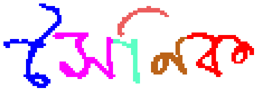
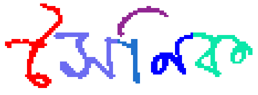


Fig. 5.22 Some samples containing inherently segmented part of character or modified shape

5.4 Experimental Results

It has already been mentioned that in this chapter an improved technique for character extraction from isolated handwritten *Bangla* word images (see section 5.2) is described. It also includes a database containing handwritten isolated word images along with corresponding GT images. These GT images represent ideally segmented CCs and these images have been used for automatic assessing of any character extraction algorithm. For automatic assessing, the performance evaluation technique, described in section 3.4.2, is used. It is noteworthy that for fitting the said evaluation strategy for performance measure of a character extraction process every character/character sub-parts in GT image and resultant image are considered as component.

Table 5.5 Five samples from the database with their character extracted image and GT image. Here each extracted/ideal character is uniquely colored

Sl. #	Word Sample	Segmented Image	GT Image
1			
2			
3			
4			
5			

5.4.1 Performance Evaluation

It has already been mentioned in previous chapters that the evaluation tool, prepared during this thesis work, provides the performance measures in terms of TP, FN, FP, recall, precision and F-measure for text line (TL), word and character extraction methods. These statistics are prepared by a bijective mapping from a segmented word image into its GT image. Using this multi-purpose evaluation tool to access the performance of current work, GT images representing ideal segmented component(s) are prepared in this chapter. Whereas for fitting the same tool, character(s)/sub-character(s) of the resultant and GT images are considered as component. The performance on five randomly selected sample word images from the said database are shown in Table 5.5. The table contains both the GT image and resultant image (i.e., output of present character extraction technique). Also, for better insight into the performance of the designed segmentation method some statistical measures are also shown in Table 5.6.

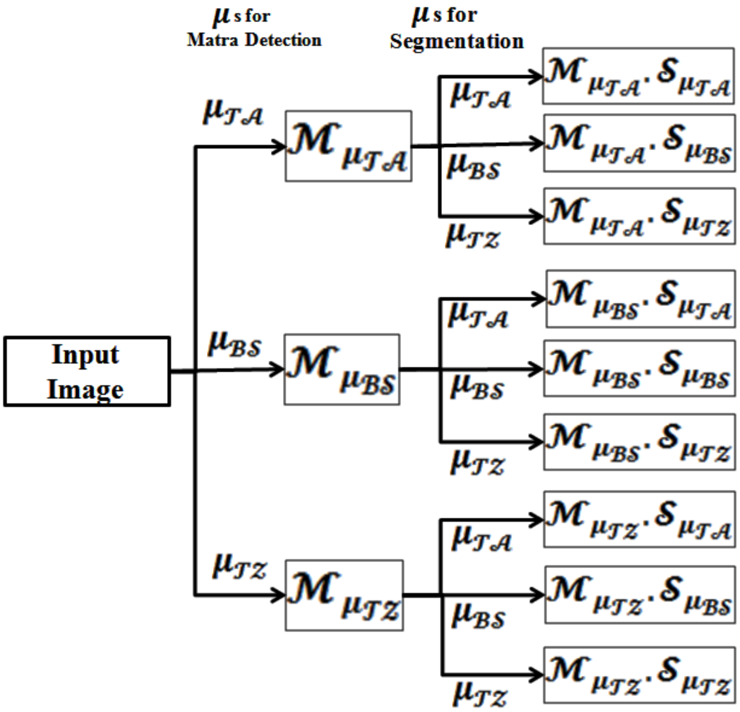

μs for Matra Detection
μs for Segmentation
Input Image
$\mu_{\mathcal{TA}}$
$\mu_{\mathcal{BS}}$
$\mu_{\mathcal{TZ}}$
$\mathcal{M}_{\mu_{\mathcal{TA}}}$
$\mathcal{M}_{\mu_{\mathcal{BS}}}$
$\mathcal{M}_{\mu_{\mathcal{TZ}}}$
$\mathcal{M}_{\mu_{\mathcal{TA}}}.\mathcal{S}_{\mu_{\mathcal{TA}}}$
$\mathcal{M}_{\mu_{\mathcal{TA}}}.\mathcal{S}_{\mu_{\mathcal{BS}}}$
$\mathcal{M}_{\mu_{\mathcal{TA}}}.\mathcal{S}_{\mu_{\mathcal{TZ}}}$
$\mathcal{M}_{\mu_{\mathcal{BS}}}.\mathcal{S}_{\mu_{\mathcal{TA}}}$
$\mathcal{M}_{\mu_{\mathcal{BS}}}.\mathcal{S}_{\mu_{\mathcal{BS}}}$
$\mathcal{M}_{\mu_{\mathcal{BS}}}.\mathcal{S}_{\mu_{\mathcal{TZ}}}$
$\mathcal{M}_{\mu_{\mathcal{TZ}}}.\mathcal{S}_{\mu_{\mathcal{TA}}}$
$\mathcal{M}_{\mu_{\mathcal{TZ}}}.\mathcal{S}_{\mu_{\mathcal{BS}}}$
$\mathcal{M}_{\mu_{\mathcal{TZ}}}.\mathcal{S}_{\mu_{\mathcal{TZ}}}$

Fig. 5.23 Possible combination of experimental setups used for *Matra* detection and confirming *Matra* pixels as segmentation points

Table 5.6 Performance of present character extraction algorithm on 5000 isolated word images prepared during this thesis work

Term	Minimum	Maximum	Average	SD	Q1	Q2	Q3	IQR
TP	0.2790	1.0000	0.8623	0.1215	0.7891	0.8908	0.9603	0.1716
FN	0.0000	0.4882	0.0528	0.0631	0.0077	0.0287	0.0790	0.0713
FP	0.0000	0.7187	0.0847	0.1064	0.0117	0.0364	0.1300	0.1183
Recall	0.4980	1.0000	0.9418	0.0692	0.9112	0.9684	0.9912	0.0801
Precision	0.2813	1.0000	0.9105	0.1108	0.8570	0.9610	0.9872	0.1302
F-measure	0.4362	1.0000	0.9212	0.0771	0.8821	0.9422	0.9798	0.0976

5.4.2 Variants of Fuzzy Function based Segmentation Technique

In this work, $\mu_{\mathcal{TZ}}$ and $\mu_{\mathcal{BS}}$ are used for detecting *Matra* region and segmentation points generation purpose. But, for these said purposes any of the fuzzy membership functions, such as $\mu_{\mathcal{TA}}$, $\mu_{\mathcal{BS}}$ and $\mu_{\mathcal{TZ}}$, can be used. Therefore, more experiments have been conducted considering all these said fuzzy membership functions. The experimental structure is shown in Fig. 5.23. In this figure, $\mathcal{M}_{\mu_1}.\mathcal{S}_{\mu_2}$ indicates that fuzzy membership functions (μ_1 and μ_2) have been used for *Matra* region detection and estimation of segmentation pixels respectively. The average measures of all these statistics for the entire database are tabulated in Table 5.7 for all the said experiments. The best TP, FN, FP, recall, precision and F-measure scores are highlighted in this Table. From the table it is clear $\mathcal{M}_{\mu_{\mathcal{TZ}}}.\mathcal{S}_{\mu_{\mathcal{BS}}}$ provides the best F-measure score which ensures the better trade-off between over, under and truly segmented components.

5.4.3 Analysis of Effects of Different Stages of Present Method

The present segmentation algorithm has several stages. As a result, an investigation is needed in order to provide the effects of the said stages on the entire process. Hence a set of experiments has been conducted on the entire database. The detail of this experiment is recorded in Table 5.8. Please note that the experimental results in this table are generated using the present segmentation mechanism.

Table 5.7 Average performances on entire database by all experimental protocols shown in Fig. 5.23

Experiment ID	**TP**	**FN**	**FP**	**Recall**	**Precision**	**F-measure**
$\mathcal{M}_{\mu_{\mathcal{TA}}}.\mathcal{S}_{\mu_{\mathcal{TA}}}$	0.8611	**0.0503**	0.0886	**0.9445**	0.9066	0.9203
$\mathcal{M}_{\mu_{\mathcal{TA}}}.\mathcal{S}_{\mu_{\mathcal{BS}}}$	0.8584	0.0514	0.0902	0.9433	0.9050	0.9187
$\mathcal{M}_{\mu_{\mathcal{TA}}}.\mathcal{S}_{\mu_{\mathcal{BS}}}$	0.8527	0.0584	0.0888	0.9355	0.9056	0.9154
$\mathcal{M}_{\mu_{\mathcal{BS}}}.\mathcal{S}_{\mu_{\mathcal{TA}}}$	0.8521	0.0600	0.0879	0.9339	0.9064	0.9150
$\mathcal{M}_{\mu_{\mathcal{BS}}}.\mathcal{S}_{\mu_{\mathcal{BS}}}$	0.8598	0.0545	0.0857	0.9403	0.9094	0.9196
$\mathcal{M}_{\mu_{\mathcal{BS}}}.\mathcal{S}_{\mu_{\mathcal{TZ}}}$	0.8543	0.0618	0.0839	0.9322	0.9104	0.9164
$\mathcal{M}_{\mu_{\mathcal{TZ}}}.\mathcal{S}_{\mu_{\mathcal{TA}}}$	0.8610	0.05091	0.0881	0.9438	0.9071	0.9202
$\boldsymbol{\mathcal{M}_{\mu_{\mathcal{TZ}}}.\mathcal{S}_{\mu_{\mathcal{BS}}}}$	**0.8623**	0.0528	**0.0847**	0.9418	**0.9105**	**0.9212**
$\mathcal{M}_{\mu_{\mathcal{TZ}}}.\mathcal{S}_{\mu_{\mathcal{TZ}}}$	0.8573	0.0535	0.0892	0.9411	0.9059	0.9180

Table 5.8 Performance of different stages of the present character extraction method

Character extraction stage	**TP**	**FN**	**FP**	**Recall**	**Precision**	**F-measure**
No segmentation	0.6624	0.0000	0.3376	1.0000	0.6624	0.7751
Segmentation lines	0.8004	0.1300	0.0696	0.8603	0.9209	0.8848
Confirmed segmentation lines	0.8426	0.0775	0.0799	0.9151	0.9133	0.9097
lower zone separation	0.8623	0.0530	0.0847	0.9418	0.9105	**0.9212**

5.4.4 Comparison with *State-of-the-art* Mechanisms

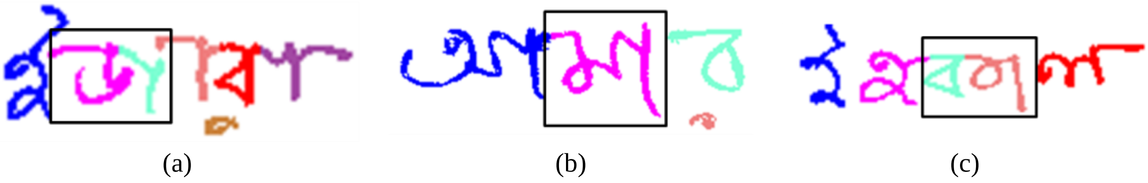
A number of works, described in section 5.1.2, are found in literature which are used to segment handwritten/printed isolated word images. Of these, the works [60, 81-83, 150-151] have applied fuzzy membership function for detection of *Matra* pixels as well as to confirm *Matra* pixels as segmentation points. Therefore, the core character segmentation module of the works [60, 81, 151] have been implemented here. Also, the effects on overall performances of these techniques while the current modifications are incorporated therein are included with the actual ones. The detail information of these experiments is summarized in Table 5.9 along with the average F-measure score.

Table 5.9 Comparison with *state*-of-*the-art* methods using f-measure score

Method	**Original**	**After proposed modification**			
		Zone boundaries detection	**Segmentation line generation**	**Segmentation line confirmation**	**Lower zone component separation**
Basu et.al. [60]	0.8291	0.8573	0.8836	0.9093	0.9203
Sarkar et. al. [151]	0.8326	0.8547	0.8817	0.9069	0.9208
Singh et. al. [81]	0.8609	0.8720	0.8720	0.8986	0.9158
Present Method	**0.9212 (Overall)**				

5.4.5 Error Case Analysis

From the above results, it can be *safely* commented that the present character segmentation method provides comparatively better result. In spite of this success, the present technique suffers from segmentation errors either in form of over segmentation or in form of under segmentation error. Under segmentation error mostly occurs for touching components within a word while over segmentation error occurs due to presence of extended characters part in the middle zone. Few outcomes of the under and/or over segmented word samples are shown in Fig. 5.24(a-c). In this figures, (a) and (b) show output images that contain error due to over and under segmentation respectively while (c) indicates an error case where both the under and over segmentation errors occurred. The erroneous regions are marked using rectangular boxes in these images.



(a) (b) (c)

Fig. 5.24 Instances of over and/or under segmented words (a) shows example of over segmentation error (b) depicts example of under segmentation error and (c) portrays example of under as well as over segmentation errors

5.6 Discussion

HWR is an important *and* an interesting research problem. Consideration of complex regional script like *Bangla* with lexicon-free handwritten words, which needs character segmentation, makes the problem even more challenging. In addition, unavailability of freely accessed databases is one of the reasons for the slow progress in this research area. Considering the above mentioned facts, in this chapter, a character extraction technique from handwritten *Bangla* word images is described. Also, to fill the shortfall of database, in this work, a database of isolated *Bangla* word images is prepared. Moreover, it is a fact that availability of GT images for this type of database would make it useful even to the researcher who is interested to work with the language s(he) does not know. GT images are prepared here in a semi-automated way. These GT images are demonstration of ideally segmented characters and modified shapes or their parts. Apart from these, the work under consideration proposes an evaluation methodology which would help the user to get rid of manual evaluation of their devised segmentation procedure. The said character extraction methodology is applied on the entire database and evaluated using the proposed evaluation mechanism. The result, as mentioned here, is 0.9212 (F-measure score) which is better than some *state-of-the-art* character extraction methods from handwritten isolated *Bangla* word images.

Chapter 6

HANDWRITTEN *BANGLA* WORD RECOGNITION: HOLISTIC APPROACH

6.1 Introduction

Automatic recognition of handwritten word images is one of the most popular research areas in the domain of document image processing. The reason of its popularity lies in its wide range of applications in human society which include postal automation [162-163], bank cheque processing [164-166], keyword searching [68-69, 71], form processing [94, 167-168], reading doctor's prescription [58, 169] etc. The main challenge in recognizing the handwritten word images is the varying writing styles of individuals. Even the script in which the words are written can pose additional challenges. Therefore, development of a comprehensive and accurate handwritten word recognition system is difficult and needs more attention from the researchers.

In the literature, two major approaches for word recognition are found *viz.*, segmentation based word recognition [15] and holistic word recognition [69, 164-166]. Since words are generally a sequence of characters, hence, a common trend followed by many researchers is to segment a word first and then recognize each segment individually in order to recognize the whole word [170]. This is called segmentation based word recognition. The most advantageous point about segmentation based approach is that it is lexicon free. But in most of the cases, it performs poorly for handwritten documents due to the ambiguity of the outcome of a character segmentation technique [171-172]. The low segmentation accuracy indicates the failure of segmentation algorithm to yield the ideal segmentation points on the handwritten word images resulting in under/over segmentation [83]. As a result, the OCR systems are forced to work with various combinations of erroneous characters or their components. Again, all possible combinations of valid characters or character subparts from over segmented characters lead to a large set of patterns to recognize. Thus, designing such system to recognize word(s) for unbounded lexicon is perhaps impossible.

In a research work stated in [173], it is found that human minds recognize words in holistic manner. This very fact encourages many researchers to apply holistic approach for word recognition [63] which removes the intricacy of word segmentation and focuses on the recognition of the word as a single unit. In addition to this, according to the authors of this work, holistic approach may succeed even when the writing style is too poor for identification of individual character boundary from the word, except it preserves the overall shape. Moreover, the authors of the work [174] have shown experimentally in their work that holistic approach performs better than its counterpart while considering limited and fixed sized lexicon. Hence, it is worth mentioning here that although the use of holistic approach is restricted to the problems with fixed or limited sized lexicon, yet, it could a better choice than its counterpart while recognizing fixed and pre-defined set of words like city names [162-163], month name [164], legal amount [165-166] and keywords [69, 71]. Therefore, considering these issues and one of the prime objectives of this thesis work (i.e., keyword searching from handwritten document images), handwritten word recognition using holistic approach is attempted here.

6.1.1 Previous Work

A number of contemporary works in literature [69, 162-166, 175-188] are found which have followed holistic approach for recognizing handwritten words. These works broadly followed two major approaches *namely*, language model independent [69, 165-166, 176-182] and language model dependent [162-164, 175, 183-188]. Techniques that follow first category of approaches use the entire word for extracting features and then the extracted features are fed into classifier to perform recognition. On the contrary, methods that fall under the second category, first perform implicit segmentation of the word images into character like shapes and then extract some features from each of the shapes to recognize them. Finally, words are recognized with the help of language model. Considering the architectural need, it is worth in mentioning that techniques that fall under first category require less time than other category of methods.

A prototype based recognition model is introduced in the work [165] for recognizing a set of legal amount words, extracted from low-quality French bank cheques, in holistic way. It has employed prototype based recognition model. In this work, first the words are clustered into 16 prototypes or models using the presence of upper, lower and middle zones. Finally, these prototype models are further classified using Hopfield network. The authors in the work [166] have proposed a template matching based technique for the same purpose. Here, first the word images are transformed using city-block distance based image transformation and then correlation coefficient are measured for performing word matching. While the methods, described in [176-177], have employed scale and rotation invariant M-band packet wavelet transformation and discrete cosine transformation (DCT) prior to feature extraction respectively. Mahalanobis distance between the transformed word images has been measured for matching distinct words in [176]. In [177], DCT coefficients of DCT transformed word image are fed to Support Vector Machine (SVM) classifier with radial basis function (RBF) kernel for classification need.

In the work [69], several structural (like projection, upper and lower profiles) and statistical (like punctuation count, ratio between punctuation and main connected parts) features are extracted from connected parts of word images. Next, a Multi-Layer Perceptron (MLP) based classifier has been used for preparing a learned module. The authors in the work [178], have described the word images in the feature space using Chebyshev moments and statistical and contour-based features (SCFs). Here, classification has been performed by combining three classifiers *namely,* SVM, MLP and Extreme Learning Machine (ELM). In another work [179], several perceptual features like reference lines, large gaps and extrema of local contours have been extracted from handwritten word images for automatic verification of street names that have been extracted from live US mail. In this case, a dynamic programming based approach has been used for matching purpose. To obtain discriminative stroke orientation in handwriting samples, Arnold transform followed by Hough transform has applied on the word images in the work [180]. Next these orientation information along with some naïve structural features (like pixel count and pixel density in overlapping windows) are extracted from a word image to recognize it holistically.

A ranking based classifier design technique is proposed in the work [181] for holistic word recognition. In this work, two different recognition models are used. The first one extracts gradient based, structural and cavity (GSC) features from word images and then fed them to binary vector matching (BVM) technique; while the second technique uses vertical projection profile (VPP) based features and dynamic time warping (DTW) based word matching protocol. Finally, highly matched words in both the approaches are considered for ranking based classifier design. In another work [182], Fourier transform on input word images has been applied prior to recognize them using a Convolutional Neural Network (CNN).

In the work [164], a character and/or character sub-parts based Hidden Markov Model (HMM) is proposed for holistically recognizing handwritten month names that are extracted from Brazilian cheque images. A classifier ensemble has been built in the work [183] for recognizing month names written in Portuguese. It approximates the word images into three different levels based on the human reading process. These approximations are carried out by introducing pseudo-segmenter which divides a word image vertically into i) 2 segments through cetroid, ii) 8 sub-images having equal width and iii) 10 segments having variable width which is decided by vertical pixel count. Next different perceptual features like count of semicircles and loops formed by contour, number of crossing-points on contour, position and size of ascendant(s) and descendant(s), concavity measure, ratio of foreground and object pixels and segment length are extracted from each of these segments. Finally, recognition outcomes from two Neural Networks (NN) and one HMM based classifiers are combined to obtain final recognition accuracy. In the work [184], a holistic word recognition system has been developed to recognize the handwritten Farsi/Arabic words using left-right discrete HMM and Self Organizing Feature Map (SOFM). Here, the chain code histograms of different stripes of the input image are used as feature vector. This method is evaluated on 17,000 handwritten word images containing 198 city names of Iran and achieved comparably good result. A continuous density HMM is proposed in the work [185] which replaces the discrete observation probabilities by a continuous probability density function (PDF) for recognizing a word image written in Devanagari script. In this work, same feature set as described in [184] has been used.

A character-based HMM model is proposed in [175, 186] for recognizing handwritten *Bangla* words. In these works, two-stage recognition scheme is used. Firstly, a given word image is segmented into three zones, *viz.*, upper, lower and middle and then each of the character like components in each of the said zones are recognized separately. Finally, recognition results are assembled to generate the recognized word. The problem of extraction of characters from the word images is solved to some extent by this approach, but it introduces zone-level segmentation which is error-prone for unconstrained handwritten word images. The authors in the work [187] have used a left-right HMM model to recognize handwritten city names written in *Bangla* script. In this work, a Genetic Algorithm (GA) based optimization technique has been employed while training the HMM. A shape based directional encoding feature has been extracted from pre-segmented word parts for recognition purpose. CNN with a recurrent model called Bidirectional Long Short-Term Memory Network (BLSTM) has been used for recognizing handwritten *Bangla* words in [188].

6.1.2 Motivation

It has already been mentioned in Chapter 1 that *Bangla* is the seventh most spoken language in the world with more than 260 million speakers worldwide. It is also the second most popular official language (out of 23 official languages) in India and the national language of Bangladesh. *Bangla* language is written using *Bangla* script which is again one of the popular script worldwide. Beyond *Bangla,* it is used to write languages like Assamese and occasionally Manipuri and Santhali. In spite of its popularity, Less number of research attempts [162-163, 175, 186-187] for handwritten *Bangla* word recognition are found in literature in comparison to recognition of handwritten samples written in Arabic [69, 176-178, 182, 184], Roman [71, 166, 170-172, 174, 179-180], Devanagari [181, 185], French [165], Brazilian [164] and Portuguese [183] scripts. These systems have recognized the word images with the help of associate language model. Hence, these methods take relatively higher processing time than the techniques following the alternative approach. Considering these issues, during this thesis work several attempts (refer to [26, 73, 84-85, 88, 189-193]) are made for recognizing handwritten *Bangla* word images in language model independent way.

In addition, irrespective of script and recognition protocol, authors of the above mentioned researches have mostly concentrated on designing new features to improve the recognition accuracy. In turn, this scenario increases the dimensionality of the feature vector under consideration and these works have not considered any technique to justify the essence of these newly designed features for recognizing the pattern classes under consideration. For example these techniques have not tried any feature selection (FS) technique which can also be helpful in reducing computational needs. To bridge this research gap, in this thesis work, attempts have been made (refer to [88, 190, 193]) to apply FS model for the said task.

Moreover, most of the works, described in [69, 164-166, 176-185] have been experimented on databases that are freely available to the research community. Whereas, the works on *Bangla* script [15, 162-163, 175, 186-187] are, in general, experimented on in-house databases. This is also a major drawback for conducting research on handwritten *Bangla* word recognition in a holistic way. Hence, during this research work, a database containing 18000 isolated word samples has been prepared and made freely available to research community [73]. It consists of 120 most popular city names of West Bengal, a state of India. Each of the city names have 150 different handwritten instances.

6.1.3 Objective of the Chapter

Based on the above discussion, the following objectives have been covered in this chapter:

1. Designing a feature set which can recognize handwritten *Bangla* words holistically in language model independent way.
2. Proposing a hierarchical feature selection (HFS) model for selecting an optimal feature subset which represents the dataset in a better way than the original one.
3. Developing a comprehensive database for performing handwritten *Bangla* word recognition holistically as well as using analytical approach.

6.2 Holistic Word Recognition

In the present work, a holistic handwritten word recognition technique in language model independent way is designed. For this purpose, three different feature sets *namely*, elliptical [73, 84], gradient based [85, 189] and topological [71, 86] features that describe shape, texture and spatial and/or geometrical properties of patterns respectively are extracted from each of the handwritten word samples. Next, a HFS model [88] is introduced for obtaining an optimized feature set which describes the data in a better way. In every stage of designed HFS model, Memetic Algorithm (MA) [194] has been used for selecting optimized feature sets. Finally, all the selected features are used for performing recognition of handwritten words. Handwritten *Bangla* city name database [195] which is prepared during this thesis work has been used for this experimental purpose. The overall working procedure of feature extraction and optimization is illustrated in Fig. 6.1. In the following subsections first MA based FS model and feature extraction method are described and then HFS model is drafted.

6.2.1 MA based FS Model

One of the major pre-requisites in solving any pattern recognition problem is to design a befitting feature vector that can uniquely represent the patterns in the feature space. Therefore, a large number of researchers over the years have devoted themselves to fix up various pattern classification problems by introducing several new/modified feature vectors based on shape, texture or geometry of the patterns. As a result, designing more and more features has become a common trend. But, increased dimension of feature vectors may not always provide better outcome, as generation of huge number of features does not ensure the orthogonal property of features in the feature space. The key reason for this is that feature values may provide contradictory and irrelevant information. Another problem of the high dimensional feature vector is the increased time requirement to build a recognition module for classifying that data since this time is directly proportional to the feature dimension under consideration.

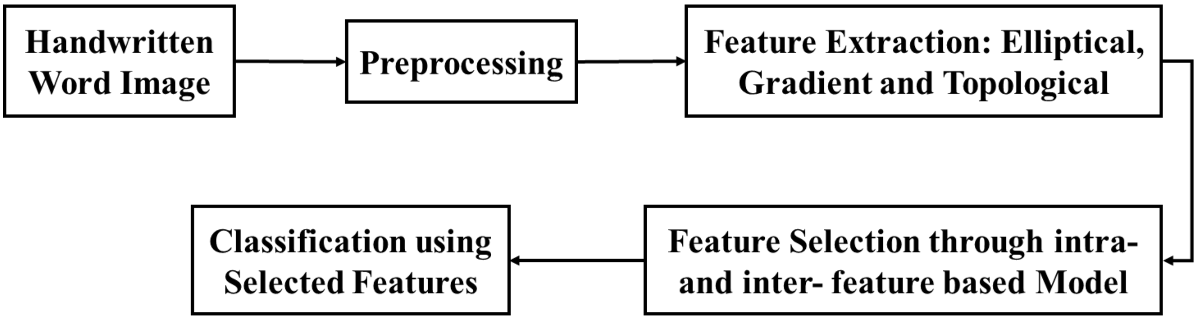

Handwritten Word Image
Preprocessing
Feature Extraction: Elliptical, Gradient and Topological
Feature Selection through intra- and inter- feature based Model
Classification using Selected Features

Fig. 6.1 Block diagram of the present word recognition model

In addition to these, combination two or more good features may not yield better result unless the features to be combined have the ability to provide some complimentary information about patterns to be classified. It is also true that extracting features from patterns using different approaches provide some new information about the patterns, but identifying such informative feature set is always a complex research problem. Here comes the usefulness of the FS algorithms. FS refers to the study of algorithms for generating a feature subset that represents underlying pattern classes in a better way. The main purpose of this is to identify an optimal set of features to represent the patterns and reduce the computational cost, without weakening discriminative capability of the same.

However, the problem of FS can be stated as a problem of selecting the best feature subset from the search space. Selecting a subset from a set is an NP hard problem i.e., there exist $2^{n} - 1$ number of possible subsets for an n-element feature set. Such presence of exponentially possible subsets in the solution space drives the researchers towards using some optimizing techniques. FS techniques, as found in literature, have followed mainly three different approaches *namely*, filter, wrapper and embedded (i.e., wrapper-filter) [196]. Here, a wrapper-filter based FS approach has been considered since it performs better than it counterparts while searching the optimal set of features that represents the patterns under consideration in a better way.

MA is a meta-heuristic optimization algorithm which follows wrapper-filter based FS implementation [89]. It is an improvement on the classical GA through the inclusion of self-improvements of memes. The combination performs FS through a wrapper method while considering the influence of intrinsic properties of the features through the filter method. In particular, the filter method fine-tunes the population of GA by adding or deleting features based on ranker (here, ReliefF [197]) information. This inclusion of local search converges it to a better solution much faster than GA. These very facts along with its spectrum of applications [198] motivate the choice of MA over GA.

In implementation of MA, an n-element feature set is represented as chromosome consisting of n genes which is encoded as n-bit binary string. A '1' in a chromosome at position i indicates that the i^{th} feature is selected (or included) and '0' denotes otherwise. MA begins with the creation of a random population of chromosomes. The chromosomes are ranked using a multi-objective optimization technique as described in section 6.2.1.4. Here, ranking is done in offline and is used multiple times to save time. In this work, popularly used filter method called ReliefF [53] has been used. The chromosomes in the population are retained using the elitism rule i.e. the child chromosomes are allowed to substitute those chromosomes in the population which are inferior (according to the value of the objective function) to them. The child chromosomes are created by the operations of local search, followed by crossover and then mutation which are described in sections 6.2.1.1, 6.2.1.2 and 6.2.1.3 respectively. The flowchart of the steps of MA is given in Fig.6.2.

6.2.1.1 Local Search

Local search is the application of Lamarkian learning [199] which involves the creation of an ordered pair (k_1, k_2), where k_1, k_2 $\in \mathbb{N}$, which is used to improve each chromosome by removing k_2 number of least ranked genes (here, features) in the chromosome and by adding k_1 number of best ranked genes from the excluded set of features. The value of the pair, however, cannot be too large as it would lead to deletion of best ranked features, so the values of k_1 and k_2 are upper bounded at 5% of the total number of features.

6.2.1.2 Crossover

This genetic operation provides the means to explore the search space. Here two-point crossover is used. The advantage of this form of crossover is the non-requirement of a probability as required for uniform crossover. Firstly, two chromosomes are selected from the population by applying the concept of a roulette wheel (discussed later in section 6.3.1.5) [200] using the ranks of the chromosomes. Two random points are selected and the bits lying between the two points are exchanged between the two parents to form two child chromosomes. The use of roulette wheel allows more probability of reproduction to the parents having better rank.

The number of crossovers done in each generation is randomly selected between [2, 5]. The crossover process is described in Algorithm 6.1.

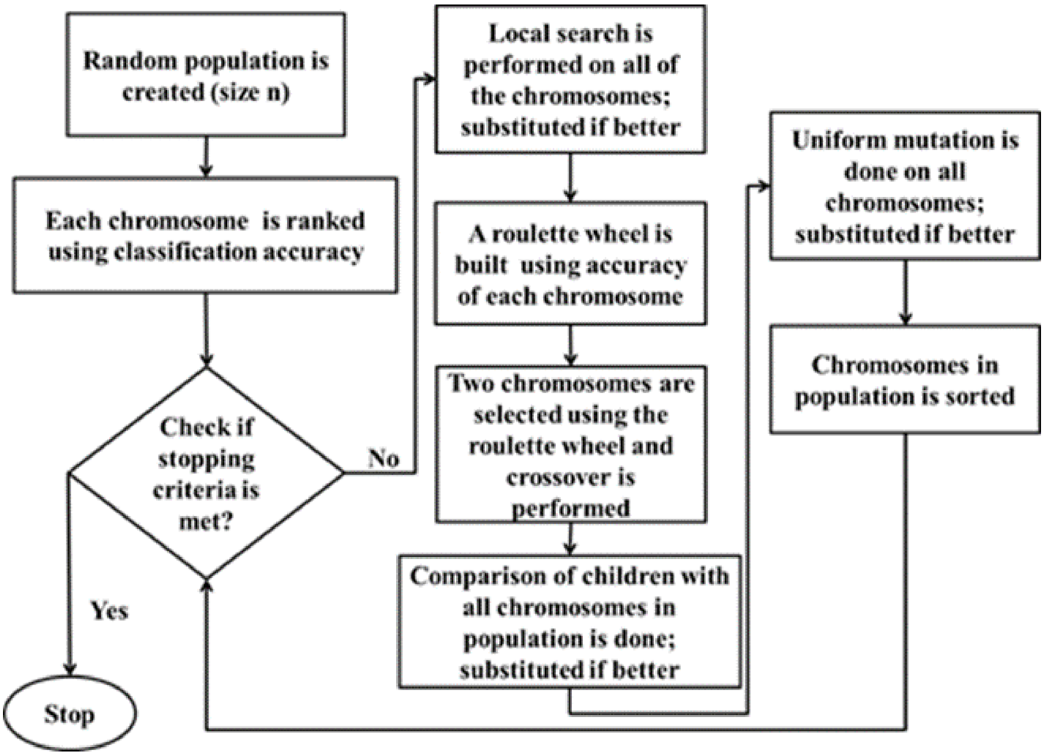

Random population is created (size n)
Each chromosome is ranked using classification accuracy
Check if stopping criteria is met?
No
Yes
Stop
Local search is performed on all of the chromosomes; substituted if better
A roulette wheel is built using accuracy of each chromosome
Two chromosomes are selected using the roulette wheel and crossover is performed
Comparison of children with all chromosomes in population is done; substituted if better
Uniform mutation is done on all chromosomes; substituted if better
Chromosomes in population is sorted

Fig. 6.2 Working procedure of MA

6.2.1.3 Mutation

Mutation is a genetic operation which opposed to crossover provides exploitation of the search space. Uniform mutation is performed here. In this regard, each bit in the chromosome is flipped with probability of p. The algorithm of the uniform mutation technique, used here, is described in Algorithm 6.2.

6.2.1.4 Objective Function

The fitness function of the multi-objective MA, used in the present work, gives more importance to the recognition accuracy than the number of features reduced. This is because though reduction of feature dimension is necessary but that should not be achieved at the cost of the recognition ability of HWR model. The designed multi-objective function is described in Algorithm 6.3. Here, MLP is used as classifier.

Algorithm 6.1 Two-point crossover

Input:

$\boldsymbol{p_1}$ and $\boldsymbol{p_2}$: Randomly selected parent chromosomes from population using the roulette wheel

Output:

$\boldsymbol{child_1}$ and $\boldsymbol{child_2}$: Children chromosome

Steps:

```
1. Let size of the chromosome be n
2. Randomly generate two natural numbers, say, $n_1$ and $n_2$ ∈ [1, n]
3. Perform two-point crossover to get $child_1$ and $child_2$ chromosomes (exchange the
   chromosome portion of $p_1$ and $p_2$ which belongs to [$n_1$, $n_2$])
4. Return children chromosome
```

Algorithm 6.2 Uniform mutation

Input:

$\boldsymbol{P(M)}$**:** Probability of mutation

$\boldsymbol{C}$: A chromosome which is selected from population

Output:

Mutated chromosome

Steps:

```
1. Find length of chromosome, say $L(C)$
2. Generate a random number, say $a$
3. For ($i = 1$ to $L(C)$)
   {
     generate a random number $a \in [0,1]$
     if (a<$P(M)$)
      {
         flip the value at position $i$
      }
   }
4. Return mutated chromosome
```

6.2.1.5 Roulette Wheel

Roulette wheel or fitness proportionate selection is a selection operator that is used to choose the parent chromosomes for performing the crossover and mutation operation. The probability of selecting i^{th} chromosome (p_i) from the population for crossover is considered as $p_i = \frac{RA(i)}{\sum_{i=1}^{z} RA(i)}$, where, $RA(i)$ is recognition accuracy using features in i^{th} chromosome and z is the size of the population. This allows the selection of fitter chromosomes to produce the children a more probable event.

6.2.1.6 Stopping Criteria

The number of generations (g) generated by MA is capped at 10 i.e. the number of iterations are restricted to 10. But the MA stops (i.e., deemed as it has converged) if for 3 continuous iterations the population undergoes no improvement i.e. no better chromosome becomes the member of population under consideration for 3 continuous iterations.

Algorithm 6.3 Multi-objective function

```
Input:
$\boldsymbol{a}$ and $\boldsymbol{b}$: Two chromosomes (here feature subset)
$\boldsymbol{w_1}$: Weight for recognition accuracy
$\boldsymbol{w_2}$: Weight for feature reduction
$\boldsymbol{\alpha}$: Recognition accuracy limit which can be compromised
Output:
    Better chromosome
if ((mod(RA($\boldsymbol{a}$)– RA ($\boldsymbol{b}$))> α)        // RA stands for recognition accuracy
{
  if(RA($\boldsymbol{a}$)> RA ($\boldsymbol{b}$))
    return the chromosome $\boldsymbol{a}$
  else
    return the chromosome $\boldsymbol{b}$
}
else
{
    define
    $\eta(a) = \frac{number\ of\ unused\ features\ in\ \boldsymbol{a}}{Total\ number\ of\ feature\ in\ \boldsymbol{a}}$ and
    $\eta(b) = \frac{number\ of\ unused\ features\ in\ \boldsymbol{b}}{Total\ number\ of\ feature\ in\ \boldsymbol{b}}$
    $val = \left(\left(w_1 \times RA(\boldsymbol{a})\right) + (w_2 \times \eta(\boldsymbol{a}))\right) - (\left(w_1 \times RA(\boldsymbol{b})\right) + (w_2 \times \eta(\boldsymbol{b})))$
    if($val > 0$)
     {
       return chromosome $\boldsymbol{a}$ as better one
     }
    else
     {
       return chromosome $\boldsymbol{b}$ as better one
     }
}
```

6.2.2. Feature Extraction

Feature is a distinctive attribute of an object. Extraction and selection of proper feature set become an integral part of any pattern recognition problem. It has already been mentioned that in this work, three different features descriptors *viz.*, elliptical [73, 84], gradient based [85, 189] and topological [71, 86] are selected as shape, texture and spatial and/or geometrical based features respectively. These three feature extraction techniques are described in the following subsections.

6.2.2.1 Elliptical Feature

This is a shape based feature extraction technique [73, 84]. This feature descriptor generates global as well as local features that provide shape information of a word image. To extract the features, a gray-scale word image, $G = \{g(x,y): (x,y) \in [1,H] \times [1 \times W]\}$, where H and W represent height and width of G respectively, is first converted to noise-free binarized image using the technique described in Chapter 2. Let, $B = \{f(x,y): (x,y) \in [1,H] \times [1 \times W]\}$ is the noiseless binarized word image where, $f(x,y) = '1'$ and $f(x,y) = '0'$ represent foreground and background pixels respectively.

After the preliminary processing, three hypothetical concentric ellipses (say, $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$ from outermost to innermost) are conceptualized over a word image and thereby this process generates four non-overlapping concentric regions $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3$ and $\mathcal{R}_4$ (see Fig. 6.3(a)) [84]. The center of each of these ellipses is assigned by center of gravity (CG) of the word image under consideration. Then, word image is further partitioned into four regions ($\mathcal{R}'_1, \mathcal{R}'_2, \mathcal{R}'_3$ and $\mathcal{R}'_4$) by drawing lines parallel to the minor axis through the center and two foci points of the outermost ellipse (see Fig. 6.3(b)). Next, different features ($\mathcal{F}_i$,$i = 1,2, \ldots ,13$ in eqs. (6.1)-(6.13)), depending on foreground (i.e., $f(x,y) = '1'$) and/or background pixel (i.e., $(x,y) = '0'$) counts, are computed from the word images and considered as global elliptical features. To extract the local characteristics of the strokes in a word image, it is divided into four sub-images along major and minor axes of the outermost ellipse (i.e., $\mathcal{E}_1$ (see Fig. 6.3 (c))). Then, all the above mentioned features are extracted from each of these sub-images. Therefore, the dimension of elliptical feature becomes 65 (13 (extracted from entire image) + 52 (extracted from 4 sub-images)).

$$\mathcal{F}_1 = |\{(x,y): (x,y) \in \mathcal{R}_1 \wedge (f(x,y) =' 1')\}| \tag{6.1}$$

$$\mathcal{F}_2 = |\{(x,y): (x,y) \in \mathcal{R}_2 \wedge (f(x,y) =' 1')\}| \tag{6.2}$$

$$\mathcal{F}_3 = |\{(x,y): (x,y) \in \mathcal{R}_3 \wedge (f(x,y) =' 1')\}| \tag{6.3}$$

$$\mathcal{F}_4 = |\{(x,y): (x,y) \in \mathcal{R}_4 \wedge (f(x,y) =' 1')\}| \tag{6.4}$$

$$\mathcal{F}_5 = \frac{|\{(x,y): (x,y) \in (\mathcal{R}_1 \cup \mathcal{R}_2 \cup \mathcal{R}_3) \wedge (f(x,y) =' 1')\}|}{|\{(x,y): (x,y) \in \mathcal{R}_4 \wedge (f(x,y) =' 1')\}|} \tag{6.5}$$

$$\mathcal{F}_6 = \frac{|\{(x,y):(x,y) \in (\mathcal{R}_1 \cup \mathcal{R}_2 \cup \mathcal{R}_3) \wedge (f(x,y) =' 1')\}|}{|\{(x,y):(x,y) \in (\mathcal{R}_1 \cup \mathcal{R}_2 \cup \mathcal{R}_3) \wedge (f(x,y) =' 0')\}|} \tag{6.6}$$

$$\mathcal{F}_7 = |\{(x,y):(x,y) \in \mathcal{R}'_1 \wedge (f(x,y) =' 1')\}| \tag{6.7}$$

$$\mathcal{F}_8 = |\{(x,y):(x,y) \in \mathcal{R}'_2 \wedge (f(x,y) =' 1')\}| \tag{6.8}$$

$$\mathcal{F}_9 = |\{(x,y):(x,y) \in \mathcal{R}'_3 \wedge (f(x,y) =' 1')\}| \tag{6.9}$$

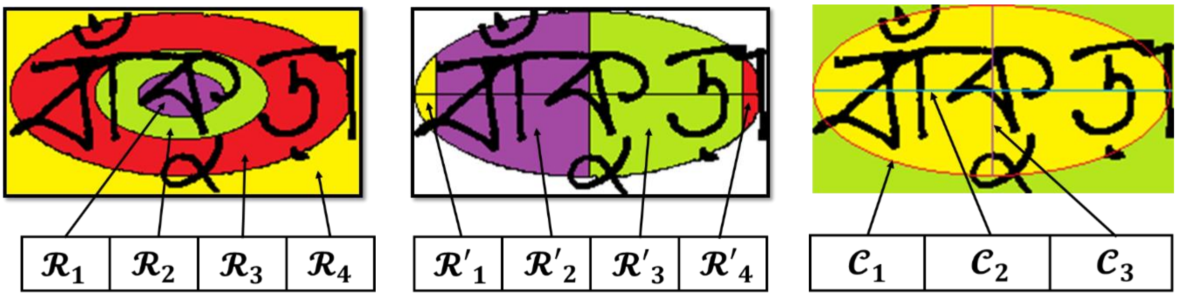
$$\mathcal{F}_{10} = |\{(x,y):(x,y) \in \mathcal{R}'_4 \wedge (f(x,y) =' 1')\}| \tag{6.10}$$

$$\mathcal{F}_{11} = |\{(x,y):f(x,y) \in \mathcal{C}_1 \wedge (f(x,y) =' 1')\}| \tag{6.11}$$

$$\mathcal{F}_{12} = |\{(x,y):f(x,y) \in \mathcal{C}_2 \wedge (f(x,y) =' 1')\}| \tag{6.12}$$

$$\mathcal{F}_{13} = |\{(x,y):f(x,y) \in \mathcal{C}_3 \wedge (f(x,y) =' 1'')\}| \tag{6.13}$$

In eqs. (6.11)- (6.13), $\mathcal{C}_i$ $(i = 1, 2, 3)$ represent the contour, major and minor axes of ellipse $\mathcal{E}_1$ (see Fig. 6.3(c)).



(a) (b) (c)

Fig. 6.3 Illustration of eight different regions, four in (a) and another four in (b). Figure in (c) indicates the positions of contour, major and minor axes of the outermost ellipse

6.2.2.2 Gradient based Feature

Histogram of Oriented Gradients (HOG) is a feature descriptor which is proven to be useful in object detection [87]. HOG feature extraction method needs image resizing of all the pattern classes into fixed size to obtain same length feature vector. This resizing not only strives to compromise with resolution, but also, destroys the aspect ratio of word images which is an important feature of a word image while considering handwritten word recognition in holistic way.

Considering the benefits of applying HOG feature descriptor, few variations, described in [85, 88, 189], have been made during this thesis work. The newly designed features are termed as gradient based feature which are extracted from hypothetically generated $m \times m$ sub-images (e.g., 6×6 in [189]). It is worthy in mentioning that this feature descriptor excludes the block/cell normalization as described in [87], which is the key factor for deciding length of a feature vector based on image dimension.

Let, $\mathrm{s}(\mathrm{x}, \mathrm{y})$ denotes image after applying Gaussian smoothing on a gray-scale image $g(x, y)$. An instance of gray-scale image and the corresponding Gaussian smoothen image are shown in Fig. 6.4(a-b). The gradient values in horizontal (g_x) and vertical (g_y) directions are calculated from $s(x, y)$ using eq (6.14) and eq (6.15) respectively.

$$g_x = s(x + 1, y) - s(x, y) \tag{6.14}$$

$$g_y = \mathrm{s}(\mathrm{x}, \mathrm{y} + 1) - \mathrm{s}(\mathrm{x}, \mathrm{y}) \tag{6.15}$$

Let, $\mathcal{M}(x, y)$ and $\mathcal{D}(x, y)$ represent the magnitude and direction at point (x, y) respectively which are calculated by eqs. (6.16) and (6.17) and these images are shown in Fig. 6.4(c-d).

$$\mathcal{M}(x, y) = \sqrt{\mathrm{g}_\mathrm{x}^2 + \mathrm{g}_\mathrm{y}^2} \tag{6.16}$$

$$\mathcal{D}(x, y) = \tan^{-1}(\frac{g_y}{g_x}) \tag{6.17}$$

Thereafter, $\mathcal{D}$ is partitioned into 8 bins ($[0^{\circ}, 45^{\circ})$, $[45^{\circ}, 90^{\circ})$, …, $(315^{\circ}, 360^{\circ}]$). Next, to acquire histogram information, gradient magnitude for a pixel (x, y) (i.e., $\mathcal{M}(x, y)$) is added to the content of the bin corresponding to the $\mathcal{D}(x, y)$ value. Applying this in a repetitive manner, histogram information in each of the 8 bins is obtained which means it generates 8 such features from an image. Such histogram information is extracted from each of the hypothetically generated $m \times m$ sub-images (also known as grid) to obtain local information of a word image. Therefore, it generates a feature vector of length $8 * (m^2 + 1)$ $(i.e. 8 * m * m + 8)$ from a word image by local and global means to represent it in feature space. However, selection of optimal grid size for handwritten word images is a challenging issue. For example, 3 samples of same word image, segmented in different grid sizes, are depicted in Fig. 6.5. These images also provide the evidence for requirement of local information extraction from word images. The images show that a word belonging to the other class bears dissimilar data distribution in the corresponding grid. Hence, in the present work, value of m is set experimentally.

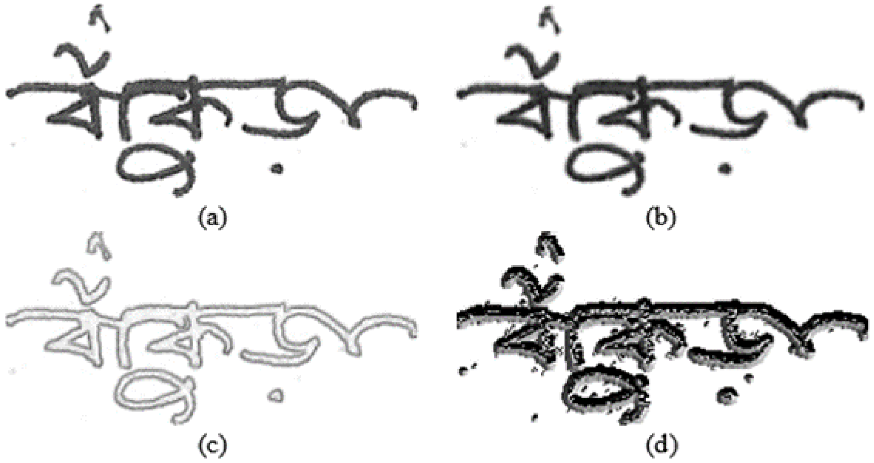
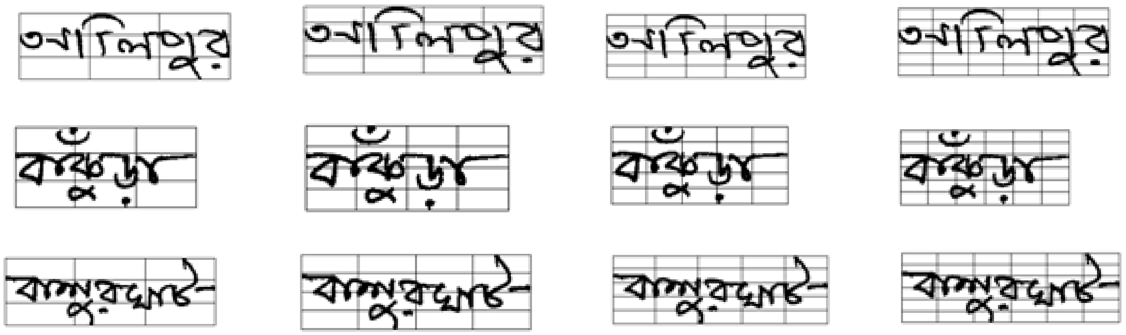


Fig. 6.4 Representation of different forms of pre-processed images generated during gradient based feature extraction: (a) gray-scale input word image (i.e., $g(x, y)$), (b) image after applying Gaussian smoothing (i.e., $s(x, y)$), (c) magnitude image (i.e., $\mathcal{M}(x, y)$) and (d) and image representing direction values (i.e., $\mathcal{D}(x, y)$)



(a) 3 × 3 grid (b) 4 × 4 grid (c) 5 × 5 grid (d) 6 × 6 grid

Fig. 6.5 Illustration of different grid sizes; each row shows different instances of the same city name which are segmented into varied number of grids

6.2.2.3 Topological Feature

Topological feature was first introduced in the work [86] which has been further modified in [71]. The features have been extracted either from entire word image (for extracting global information) or from sub-images (for extracting local information). For calculation of features from sub-images, first word images are divided into a p × q grid (for example see Fig. 6.5(a) where $p = q = 3$) and then minimal bounding box enclosing all the object pixels inside a grid is estimated. The values of p and q are set here experimentally. The topological features extracted here are described below.

Area: Areas covered by the different patterns (here, query and target words) vary a lot because of the presence of different number of characters and also for shape variations of these characters therein. The area of B is calculated by H $*$ W. Area based feature values are extracted from sub-images only. Therefore, total number of area based feature ($\mathcal{L}_A$) extracted from an image is $p * q$. It is to be noted that the area in each of grids is calculated

Pixel Density: It represents number of object pixels per unit area for each sub-image which is calculated as

$$PD = \frac{|\{(i,j) : f(i,j) = 1 \wedge (i,j) \in [1,H] \times [1,W]\}|}{Area\ of\ the\ sub - image} \tag{6.18}$$

Pixel density is extracted from each of the sub-images and therefore the length this feature vector, hereafter acquainted as $\mathcal{L}_{PD}$, is $p * q$.

Aspect Ratio (AR): AR of an image is defined as ratio of width to height which generally differs when word images contain different number of characters. So, this feature is also considered here and the feature value is calculated as

$$AR = \frac{W}{\mathrm{H}} \tag{6.19}$$

AR is extracted from entire image and each of the sub-images. Therefore, total number of AR features ($\mathcal{L}_{AR}$) extracted from a word image is represented as

$$\mathcal{L}_{AR} = p * q + 1 \tag{6.20}$$

Longest *Run:* Longest run feature is considered here as another topological feature. This feature consists of two feature values *viz.*, lengths of the longest run along X and Y directions respectively. Let, B has $N_i(\geq 1)$ number of runs (occurrence of continuous object pixels) along i^{th} row and HL_{ij} represents the length of j^{th} run along i^{th} row. The horizontal longest run (HLR) is calculated as

$$HLR = \max_i\{\max_j\{HL_{ij}\}\}, 0 \leq j \leq N_i, 1 \leq i \leq H \tag{6.21}$$

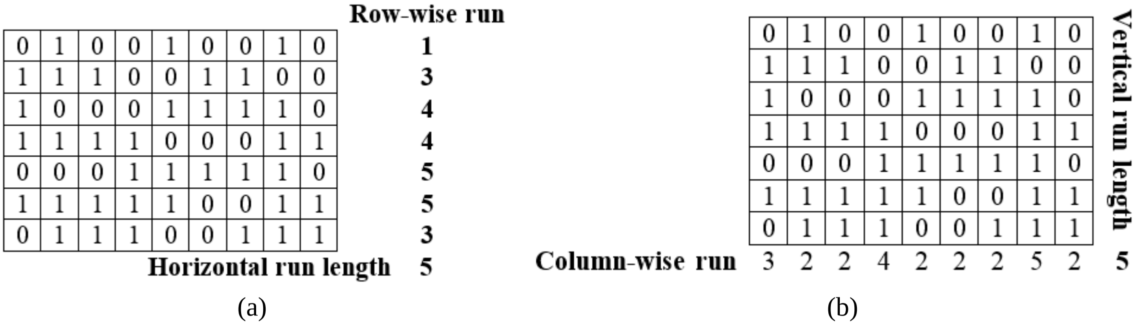
Vertical longest-run (VRL) feature is computed by identifying runs in column-wise manner. Let, B has $N_i(\geq 1)$ number of runs along i^{th} column and VL_{ij} represents the length of j^{th} run along i^{th} column. Now, VLR of B is measured as

$$VRL = \max_i\{\max_j\{VL_{ij}\}\}, 0 \leq j \leq N_i, 1 \leq i \leq W \tag{6.22}$$

These feature values are extracted from entire word image and each of the sub-images. Therefore, a feature vector of length $2 * (p * q + 1)$ is generated which means total number of longest run features ($\mathcal{L}_{LR}$) is defined by

$$\mathcal{L}_{LR} = 2 * (p * q + 1) \tag{6.23}$$

Both the features are described in Fig. 6.6(a-b) using example of a typical image segment.



									Row-wise run
0	1	0	0	1	0	0	1	0	1
1	1	1	0	0	1	1	0	0	3
1	0	0	0	1	1	1	1	0	4
1	1	1	1	0	0	0	1	1	4
0	0	0	1	1	1	1	1	0	5
1	1	1	1	1	0	0	1	1	5
0	1	1	1	0	0	1	1	1	3
Horizontal run length									5

(a)

	0	1	0	0	1	0	0	1	0
	1	1	1	0	0	1	1	0	0
	1	0	0	0	1	1	1	1	0
	1	1	1	1	0	0	0	1	1
	0	0	0	1	1	1	1	1	0
	1	1	1	1	1	0	0	1	1
	0	1	1	1	0	0	1	1	1
Column-wise run	3	2	2	4	2	2	2	5	2

Vertical run length 5

(b)

Fig. 6.6 Illustration of run-length features: (a) horizontal and (b) vertical

Centroid: Centroid feature is based on the CG of an image. This position varies depending on the shape of pattern. Two sample images are shown in Fig. 6.7(a-b) where position of centroid is marked. Centroid coordinates (C_X, C_Y) are calculated as

$$C_X = \frac{\sum i}{|\{(\mathrm{i,j}): \mathrm{f(i,j)} = 1 \wedge (i,j) \in [1,H] \times [1,W]\}|} \tag{6.24}$$

$$C_Y = \frac{\sum j}{|\{(\mathrm{i,j}): \mathrm{f(i,j)} = 1 \wedge (i,j) \in [1,H] \times [1,W]\}|} \tag{6.25}$$

These two features are extracted (in global way) from an entire image and from upper and lower triangular parts of an images, which are separated by principal and non-principal diagonals (see Fig. 6.8(a-b)). Therefore number of feature generated from an image is $2 * (1 + 2 + 2) = 10$. Also, all these features are again extracted from all the sub-images for obtaining local information. Therefore, total number of centroid features ($\mathcal{L}_C$) can be formulated as

$$\mathcal{L}_C = 10 * (p * q + 1) \tag{6.26}$$

Projection Length: Projection lengths of word image on the principal axes i.e., on X and Y axes are considered here as feature values. Let $\mathcal{R}$ and $\mathcal{C}$ are sets of row index and column index respectively that contain at least one object pixel and are defined as

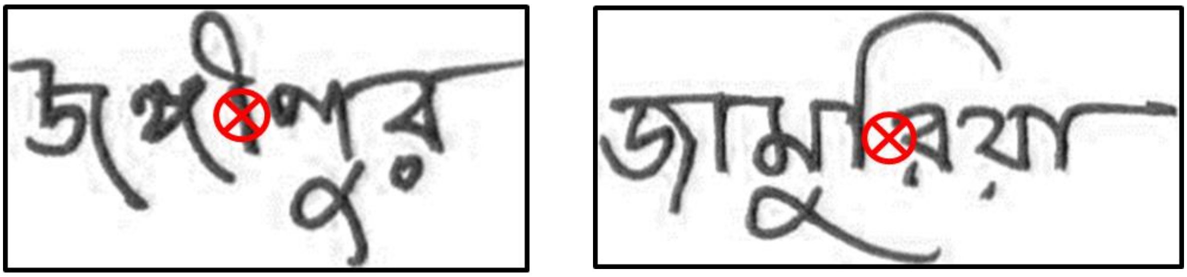
$$\mathcal{R} = \{i: f(i,j) = 1 \wedge (i,j) \in [1,H] \times [1,W]\ for\ some\ j \epsilon\ [C1, C2]\} \tag{6.27}$$

$$\mathcal{C} = \{j: f(i,j) = 1 \wedge (i,j) \in [1,H] \times [1,W]\ for\ some\ i \epsilon\ [R1, R4]\} \tag{6.28}$$

Now the horizontal projection length (HPL) and vertical projection length (VPL), illustrated in Fig. 6.9, are defined as

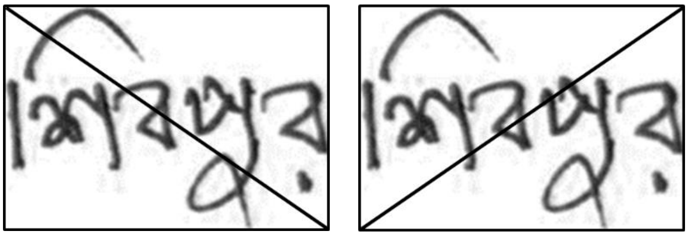
$$HPL = |\mathcal{R}| \tag{6.29}$$

$$VPL = |\mathcal{C}| \tag{6.30}$$



(a) (b)

Fig. 6.7 Depiction of position of centroid point for two different word images. The position of centroid is marked using ⊗



(a) (b)

Fig. 6.8 Showing hypothetically generated sub-images by segmenting an image along (a) principal diagonal and (b) non-principal diagonal

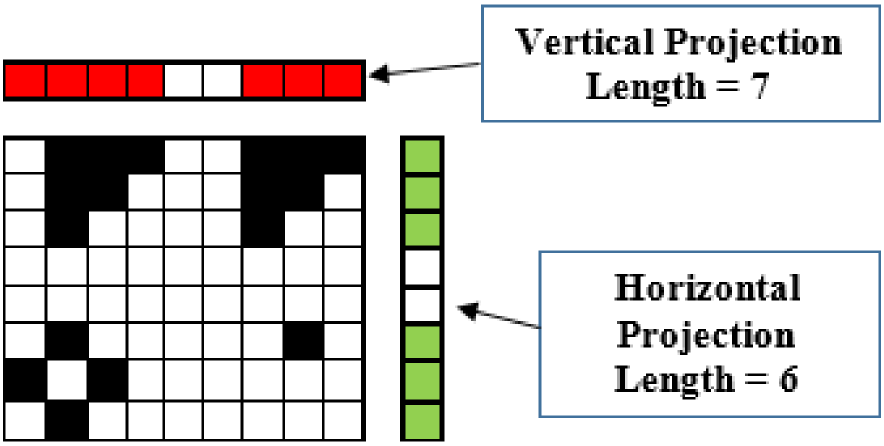

Vertical Projection Length = 7
Horizontal Projection Length = 6

Fig. 6.9 Illustration of vertical and horizontal projection lengths. Black and white cells indicate foreground and background pixels respectively

These two features are extracted from entire image and from upper and lower triangular parts of the image which are separated by principal and non-principal diagonals (see Fig. 6.8(a-b)). Therefore, using this projection length based concept, the length of feature vector ($\mathcal{L}_{PL}$) which is extracted from a word images is $2 * (1 + 2 + 2) = 10$

By accumulating all the above mentioned features, length of the topological feature becomes

$$\mathcal{L}_S = \mathcal{L}_A + \mathcal{L}_{PD} + \mathcal{L}_{AR} + \mathcal{L}_{LR} + \mathcal{L}_C + \mathcal{L}_{PL}$$

$$or, \mathcal{L}_S = p * \mathrm{q} + p * \mathrm{q} + (\mathrm{p} * q + 1) + 2(\mathrm{p} * q + 1) + 10 * (\mathrm{p} * q + 1) + 10$$

$$\therefore \mathcal{L}_S = 15 * p * q + 23 \tag{6.31}$$

6.2.3 Hierarchical Feature Selection Model

It has already been mentioned that in this work, three different feature descriptors *viz.*, elliptical [73, 84], gradient based [85, 189] and topological [71, 86] have been extracted for recognizing handwritten *Bangla* word images in a holistic way. Some of these features may provide non-complimentary information individually or while combined. Therefore, to select an optimal feature set that represents the data in better way, a HFS model is introduced during this thesis work. For optimization purpose MA has been used here. Basically, MA has been applied at different levels of present HFS model to find out an optimal feature set. The selection strategies are described below.

(i) Features, in general, are either extracted from entire pattern or from different closed local regions to obtain global or regional characteristic of the patterns. The global features normally carry perceptual information of the entire patterns while local features of a pattern, in most of the cases, are heavily reliant on the structure of local regions. Thus, the combined global and local features, in general, may generate noisy and irrelevant features. Therefore, such feature vector should to be passed through FS technique to get optimal combination of features. Here, all the feature sets *viz.*, elliptical, gradient based and topological are extracted in global and local means and each type of features may contain noisy and irrelevant features. Therefore, an intra-feature set based FS procedure is employed for obtaining optimal features from each of the feature sets (see Fig. 6.10).

(ii) Combining two or more good features (individual or group of features) may not result in an optimal feature set unless they provide complementary information about the patterns. Finding this optimal subset from a large dimension of features is not possible without applying some intelligent FS methods. Hence, MA based FS is further applied on the features that have been obtained after combining the optimal features generated through intra-feature set based FS (see Fig. 6.10).

The feature extraction strategies along with the present FS technique is illustrated in Fig. 6.10. This figure illustrate that, first, each of the three different feature sets is optimized through MA based FS technique and then these features are concatenated to get optimized again. This entire process of FS is termed here as HFS model.

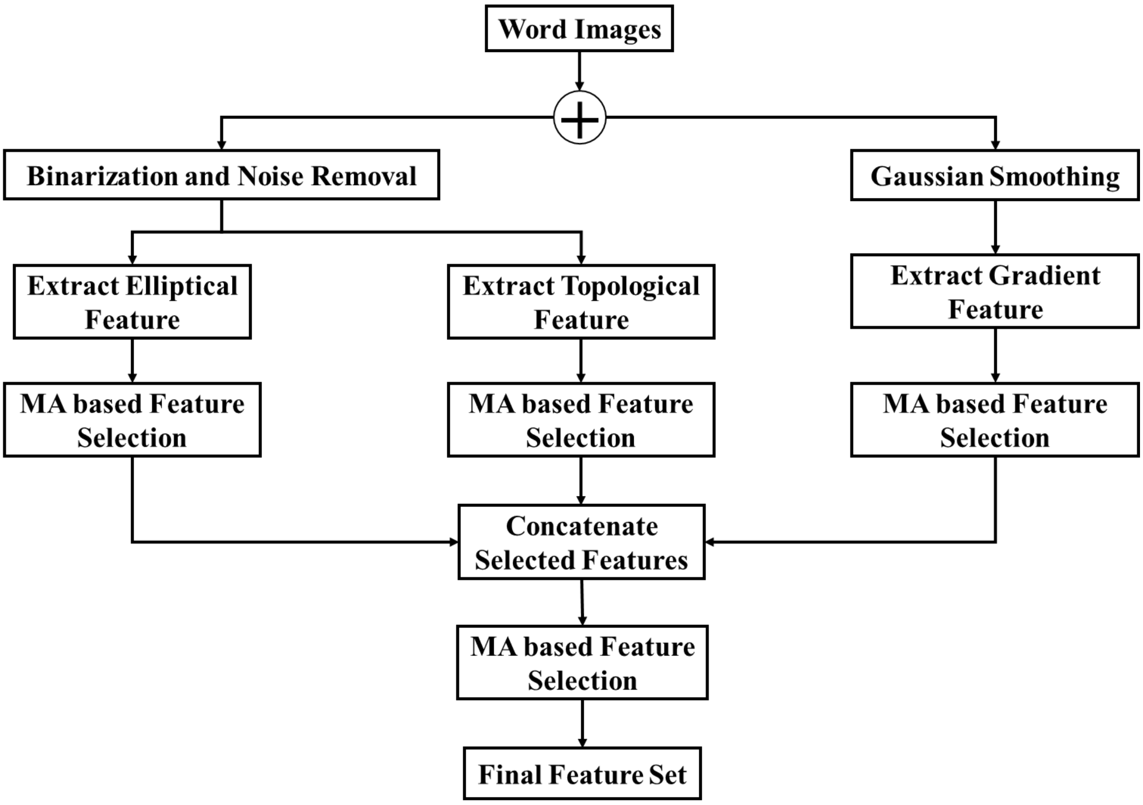

Word Images
Binarization and Noise Removal
Gaussian Smoothing
Extract Elliptical Feature
Extract Topological Feature
Extract Gradient Feature
MA based Feature Selection
MA based Feature Selection
MA based Feature Selection
Concatenate Selected Features
MA based Feature Selection
Final Feature Set

Fig. 6.10 Block diagram of present HFS model

6.3 Data Preparation

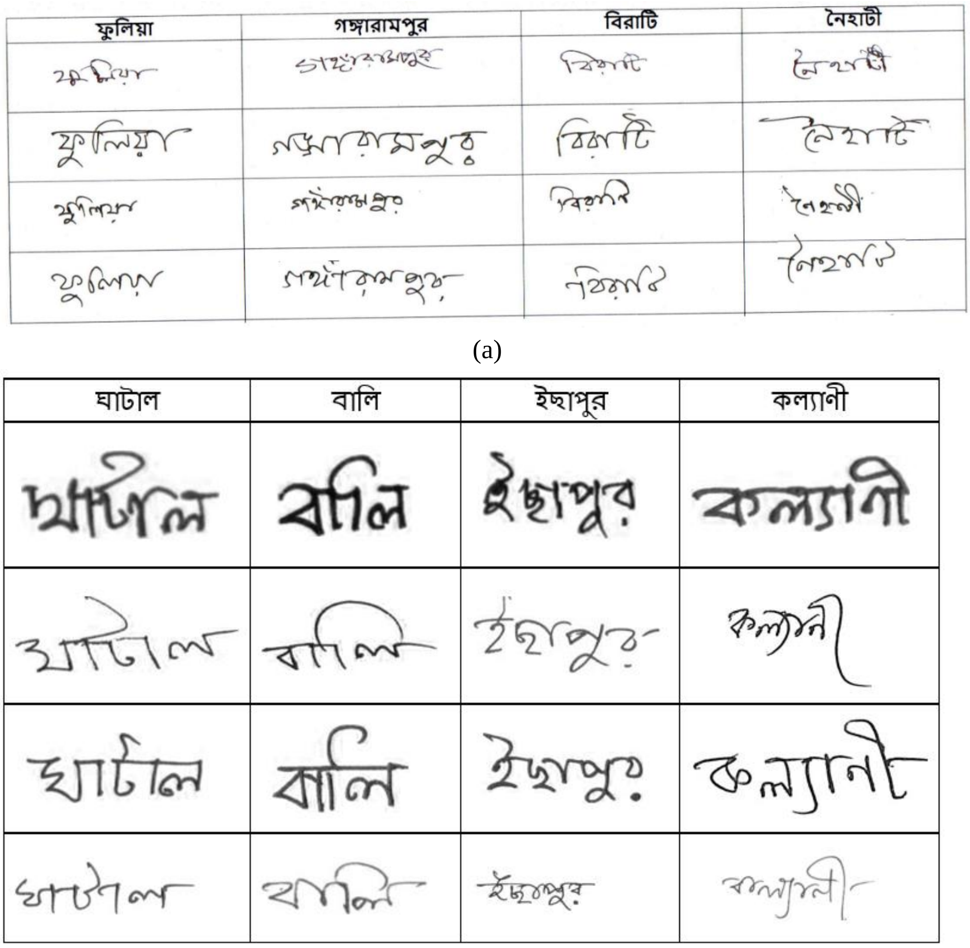
One of the main reasons for the slow progress of research on handwritten word recognition for regional languages is the unavailability of suitable databases. As holistic handwritten word recognition systems are generally developed for specific applications, these mainly deal with limited lexicons. To carry out training and evaluation of such systems, some handwritten word databases like IFN/ENIT (in Arabic script) [201] and CENPERMI [202] and IAM [203] (in Roman script) are made available to the research community either on-demand basis or through subscription charges. However, no such standard database of *Bangla* handwritten words is found in the literature, which can be used to evaluate any newly designed holistic handwritten

Bangla word recognition system. To address this need, in the present work, a database of 18,000 handwritten *Bangla* word images, written by around 300 different native writers belonging to different age groups, sex and educational backgrounds, is prepared. Word images in the current database contain the names of 120 different cities in West Bengal, a state of India, with 150 samples for each city name. The city names listed in Table 6.1 are chosen based on their population and the literacy rate. The present database includes almost all urban regions of West Bengal.

Handwritten words, in this work, have been collected in A4 size datasheets containing a grid of 10 rows and 3 to 5 columns depending upon the word length. The writers were asked to write each city name inside the rectangular boxes only. Such datasheets are then scanned using a flat-bed scanner with a resolution of 300 dpi and are stored as 24 bit BMP image file. Two sample portions of filled-in datasheets are shown in Fig. 6.11 (a-b). From each such image of the datasheets, handwritten word images are cropped automatically. To get an idea about the diversity present in the database, some samples of skewed/broken, misspelled/differently spelled words, structurally similar word groups and skewed samples are shown in Table 6.2, Table 6.3, Table 6.4 and Table 6.5 respectively. In addition, the collected word samples possess notable intra-class size variations in terms of word height and width (see Fig. 6.11(a)) as well as variations in stroke width variations (see Fig. 6.11(b)). The said variations in each of the word classes are illustrated in Fig. 6.12(a-f). In addition, some cursive word samples taken from the prepared database are shown in Table 6.6 which demonstrates the said variations. The link, given in [195], provide more detail description regarding the height, width, number of ascendants and descendants of the word images present in each word class, and these are stored in a file named as “ReadMeCMATERdb2.1.2.pdf” along with the entire database. One of the main contributions of the current work is that the database is already made freely available to the entire research community.

Table 6.1 List of city names, written in Bangla, in the database

Class #	Name	Class#	Name	Class#	Name	Class #	Name
1	আলিপুর	31	বনগাঁ	61	এগরা	91	কোন্নগর
2	বালুরঘাট	32	বানপুর	62	ফুলিয়া	92	কুলটি
3	বাঁকুড়া	33	বাঁশবেড়িয়া	63	গঙ্গারামপুর	93	লালগোলা
4	বারাসাত	34	বাঁশড়া	64	গারুলিয়া	94	মধ্যমগ্রাম
5	বর্ধমান	35	ব্যারাকপুর	65	গায়েশপুর	95	মহেশতলা
6	বহরমপুর	36	বরানগর	66	ঘাটাল	96	মেমারি
7	চুঁচুড়া	37	বারুইপুর	67	গোবরডাঙ্গা	97	মুর্শিদাবাদ
8	কোচবিহার	38	বসিরহাট	68	গুসকরা	98	নবদ্বীপ
9	দার্জিলিং	39	বেলডাঙা	69	হাবড়া	99	নৈহাটী
10	ইংলিশবাজার	40	বেলঘরিয়া	70	হলদিয়া	100	নলহাটি
11	হাওড়া	41	ভাটপাড়া	71	হালিশহর	101	ঔরঙ্গাবাদ
12	জলপাইগুড়ি	42	বিরাটি	72	হিজুলি	102	পানিহাটি
13	কলকাতা	43	বীরনাগার	73	ইছাপুর	103	পলাশী
14	কৃষ্ণনগর	44	বিষ্ণুপুর	74	ইসলামপুর	104	রামপুরহাট
15	মালদা	45	বোলপুর	75	জামুরিয়া	105	রানাঘাট
16	মেদিনীপুর	46	বজবজ	76	জঙ্গীপুর	106	রিষড়া
17	পুরুলিয়া	47	চাকদহ	77	ঝাড়গ্রাম	107	সাঁইথিয়া
18	রায়গঞ্জ	48	চাঁপদানি	78	কাজোড়া	108	শান্তিপুর
19	সিউড়ি	49	চন্দননগর	79	কালনা	109	শিবপুর
20	তমলুক	50	চিত্তরঞ্জন	80	কল্যাণী	110	শিলিগুড়ি
21	আগারপাড়া	51	দাঁইহাট	81	কামারহাটী	111	শ্যামনগর
22	আজিমগঞ্জ	52	ডালখোলা	82	কাঁচরাপাড়া	112	সোদপুর
23	আরামবাগ	53	ডানকুনি	83	কান্দি	113	সোনামুখী
24	আসানসোল	54	ধুলিয়ান	84	কাঁকশা	114	সোনারপুর
25	অশোকনগর	55	ধূপগুড়ি	85	কাঁথি	115	শ্রীরামপুর
26	বাদকুল্লা	56	দিনহাটা	86	করিমপুর	116	তারকেশ্বর
27	বৈদ্যবাটী	57	ডোমকল	87	কাটোয়া	117	টিটাগড়
28	বলরামপুর	58	দুবরাজপুর	88	খড়্গপুর	118	উখরা
29	বালি	59	দমদম	89	খড়দহ	119	উলুবেড়িয়া
30	ব্যান্ডেল	60	দুর্গাপুর	90	কোলাঘাট	120	উত্তরপাড়া


ফুলিয়া
গঙ্গারামপুর
বিরাটি
নৈহাটী

(a)

ঘাটাল
বালি
ইছাপুর
কল্যাণী

(b)

Fig. 7.11 Sample portion of filled-in datasheets where word samples (in each column) vary significantly in terms of (a) size and (b) stroke width

Table 6.2 Skewed and broken sample word images present in the current database

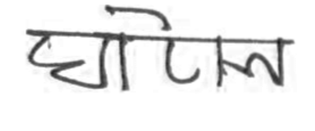
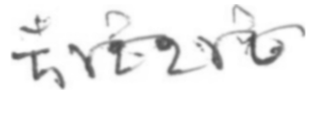
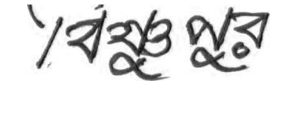
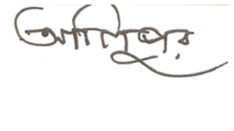
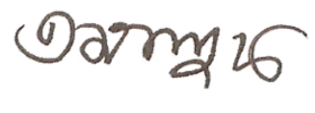
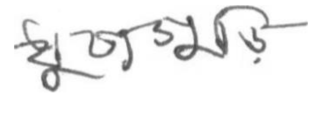
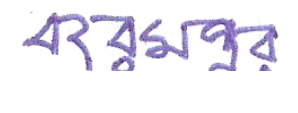
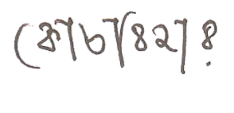
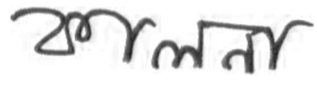
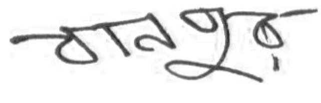
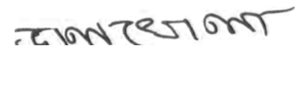
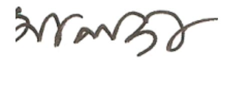
Skewed word image		**Broken word image**	
			
			
			

Table 6.3 Sample word images those are either misspelled or having spelling confusion

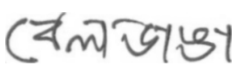
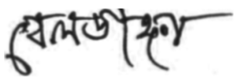
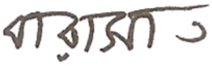
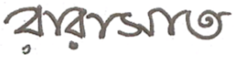
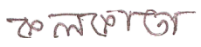
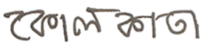
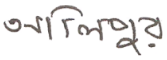
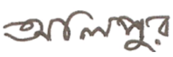
Word images having alternate spelling		Instances of word images which are	
Mostly samples are written as	**Around 10% (in this database) samples written as**	**Rightly spelled**	**Wrongly spelled**
			
			

Table 6.4 Some sample classes of images (in each row) that possess high structural similarity

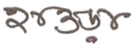
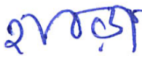
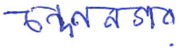
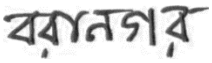
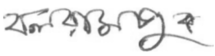
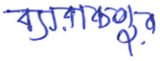
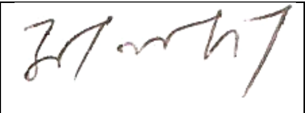
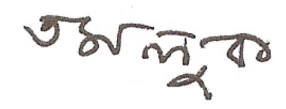
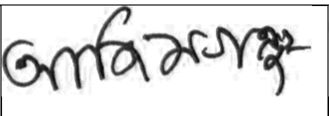
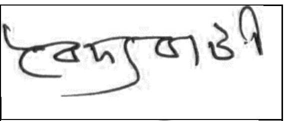
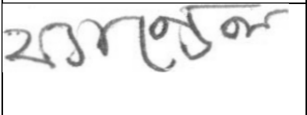
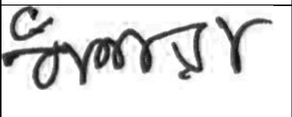
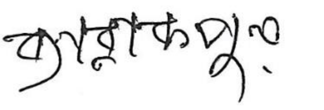
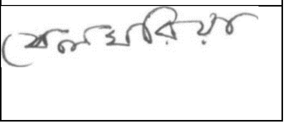
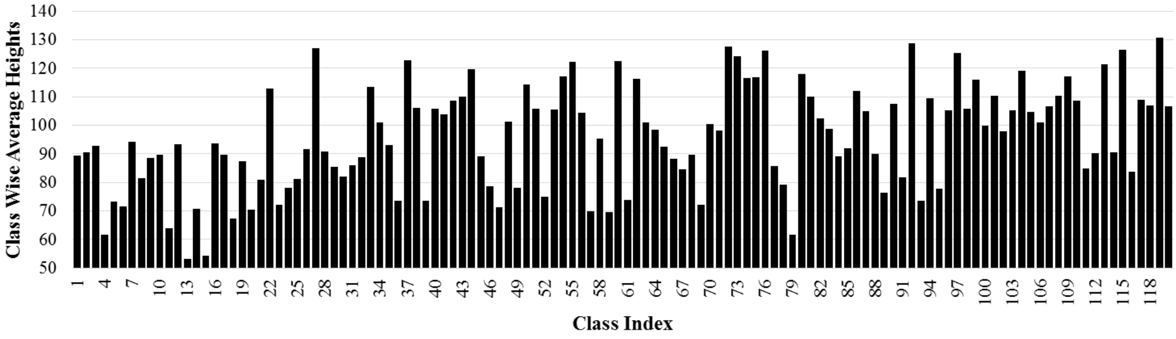
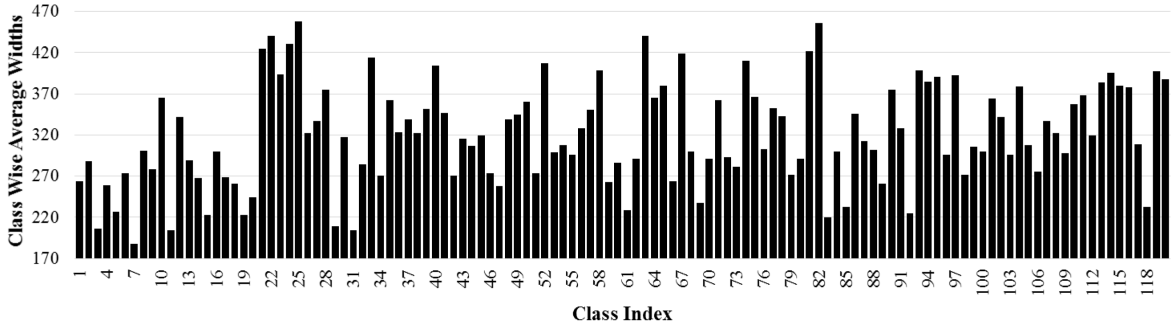
Case	Instance 1	Instance 2
1		
2		
3		

Table 6.5 Some instances of skewed word images present in the current database

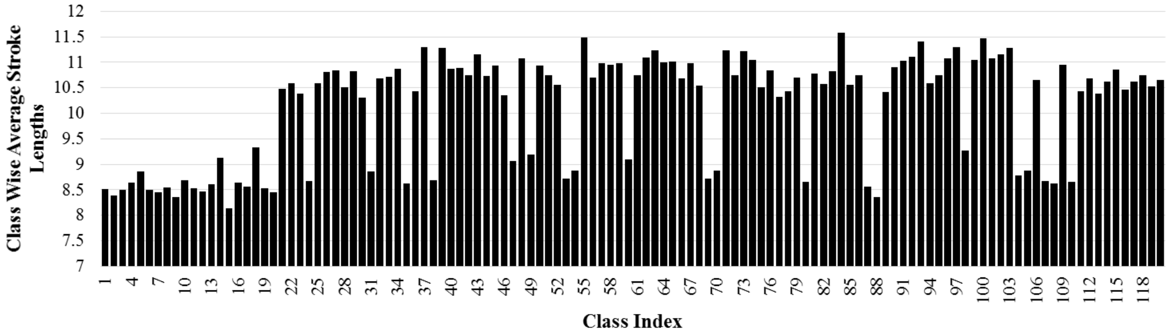
			
			



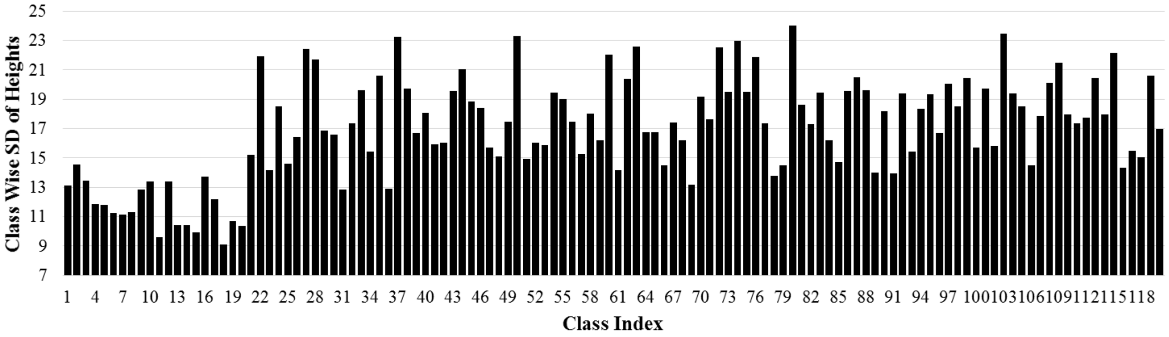
(a)


Class Wise Average Widths
470
420
370
320
270
220
170
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70 73 76 79 82 85 88 91 94 97 100 103 106 109 112 115 118
Class Index

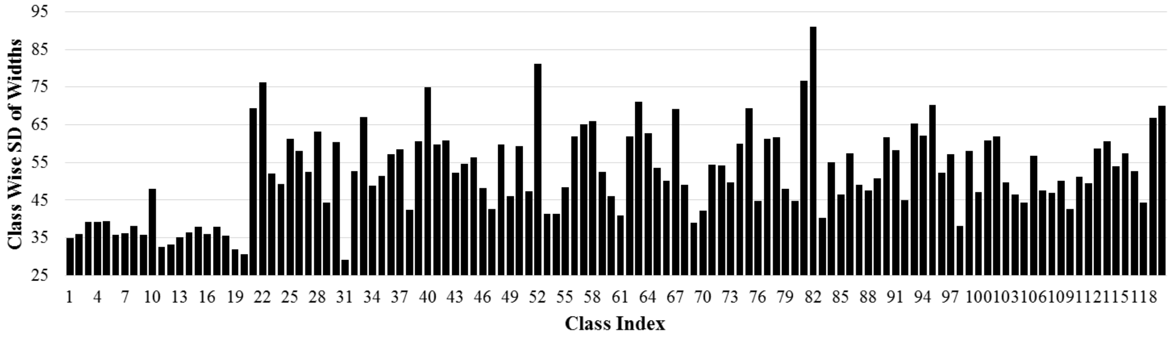
(b)


Class Wise Average Stroke Lengths
12
11.5
11
10.5
10
9.5
9
8.5
8
7.5
7
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70 73 76 79 82 85 88 91 94 97 100 103 106 109 112 115 118
Class Index

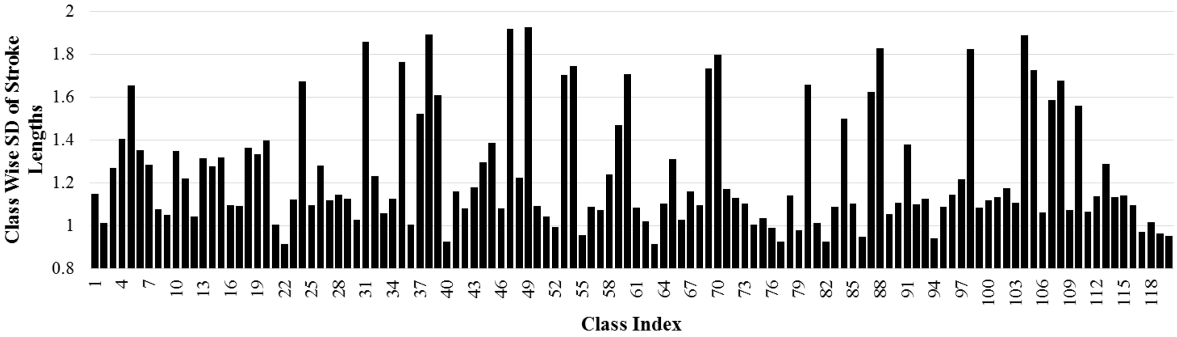
(c)


Class Wise SD of Heights
25
23
21
19
17
15
13
11
9
7
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70 73 76 79 82 85 88 91 94 97 100 103 106 109 112 115 118
Class Index

(d)


Class Wise SD of Widths
95
85
75
65
55
45
35
25
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70 73 76 79 82 85 88 91 94 97 100 103 106 109 112 115 118
Class Index

(e)



(f)

Fig. 6.12 Illustration of class-wise average and standard deviation (SD) of word samples' heights, widths and stroke lengths. These measures are in terms of pixel count

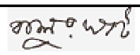
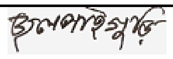
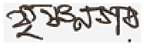
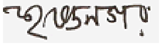
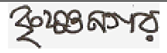
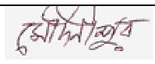
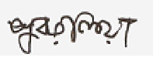
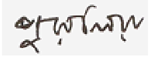
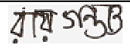
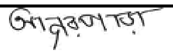
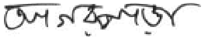
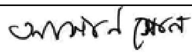
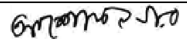
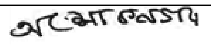
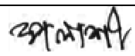
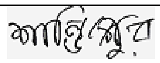
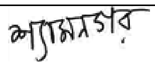
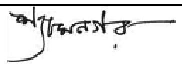
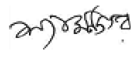
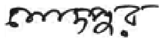
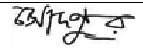
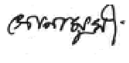
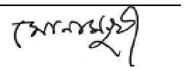
6.4 Experimental Results

The current method is evaluated on the created dataset. Here the classification is carried out using MLP classifier. Also, FS has been carried out using MA based HFS model. All the steps of the current experiment are described in the following subsections.

6.4.1 Parameter Optimization for Feature Extraction

It has already been mentioned in section 6.3.2 that three different feature sets *namely*, elliptical, gradient based and topological features are extracted from handwritten word images for recognizing them. Of these, length of elliptical feature vector is 65 while other two depend on number of sub-images generated from a word image. Hence, experiments have been conducted for optimal selection of the number of sub-images per word image for gradient based and topological features. For obtaining recognition scores, a 5-fold cross validation using entire database is considered. For recognition purpose a MLP based classifier has been used and the number of neurons in hidden layer is varied from 80 to 180 with step size 5.

Table 6.6 Examples of some cursive word samples reflecting the shape variation present in the prepared database

Sl. #	Word Class	Sample 1	Sample 2	Sample 3
1	আলিপুর			
2	বালুরঘাট			
3	বাঁকুড়া			
4	বারাসাত			
5	বর্ধমান			
6	বহরমপুর			
7	চুঁচুড়া			
8	দার্জিলিং			
9	ইংলিশবাজার			
10	জলপাইগুড়ি			
11	কলকাতা			
12	কৃষ্ণনগর			
13	মেদিনীপুর			
14	পুরুলিয়া			
15	রায়গঞ্জ			
16	সিউড়ি			
17	আগারপাড়া			
18	আজিমগঞ্জ			
19	আসানসোল			
20	অশোকনগর			
21	বৈদ্যবাটী			
22	পলাশী			
23	শান্তিপুর			
24	শিলিগুড়ি			
25	শ্যামনগর			
26	সোদপুর			
27	সোনামুখী			

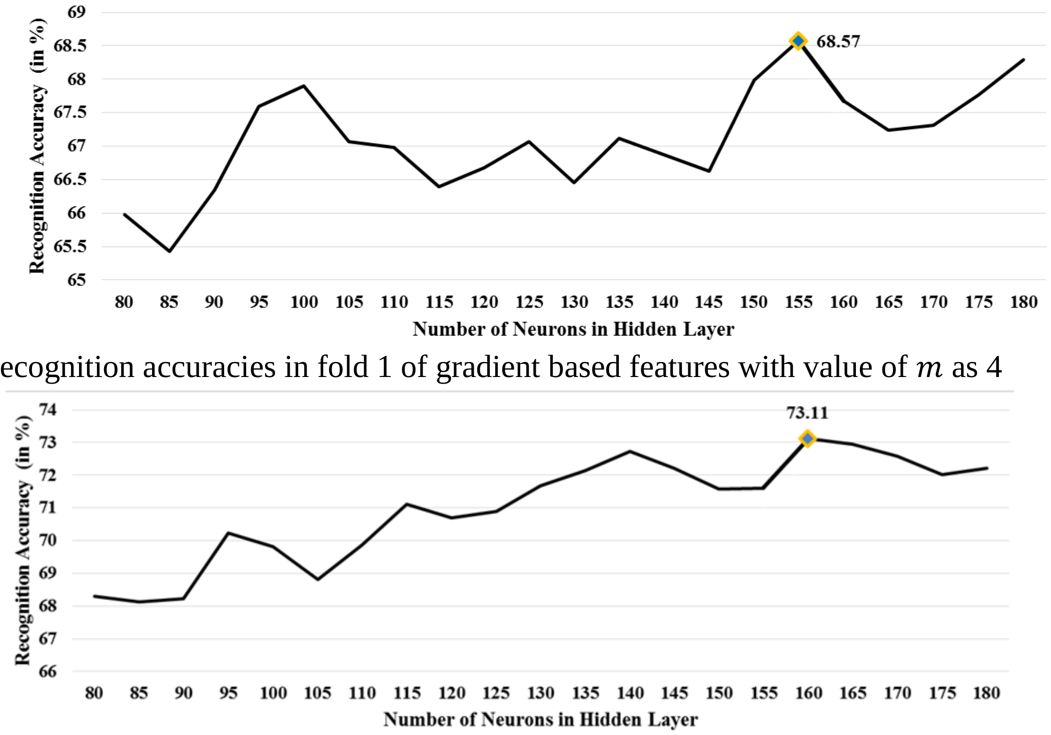
The selection of optimal parameter values of m (number of grids in gradient based feature), p and q (number of grids in topological feature) is carried out here. Here values of m are varied in between 3 to 8. Whereas values of p are varied in between 2 to 4 for preserving the nature of zonal variation inside a word image while the values of q are kept within 3 to 7 for including more local information of the word images. Therefore, first experimental structure leads to 6 number of experiments while the other one requires $4 \times 4 = 16$. The best recognition accuracies, obtained by varying number neurons in hidden layer of MLP based classifier, for all experiments are provided in Table 6.7 and Table 6.8 for selection of gradient based and topological features respectively. One example, showing recognition performances with varying number of neurons in hidden layer, of both the experiments are shown in Fig. 6.13 (a-b). Finally, the parameter values for which best average recognition accuracy is observed are considered for performing later experiments.

Table 6.7 Feature dimension and recognition performances of different experiments with varying number of grids for gradient based feature

m	Feature dimension	Recognition Accuracy (in %)					
		fold 1	fold 2	fold 3	fold 4	fold 5	Average
3	80	60.76	55.46	62.12	60.71	55.57	58.92
4	136	68.56	65.21	71.62	68.71	64.54	67.73
5	208	70.87	66.9	75.46	71.62	67.98	70.57
6	296	71.93	68.79	**76.84**	72.09	69.68	**71.87**
7	400	70.82	68.68	75.82	72.4	69.23	71.39
8	520	70.76	67.65	76.96	72.46	69.46	71.46

Table 6.8 Feature dimension and recognition performances of different experiments with varying number of grids for topological feature

p	q	Feature dimension	Recognition Accuracy (in %)					
			fold 1	fold 2	fold 3	fold 4	fold 5	Average
2	3	113	73.44	71.00	77.19	73.42	70.56	73.12
2	4	143	74.94	71.56	78.47	74.06	72.11	74.23
2	5	173	74.48	72.31	78.39	74.49	73.39	74.61
2	6	203	75.64	72.25	**79.28**	74.58	72.39	**74.83**
2	7	233	73.97	71.44	77.97	73.22	71.39	73.60
3	3	158	72.11	69.72	77.17	72.22	69.14	72.07
3	4	203	74.07	69.69	77.36	73.69	71.03	73.17
3	5	248	73.92	71.17	79.11	74.06	71.22	73.90
3	6	293	73.31	71.72	78.67	73.69	71.53	73.78
3	7	338	73.5	70.39	77.00	72.69	70.78	72.87
4	3	203	71.58	66.53	74.94	70.78	68.58	70.48
4	4	265	72.25	69.58	76.58	71.67	69.17	71.85
4	5	323	73.31	70.78	76.69	73.69	70.42	72.98
4	6	383	72.44	70.72	77.44	73.17	70.11	72.78
4	7	443	71.94	69.22	76.39	72.19	69.03	71.75



(a) Recognition accuracies in fold 1 of gradient based features with value of m as 4

(b) Recognition accuracies in fold 1 of topological features with values of p and q as 2 and 3 respectively

Fig. 6.13 Instances of recognition accuracy with varying number of neurons in hidden layer of MLP

6.4.2 Recognition of Word Images

From the above experiments it is clear that best average recognition accuracies are observed while values of m, p and q are respectively 6, 2 and 6 which implies the lengths of feature vectors are $6 * 6 * 8 + 8 (= 296)$ and $2 * 6 * 15 + 23 (= 203)$ respectively for gradient based and topological features. The length of features after combining elliptical, gradient based and topological feature vectors becomes $65 + 296 + 203 = 564$. This entire feature vector is used to recognize the word samples using 5-fold cross validation schema. Number of neurons in hidden layer of MLP is varied from 80 to 180 with step size 5. The entire experimental results are recorded in Table 6.9. In this table only the best recognition accuracies of each of these experiments are recorded. This table also includes the recognition results for all the elementary feature vectors (i.e., elliptical, gradient based and topological) and their all possible combinations. Also, average and standard deviation (SD) of obtained recognition accuracies (fold-wise) are also shown therein to indicate the variations in fold-wise results. The best recognition accuracy observed is 88.31% when all the three individual feature categories are combined.

Table 6.9 Showing recognition result of final feature set (i.e., after optimally selecting parameters for gradient based and topological features). Bold faced numbers indicate best recognition performances

Feature Set	Recognition Accuracy (in %)						
	fold 1	fold 2	fold 3	fold 4	fold 5	average	SD
Elliptical	55.69	53.78	**58.39**	54.28	51.53	54.73	2.27
Gradient based	71.93	68.79	**76.84**	72.09	69.68	71.87	2.80
Topological	75.64	72.25	**79.28**	74.58	72.39	74.83	2.57
Elliptical + Topological	78.51	76.29	**82.90**	77.54	76.65	78.38	2.39
Elliptical + Gradient based	80.06	77.53	**83.89**	79.75	77.36	79.72	2.36
Gradient based + Topological	82.42	80.56	**87.47**	82.28	80.94	82.73	2.48
Elliptical + Gradient based+ Topological	84.39	82.33	**88.31**	83.64	82.69	84.27	2.14

6.4.3 Results on Each Level of HFS Model

In this section, experimental results in each stage of present HFS model, where MA based FS technique is used as fundamental optimizer, are investigated. The parameter values of MA based FS are listed in Table 6.10. For experimental need, 20% of word images per class of the present word image database are randomly selected as test samples while rest of the samples have been considered for training. In the first stage of the HFS model, all the elementary features (i.e., elliptical, gradient based and topological) are passed through optimizer i.e., intra-feature set based FS is performed. The experimental outcomes of all the intra-feature set based FS are shown in Tables 6.11, 6.12 and 6.13. The best recognition results obtained for elliptical, gradient based and topological features are 57.44%, 78.02% and 78.18% respectively. These recognition accuracies are found with feature length 61 (for elliptical feature), 290 (for gradient based feature) and 180 (topological feature) respectively.

Table 6.10 The parameter values that have been used in the present MA based FS technique

Parameter	Notation	Value
Probability of mutation	$P(M)$	0.1
Size of population	z	10
Number of generations	g	10
Weight for recognition accuracy	$w1$	1
Weight for feature reduction	$w2$	10
Maximum recognition accuracy which can be compromised	α	0.1%
Number of neurons in hidden layer of MLP	-	100

Table 6.11 Performance of all the chromosomes in the final population of the MA based FS for elliptical feature. Bold faced numbers indicate best performances and RA stands for recognition accuracy

Before Feature Selection		**After Feature Selection**			
Size of feature vector	**RA (in %)**	**Size of feature vector**	**RA (in %)**	**Improvement in RA (in %)**	**Reduction in feature length (in %)**
65	56.48	61	57.44	0.96	06.15
		64	57.00	0.52	01.54
		61	56.89	0.41	06.15
		57	56.67	0.19	12.31
		62	56.61	0.13	04.62
		61	56.47	-0.01	06.15
		61	56.39	-0.09	06.15
		62	56.19	-0.29	04.62
		64	56.19	-0.29	01.54
		58	56.17	-0.31	10.77

Table 6.12 Performance of all the chromosomes in the final population of the MA based FS for gradient based feature. Bold faced numbers indicate best performances and RA stands for recognition accuracy

Before Feature Selection		**After Feature Selection**			
Size of feature vector	**RA (in %)**	**Size of feature vector**	**RA (in %)**	**Improvement in RA (in %)**	**Reduction in feature length (in %)**
296	75.19	290	78.02	2.83	2.03
		283	77.82	2.63	4.39
		287	77.71	2.52	3.04
		289	77.63	2.44	2.36
		283	77.60	2.41	4.39
		270	77.49	2.30	8.78
		285	77.43	2.24	3.72
		287	77.43	2.24	3.04
		282	77.38	2.19	4.73
		283	77.27	2.08	4.39

Table 6.13 Performance of all the chromosomes in the final population of the MA based FS for topological feature. Bold faced numbers indicate best performances and RA stands for recognition accuracy

Before Feature Selection		**After Feature Selection**			
Size of feature vector	**RA (in %)**	**Size of feature vector**	**RA (in %)**	**Improvement in RA (in %)**	**Reduction in feature length (in %)**
203	77.67	180	78.18	0.51	11.33
		180	78.07	0.40	11.33
		182	77.99	0.32	10.34
		191	77.79	0.12	05.91
		181	77.77	0.10	10.84
		191	77.71	0.04	05.91
		192	77.71	0.04	05.42
		176	77.66	-0.01	13.30
		182	77.63	-0.04	10.34
		182	77.57	-0.10	10.34

In the final stage of the said HFS, first the best performing feature sets from the preceding stage are combined and then this combined feature vector is passed through MA based FS for further optimization. The result of this stage is listed in Table 6.14. Result indicates that the MA based FS technique not only improves the recognition accuracy but also decreases the feature dimension satisfactorily.

Table 6.14 Performance of all the chromosomes in the final population of the MA based FS for feature generated after combining best feature sets from Table 6.11, Table 6.12 and Table 6.13. Bold faced numbers indicate best performances and RA stands for recognition accuracy

Before Feature Selection		**After Feature Selection**			
Size of feature vector	**RA (in %)**	**Size of feature vector**	**RA (in %)**	**Improvement in RA (in %)**	**Reduction in feature length (in %)**
531	87.12	507	89.55	2.43	04.52
		526	89.50	2.38	00.94
		504	89.39	2.27	05.08
		513	89.36	2.24	03.39
		519	89.25	2.13	02.26
		470	89.19	2.07	11.49
		526	89.11	1.99	00.94
		495	89.08	1.96	06.78
		519	89.05	1.93	02.26
		478	89.02	1.90	09.98

To investigate the effectiveness of present HFS, the entire feature set is passed through the said MA based FS technique (i.e., single stage FS) for optimization and the experimental outcomes have been shown in Table 6.15. The results, tabulated in Tables 6.15 and 6.15, show that the present HFS technique not only produces better recognition accuracy (i.e., 1.13% (2.43%-1.30%) more) but also reduces the dimension of the feature vector (i.e., 45 (552-507) more) than the single stage FS.

Table 6.15 Performance of all the chromosomes in the final population of the MA based FS for all combined non-optimized features. Bold faced numbers indicate best performances and RA stands for recognition accuracy

Before Feature Selection		**After Feature Selection**			
Size of feature vector	**RA (in %)**	**Size of feature vector**	**RA (in %)**	**Improvement in RA (in %)**	**Reduction in feature length (in %)**
564	86.93	552	88.23	1.30	02.13
		534	87.98	1.05	05.32
		545	87.98	1.05	03.37
		518	87.95	1.02	08.16
		475	87.90	0.97	15.78
		512	87.87	0.94	09.22
		497	87.84	0.91	11.88
		545	87.76	0.83	03.37
		501	87.67	0.74	11.17
		511	87.67	0.74	09.40

6.4.4 Result on Optimized Feature Set

From the above results, shown in Tables 6.14 and 6.15, it is clear that present HFS performs better in terms of feature dimension reduction with improved recognition accuracy. The length of optimized feature vector for the current HFS model is 507. Now, this optimal feature set for each of the word image samples of the present database is considered while designing a MLP based classifier using 5-fold cross validation schema. The recognition results are recorded in Table 6.16. In addition, the optimized elliptical, topological and gradient based feature vectors are also taken from the word images and tested in the same way. These results (recognition accuracies) are shown in Table 6.17. Along with these, to describe the usefulness of underlying FS a comparison summery is shown in Table 6.18. In this table the best recognition accuracies, before and after applying FS, of each fold are recorded for each of the elementary feature vectors *viz.* elliptical, gradient based and topological and their combined features. From this table it can be concluded that the present HFS model not only improves the recognition accuracies with reduced feature dimension but also diminishes the variation in recognition accuracy among different folds for all the cases.

Table 6.16 Performances of all optimized elementary feature sets and the combined feature sets generated during present HFS model

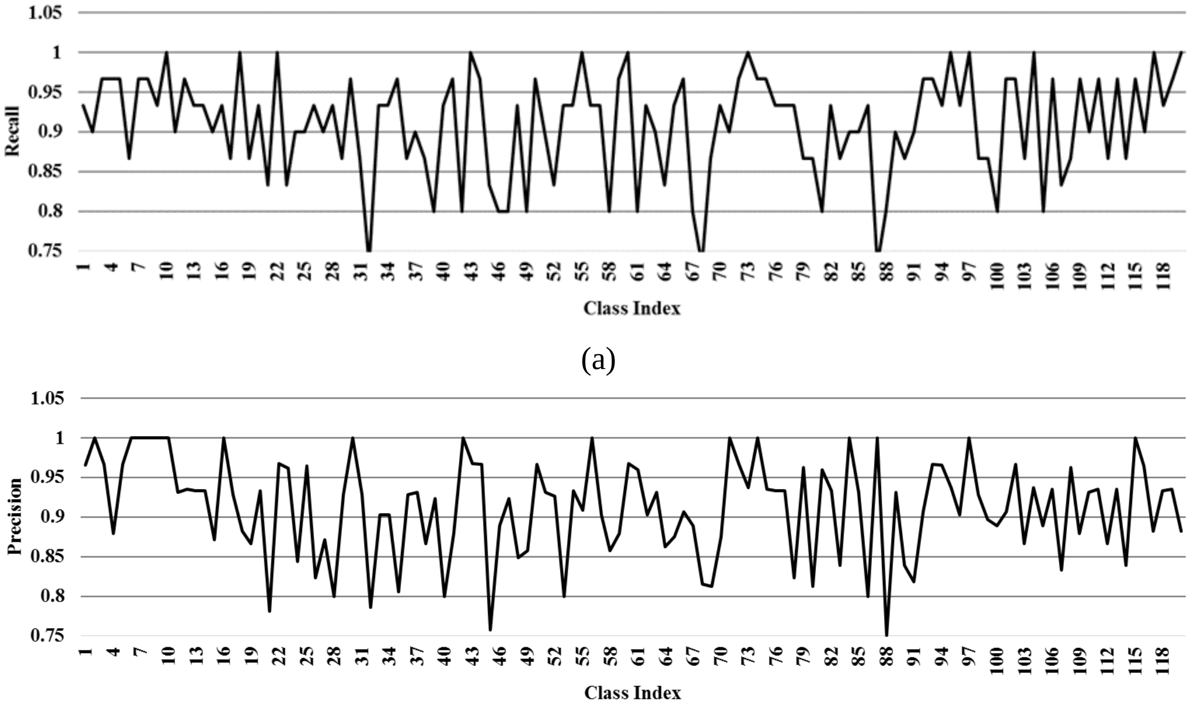
Feature Set (optimized)	**Recognition Accuracy (in %)**						
	fold 1	**fold 2**	**fold 3**	**fold 4**	**fold 5**	**average**	**SD**
Elliptical	59.67	58.67	62.78	61.36	57.33	59.96	1.93
Gradient based	76.89	73.97	81.08	76.00	74.11	76.41	2.59
Topological	76.72	74.08	80.69	75.75	74.61	76.37	2.35
Combining all	87.25	86.03	90.94	85.94	86.94	87.45	1.89

Table 6.17 Comparative performances (with and without applying MA based FS) of all the elementary feature set and combined feature set

Feature used	**Recognition Accuracy (in %)**						
	fold 1	**fold 2**	**fold 3**	**fold 4**	**fold 5**	**average**	**SD**
Elliptical Feature							
Without FS	55.69	53.78	58.39	54.28	51.53	54.73	2.27
With FS	59.67	58.67	62.78	61.36	57.33	59.96	1.93
Gradient based Feature							
Without FS	71.93	68.79	76.84	72.09	69.68	71.87	2.80
With FS	76.89	73.97	81.08	76.00	74.11	76.41	2.59
Topological Feature							
Without FS	75.64	72.25	79.28	74.58	72.39	74.83	2.57
With FS	76.72	74.08	80.69	75.75	74.61	76.37	2.35
Combined Elliptical, Gradient based and Topological Features							
Without FS	84.39	82.33	88.31	83.64	82.69	84.27	2.14
With FS	87.03	85.81	90.94	86.36	85.43	87.11	1.99

From the Table 6.16 and Table 6.17, it can be found that the best, worst and average recognition accuracies as achieved are 90.94%, 85.43% and 87.11% respectively. To analyze the recognition results in a better way, two measures related to class-wise true classification score *namely*, true positive rate (TPR) i.e., recall and positive predictive value (PPV) i.e., precision [64] are considered. Precision and recall of i^{th} $(i = 1, 2, \dots, 120)$ class (say, p_i and r_i respectively) are calculated by the formula $p_i = \frac{T_i}{\mathcal{M}_i}$ and $r_i = \frac{T_i}{\mathcal{N}_i}$ respectively. Here, the parameters T_i, $\mathcal{M}_i$ and $\mathcal{N}_i$ indicate number of word images of i^{th} class is recognized as i^{th} class, number of word samples that are classified as i^{th} class and number of actual word images in i^{th} class respectively. Class-wise recall and precision are shown in Fig. 6.14 (a-b).

From Fig. 6.14 (b), it is clear that class-wise best recall is 1.00 which is obtained for 12 classes (class index are 10, 18, 22, 43, 55, 60, 73, 95, 97, 104, 117, 120) i.e., 10% of the word classes achieved 100% recognition. Whereas only 3 classes (class index are 32, 68 and 87) provide worst class-wise recall which is here $\frac{22}{30} = 0.77$.



(a)

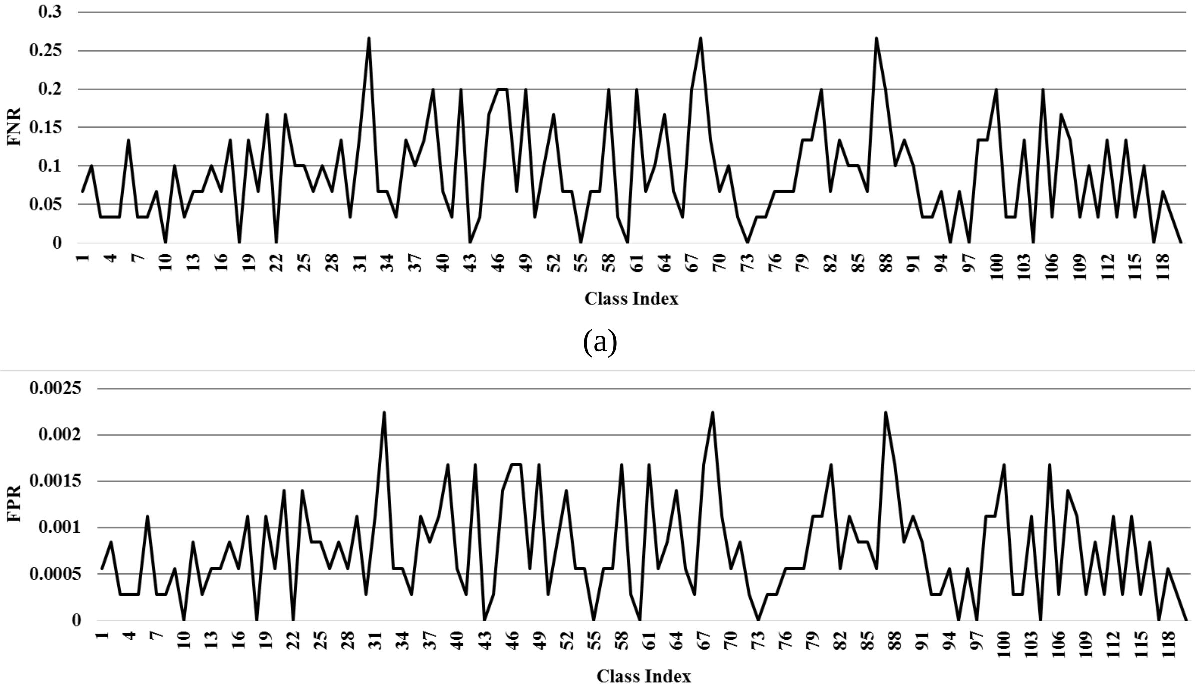
(b)

Fig. 6.14 Depiction of class-wise (a) Recall and (b) True Classification Rate

6.4.5 Error Case Analysis

Though the proposed HFS based handwritten word recognition method has satisfactorily recognized most of the pattern classes, but some misclassification of the word samples are also found. The possible reasons behind such misclassification are analyzed statistically in the current sub-section. The analyzes are performed based on two statistical error analysis measures *namely*, false negative rate (FNR) or miss rate and false positive rate (FPR) or fall out [64]. FNR indicates number of samples of a particular class is misclassified as other (remaining) classes while the FPR represents the number of samples belonging to other classes is misclassified as pattern class under consideration. Class-wise FNR (say, $FNR_i, i = 1, 2, \dots, 120$) is defined as $FNR_i = \frac{\zeta_i}{\kappa_i}$, where ζ_i and κ_i are the number of samples misclassified as i^{th} class and the number of samples classified as i^{th} class respectively. Whereas class-wise FPR (say, $FPR_i, i = 1, 2, \dots, 120$) is calculated as $FPR_i = \frac{\eta_i}{\xi_i}$, where η_i and ξ_i are the number of samples of i^{th} class that is not classified as i^{th} class and total number of samples beyond i^{th} class respectively.

The graphical representations of class-wise FNR and FPR are depicted in Fig. 6.15(a-b). From this figure Fig. 6.15(a), it is found that samples belonging to 32th class are mostly misclassified as other classes while samples that belong to classes other than 88th class are mostly classified as 88th class. Information of the associated classes that perform poorly in terms of FNR and FPR are tabulated in Table 6.18 and Table 6.19 respectively. The word classes that are associated with such poor performance are depicted in these tables. These results reveal that the main reason for misclassifications are shape similarity among these classes.



(a)

(b)

Fig. 6.15 Illustration of class-wise (a) FNR and (b) FPR

Table 6.18 Sample word classes that are significantly contributed to poor FNR performances

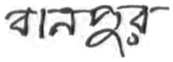
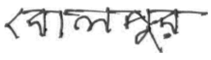
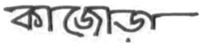
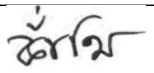
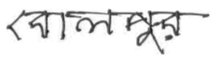
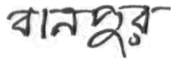
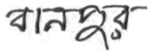
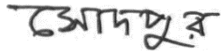
Actual Class		**Classified as**		**Number of samples misclassified**
Class Index	**Sample word image**	**Class Index**	**Sample word image**	
32		45		4
78		87		4
83		85		3

Table 6.19 Sample word classes that are significantly contributed to poor FPR performances

Actual Class		**Classified as**		**Number of samples misclassified**
Class Index	**Sample word image**	**Class Index**	**Sample word image**	
88		45		2
45		32		4
32		112		2

6.4.6 Performance Comparison

Performance comparison is an essential part of any newly designed research outcome to indicate its usefulness in global perspective. In this work, the present research outcomes are compared with the *state-of-the-art* methods with respect to design of feature descriptor and method. The comparative results are discussed in the following sub-sections.

6.3.6.1 Comparison with State-of-the-art Feature Descriptors

The one of the primary motives of this work is to introduce a feature descriptor which recognizes handwritten word images holistically in a better way. For that reason, in the present work, the feature descriptors used here is compared with some *state-of-the-art* feature descriptors along with the elementary feature sets used here. Comparison with the elementary features along with their possible combinations are already provided in Table 6.9. To evaluate the recognition performance excluding these elementary ones (i.e., elliptical, gradient based and topological), 5-fold cross validation mechanism is employed having MLP as sole classifier. Feature descriptors considered for comparison include *viz.*, Topological [86], Convex hull based [26], Statistical and contour-based feature [178], Tetragonal [73], Shape-context features [191], Local Gradient of Histogram (LGH) [186], Pyramid Histogram of Oriented Gradient (PHOG) [175], combination of GABOR and PHOG called G_PHOG [204]. The comparative results are tabulated in Table 6.20. From this table it can safely be commented that proposed feature descriptor outperforms *state-of-the-art* feature descriptors.

6.3.6.2 Comparison with State-of-the-art Holistic Word Recognition Techniques

Finally, the proposed method is compared with some of recently published holistic word recognition methods [26, 73, 84-86, 180, 189-193]. Except the works in [180] and [86], which have dealt with handwritten English and Hindi word recognition respectively, all the works have been implemented on the *Bangla* dataset. However, only the work [73] have been performed on the entire dataset previously. Rest of the works have either considered less number of sample classes from this database (e.g., 20 classes in [84], 40 classes in [190], 50 classes in [193], 80 classes in [191-192]) or have been experimented on different dataset (e.g., 20 most popular city names of West Bengal [189], most frequent words from CMATERdb1 in [26], Capital names of states and territories of India written in Devanagari script in [44], *Bangla* common name words in [85]). The best, worst, average cases of recognition accuracies in these experiments along with feature information and the classifier used by these techniques are summarized in Table 6.21. It also includes deviation from average recognition rate. From the table it is clear that the present technique outperforms the said methods.

Table 6.20 Performance comparison of proposed feature descriptor with the *state-of-the-art* feature descriptors on present database

Feature Descriptor	**Dimension**	**Recognition Accuracy (in %)**			
		Best	**Worst**	**Average**	**SD**
Convex hull features	140	76.53	67.64	72.63	2.92
Topological	89	72.53	66.32	69.22	2.19
Tetragonal	185	76.42	70.00	72.74	3.27
Shape context features	64	65.73	57.60	60.51	2.98
Statistical and contour-based feature	220	74.43	65.80	68.85	3.08
LGH	786	83.40	76.10	78.55	2.91
PHOG	672	78.42	73.47	75.18	2.97
G_PHOG	720	72.53	67.14	68.93	3.08
Proposed (without FS)	564	88.31	82.33	84.27	2.14
Proposed (with FS)	507	**90.94**	**85.81**	**87.11**	**1.99**

6.5 Discussion

Handwritten word recognition is an important and an interesting research problem which is solved using either segmentation based approach or holistic approach. For complex regional script like *Bangla* word recognition in segmentation based approach, which needs character segmentation, becomes almost impossible. In addition, unavailability of freely accessed databases is one of the reasons for the slow progress in this research area. Considering these facts, in this chapter, a HWR system, following holistic approach, for handwritten Bangle word images is developed.

Table 6.21 Comparison of the current handwritten word recognition system with *state-of-the-art* methods

Method	**Feature Description**	**Feature Length**	**Classifier used**	**Recognition accuracy (in %)**			
				best	**worst**	**average**	**SD**
Dasgupta et al., [180]	Arnold transform based and naïve directional features	620	SVM	76.69	68.56	72.71	2.81
Bhowmik et al. [84]	Elliptical	65	MLP	58.39	51.53	54.73	2.27
Bhowmik et al. [85]	Basic HOG features	80		69.82	62.93	65.31	2.46
Bhowmik et al. [26]	Convex hull features	140		72.53	64.64	68.63	2.92
Malakar et al. [86]	Topological feature	89	Sequential minimal optimization (SMO)	73.03	67.42	69.72	2.11
Barua et al. [189]	Gradient based	288		83.06	75.75	79.06	2.72
Sahoo et al. [191]	Shape context features	64	Classifier ensembling with SMO, Simple Logistics and CVParameter selection [68] as classifier	66.53	58.18	60.97	2.91
Sahoo et al. [192]	Shape based features extracted from negative refraction based image transformation	186		82.23	75.64	78.63	2.31
Bhowmik et al. [73]	Elliptical, vertical pixel density histogram based and tetragonal	252	SVM	83.64	77.19	79.38	2.33
Ghosh et al. [190]	Gradient based	288 (272)*	MA based FS having MLP as classifier	80.08	73.91	76.21	2.25
Ghosh et al. [193]	Gradient based features and statistical contour based	508 (412)*		81.88	75.89	78.27	2.17
Malakar et al. [88]	Gradient based and elliptical	361 (319)*	GA based HFS having MLP as classifier	86.82	81.13	83.00	2.08
Proposed	Elliptical, gradient based and topological	561 (507)*	MA based HFS having MLP as classifier	**90.94**	**85.81**	**87.11**	**1.99**

* The feature dimension presented in format: actual feature length (optimized feature lengths). The optimized features are used to generate recognition score.

To recognize word images holistically 3 different elementary features *viz*., elliptical, gradient based and topological are considered and all these feature vectors are passed through a HFS model where MA based FS is the fundamental optimizer to obtain optimized feature vector that represents the underlying dataset in a better way. Also, to fill the shortfall of database, in this work, a database comprising most popular city names of West Bengal, India is prepared. The HWR system is employed on the entire database. The recognition accuracy, as found here, is 90.94% which outperforms some *state-of-the-art* holistic HWR methods on the said database.

Chapter 7

KEYWORD BASED HANDWRITTEN *BANGLA* DOCUMENT IMAGE CLASSIFICATION

7.1 Introduction

The classification problem of machine encoded text documents, like short messages, electronic mails and web content applications, has been widely studied by the data mining, machine learning, and information retrieval communities [66]. The main objective of such document classification is to group documents into suitable and predefined categories. This classification problem is associated with two major issues *viz.*, (i) voluminous text documents and large number of users and (ii) diversity of the contents and users with varied interests. Handling the diverse interest of user, in general, is more critical than its counterpart [205], since user choice is most important but it is in most of the cases imprecise. As a result, the researcher have introduced personalized document classification where users create their own choice of categories and the classifiers are trained to classify these categories automatically.

The processes that are used in literature for personalized document classification (or clustering) are either content-based or request-based [206]. In content-based classification, weight is assigned to particular subjects (i.e., keywords / key terms / key phrases) in documents in corpora that determine the class of the document. This type of document classification could be done based on the number of times a given keywords appears in a document [207]. On the other hand, request-based classification (or indexing) classifies document to anticipate request from users. In general, irrespective of the methods of classification, weight to a document is given depending on presence/frequency of some keywords (also sometime key term/key phrase is considered). The keywords are normally selected during actual classification using some algorithms [66].

In handwritten document classification, deciding keywords for some predefined categories of document is still near impossible using the way it is carried out in machine encoded textual document [208]. The major reason behind this is poor performance of current handwritten OCR engines for large lexicon size [73]. Therefore, the alternative solution is to classify the documents using user given keyword. However, the prerequisite for such document classification is searching a keyword, provided by user, from a handwritten document. This need gives rise to new research problem, termed as word/keyword searching/spotting from/in a handwritten/printed document images, inside the research community. The keyword spotting is already used in a number of real life applications such as:

- Retrieval of documents containing specific keyword(s) from large-scale document files in company, office, institutions etc. [44].
- Sorting of handwritten mails based on keywords like “urgent”, “cancellation”, “complain” etc. [64].
- Marking of figures and their corresponding captions [65].
- Retrieval of pre-hospital care reports (PCR forms) based on predefined keyword [209].
- Spotting of word in graphical documents such as maps [210].
- Retrieval of cuneiform structures from ancient clay tablets [211].

Considering the above facts, it can safely be commented that keyword searching from document images plays vital role for classification/indexing of the same. A word (keyword) searching methodology tries to locate the occurrence(s) of some predefined words in a document image. The word spotting/ searching mechanisms, can be applied on either printed [212] or on handwritten [67] document images. It is obvious that searchin*g* a word in a handwritten document poses more challenges than searching the word from its printed counterpart. One of the key factors for this is the variation in writing styles, not only among different writers but also for the same writer at different times. Also some typical problems of handwriting like skew, slant, overlapping and/or joining of words/characters add complexities in keyword searching/spotting in a handwritten document image.

7.1.1 Classification of Word (Keyword) Searching Methods

In the literature on word (keyword) searching, the methods are found to differ on implementation aspects like (i) how a query word is fed to a word searching system, (ii) whether a keyword is searched from whole page without applying page segmentation or from segmented components like text line (TL) and word and (iii) how a target word is matched with a query word. Brief description of word searching techniques based on these three aspects is provided in this section.

A typical way of categorizing word searching techniques is based on how query words are fed into the system. Based on this, existing techniques are classified into two categories *namely*, query-by-example (QBE) [44, 65] and query-by-string (QBS) [44, 213]. In the former category, a word image is provided to the system and the system returns all occurrences of that given word in the document image. Methods of this approach have mostly relied on image matching techniques in an unsupervised way. On the other hand the techniques that follow QBS approach consider an arbitrary word as a string which needs to be searched from a document image. Therefore these methods, in general, need model for every character present in the script in which it is written. Consequently, these methods try to follow some classical word recognition model (detail description is provided in section 7.1.2).

Not only these, based on processes followed in page segmentation algorithms [46, 57], word searching techniques can also be categorized as segmentation based approach [45] and segmentation free approach [72]. The first category of approaches has some in-built page segmentation methods to obtain pre-segmented text lines (TLs) [45] or words [214]. Some segmentation-based word spotting methods consider that datasets are already segmented into text lines/words i.e., these works use pre-segmented TL [215-216] and / or word [216] which may be achieved by using TL and/or word level ground truth (GT) images. Whereas, the other category of techniques (e.g., refer to [72, 217]) tries to locate the words to be searched in document images without spending time for page segmentation.

Use of multiple reference sample images for a given query word is found in recognition based word spotting techniques [69, 71]. This reference sample images are collected in offline mode for preparing training module. On the other hand, recognition free approaches [68, 218] try to spot search words using some matching technique. The present work is an instance of recognition based approach which uses holistic word recognition technique that has already been described in previous chapter.

7.1.2 Literature Survey

From the above discussion it is clear that the fundamental need for any document image classification is a searching technique that can search keywords (or terms) in the document i.e., document classification requires a keyword spotting method that performs satisfactorily. Therefore, in this section few researches on keyword spotting are studied. Moreover, discussion in pervious subsection states that word searching techniques could be categorized in different ways. But, here only some methods that follow QBE and/or QBS approaches are discussed.

7.1.2.1 QBE based Word Searching Methods

From the literature survey, it is observed that, till date, many researchers have applied several distance based methods [70, 216, 219-220] for searching a word from document image following QBE way. The work [70] has described a word searching mechanism which extracts query word from historical document images. For searching of keywords, it has used scale-invariant feature transform (SIFT) features from word images and search has been confirmed by cosine and Euclidean distance based similarity measures. Whereas the work, described in [219], has extracted Gabor features from the word images and then used Euclidian distance based similarity check for word searching in QBE way. In another work [216], in order to handle the said problem, gradient angle and its magnitude have been extracted from word images to obtain query words using a disc based matching schema. Authors of this article have also prepared a historical document database, called HADARA80P dataset, and made the same freely available to research community along with TL and word level GT information. Pixel value information has been used in [220] where word searching in QBE way has been carried out by Bray-Curtis dissimilarity measure.

In another category of works [68, 218, 221], several string matching approaches have been used for finding solution to the said problem. Multi angular feature descriptor has been used in [221] where word images are represented by variable length feature vector. Dynamic time warping (DTW) is used for matching the words. The work described in [68] has introduced a flexible sequence matching (FSM) technique for comparing query word and target word. In this work, 8 feature values (1 from gray image and 7 from binarized image), that are extracted from each vertical line of the word image, have been used. The same set of feature values are used in [218] for investigating effectiveness of some conventional time series matching techniques for word searching from handwritten and historical document images. In this work, it has been shown that continuous dynamic programming (CDP) provides best score.

Graph similarity based methods [45, 222-223] are also found in literature of word spotting. In the work [222], attributed graphs have been constructed using the graphemes that are extracted from words by a part-based approach. It has employed edit distance based similarity measure.

Authors in [45, 223] have used graph edit distance based similarity score for word spotting. In [223], each word image is represented as a sequence of skeleton-based graphs using context labeled vertices for connected components (CCs). A similar approach has been found in [45]. The only difference lies there is the way calculation of graph edit distance is done i.e., in [223], bipartite graph matching schema has been used, while in [45] DTW alignment method has been used.

7.1.2.2 QBS based Word Searching Methods

Classical word recognition model are used in [64, 213, 220] for performing word spotting in QBS way. In the work [220], pyramidal histogram of characters (PHOC) features have been extracted for performing word searching in QBS way. Bray-Curtis dissimilarity measure between target word and query word has been measured for performing word similarity task. Geometrical features like number of object pixels, center of gravity, second order moments, positions of the upper and lower contours, gradients of the upper and lower contours, number of object pixel to background pixel transitions and fraction of object pixels between the upper and lower contours have been extracted from a word image in [213] for performing word spotting. HMM based character level trained model has been applied for finding similarity score in this work. A statistical framework for the word spotting problem, introduced in [64], has explored the use of two types of HMMs *namely*, continuous HMMs (C-HMMs) and semi-continuous HMMs (SC-HMMs).

Researchers have also attempted to devise techniques [72, 215, 217, 224] where they have segmented a word into characters or character sub-parts. In the work [224], water reservoir based character segmentation model is used to extract primitive components (i.e., character or character subpart) from historical printed document written in French. Next these extracted primitives are clustered using cross correlation based template matching technique. Finally, document image and query image are represented as string using the cluster information to perform word searching in QBS way. The authors of the work [72] have used local image feature representatives called SIFT descriptor for recognition purpose in their Bag-of-Features (BoF) based HMMs (BoF-HMM) model. An approach similar to [72] has been proposed in [217]. In this case, the model has been developed to search any arbitrary word and does not require any pre-segmentation of document pages. A semi-supervised handwritten word recognition based technique using bi-directional long short-term memory (BLSTM) model is presented in [215]. In this work, authors have used TL based model to extract the query words i.e., pre-segmented TLs are fed as the input to their word searching system.

Holistic word recognition paradigm is used in [69, 225] for spotting of search words from historical Arabic handwritten manuscripts. In [69], several structural and statistical features, extracted from connected parts of word images, are fed into Multi-Layer Perceptron (MLP) based classifier for preparing a learning module. In this work, it has been shown that features extracted from connected parts of the word perform well in comparison with features extracted from entire word. The work, described in [225], has introduced a hierarchical classifier, comprising of Support Vector Machine (SVM) and Regularized Discriminant Analysis (RDA), for searching a word written in Arabic script. The said classifiers are employed in a sequential manner. A gradient based feature vector is used for discriminating the words in feature space.

7.1.3 Motivation

Literature survey as studied in section 7.1.2, reveals that mainly two different approaches *viz.*, recognition free [216, 220] and recognition based [215, 224], are followed by the researchers for providing solution to keyword searching. The first category of works uses geometrical/shape characteristics of word image (e.g., [68, 216, 218]) and then applies some similarity measure for word matching. Though, this mechanism is fast, it generally retrieves more irrelevant words with respect to actual search word. On contrary, matching techniques as applied in recognition based methods (e.g., [69, 72, 225]) have tried to identify a query word in the document images by recognizing all the words present therein. Therefore, these mechanisms not only suffer from high time requirement but also, come up with more irrelevant words retrieval.

Considering the above facts, in the present work, a two-stage word searching technique, similar to the work described in [71] is introduced. This work first employs a word selection technique following recognition free keyword spotting approach to accept some target words as probable candidate keyword and then matching schema which relies on recognition based model. However, the recognition based schemas [215, 224] for matching between query word and target word, found in literature, mostly follow classical approaches of word recognition which are based on character / language model [64-65]. Therefore, these works need to employ character extraction technique prior to recognition (e.g., [64, 213]) which is sometimes erroneous and complex in nature [63, 83]. Hence holistic paradigm for matching a target word with keywords is considered here.

In addition to these, most of the abovementioned researches (e.g., [45, 69, 72, 214, 216-217]) have described document image classification/indexing as an application of keyword spotting. But, no significant work on classification of document images has been found. This may be due to more number of target words retrieval from document images than expected or not having suitable dataset. Therefore, in this chapter a suitable database which is developed during this thesis work along with a machine learning based handwritten document image classification technique have been described.

7.1.4 Objective

Based on the above discussion, following objectives have been covered in this chapter:

- Design of an efficient word searching technique to retrieve keywords (user given) from handwritten document page images.
- Introducing a handwritten document classification technique based on the user provided keywords to classify the documents into some pre-defined categories.
- Preparing a comprehensive database to accomplish all the related experiments.

7.2 Keyword Searching and Document Classification

In the present work, a personalized keywords set based handwritten document classification method is devised. The task has followed two major modules *viz.*, i) keyword searching from handwritten document page images and ii) classifying input document pages into predefined categories. To implement the first module all the components *viz.*, words and punctuation marks, present in a document page, are extracted using CC based page-to-word extraction technique, described in Chapter 4, and that are hereafter termed as target word. Next, these target words are matched with the search word, which is here a keyword, to decide whether the underlying target word is the desired keyword or not. For the matching purpose here a two-stage technique, described in section 7.2.1, has been used. Whereas for the later module, first a set of relevant keywords for each of the personalized document categories (see Chapter 2) is selected and then number of these keywords is counted using the present word searching technique. Finally, classification is done relying on feature vector, prepared using counts of keywords, and MLP based classifier. The overall working procedure of the proposed technique is depicted in Fig. 7.1.

7.2.1 Keyword Searching

It has already been mentioned that mainly two different approaches *viz.*, recognition free and recognition based, are followed by the researchers for developing a word searching method. In the present work, a two-stage approach is designed where both of these concepts are combined for searching a given word in handwritten document images. That means, first some of the target words, those appear in a handwritten document page image, are selected as probable candidate keywords and then these probable candidate keywords are either confirmed as query keyword or rejected. In the first stage, irrelevant target words (with respect to a keyword to be searched) from a document image are filtered out. In the second stage, the probable candidate keywords are confirmed as given query word image (i.e., keyword). Detailing of these stages is performed in following subsections.

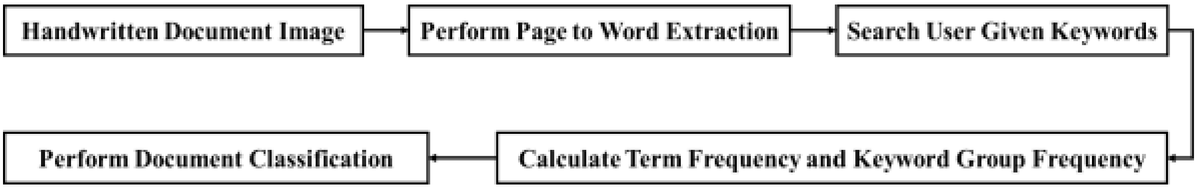

Handwritten Document Image
Perform Page to Word Extraction
Search User Given Keywords
Calculate Term Frequency and Keyword Group Frequency
Perform Document Classification

Fig. 7.1 Block diagram of the present handwritten document classification technique

7.2.1.1 Extraction of Probable Candidate Keywords from Document Images Corresponding to a Keyword

Words in a document, generally, have varying number of characters associated with punctuation marks like comma (‘,’), periods (‘.’), hyphen (‘-’). Apart from this, the variation in number of characters among keywords is very common. In this scenario, searching a word directly in a document image increases the chances of mismatch as well as it enhances computational cost. Sometimes, it leads to selection of irrelevant words as search word. That is why, in the present work, an initiative has been taken to filter out all the irrelevant target words (i.e., words having different number of characters than search word under consideration), from document images. For doing this, a feature vector (i.e.,$\mathfrak{F}_1$) comprising of 4 zonal information such as Mean of transition points (MTP) and number of characters or character like shapes in middle zone and number of dominant CCs in upper and lower zones are extracted. Then probable candidate keywords are chosen using a rule based method. In this section, first feature extraction method and then formulation of required rules for decision making are discussed. The entire selection process is illustrated Fig. 7.2.

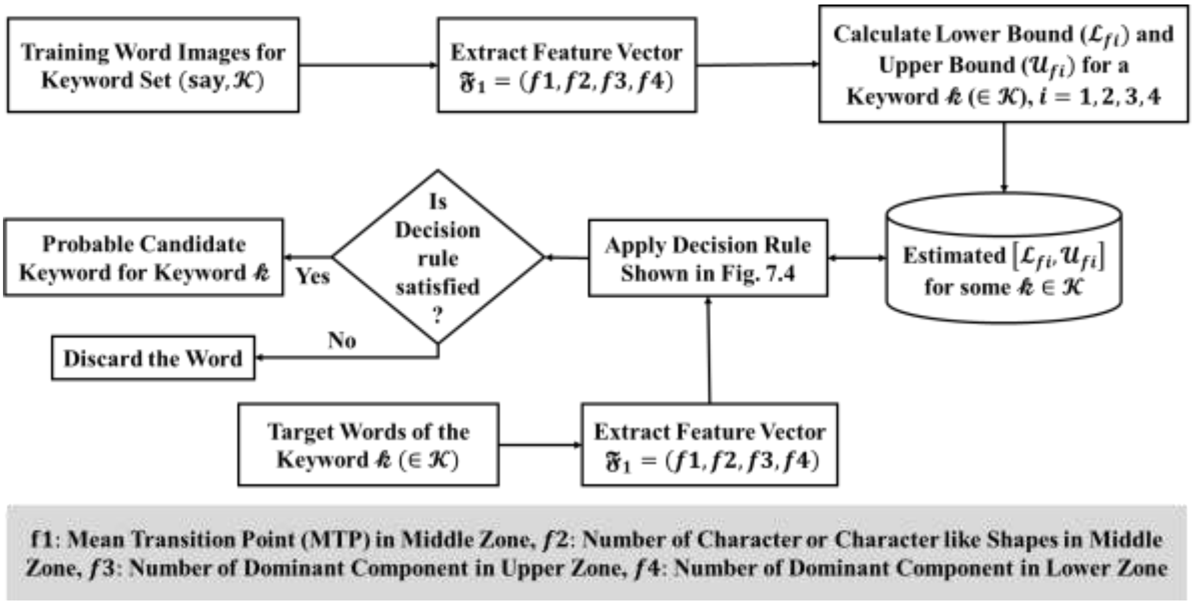

Training Word Images for Keyword Set (say, $\mathcal{K}$)
Extract Feature Vector $\mathfrak{F}_1 = (f1, f2, f3, f4)$
Calculate Lower Bound ($\mathcal{L}_{fi}$) and Upper Bound ($\mathcal{U}_{fi}$) for a Keyword $\mathfrak{k}$ ($\in \mathcal{K}$), $i = 1,2,3,4$
Estimated $[\mathcal{L}_{fi}, \mathcal{U}_{fi}]$ for some $\mathfrak{k} \in \mathcal{K}$
Apply Decision Rule Shown in Fig. 7.4
Is Decision rule satisfied ?
Yes
No
Probable Candidate Keyword for Keyword $\mathfrak{k}$
Discard the Word
Target Words of the Keyword $\mathfrak{k}$ ($\in \mathcal{K}$)
Extract Feature Vector $\mathfrak{F}_1 = (f1, f2, f3, f4)$
f1: Mean Transition Point (MTP) in Middle Zone, $f2$: Number of Character or Character like Shapes in Middle Zone, $f3$: Number of Dominant Component in Upper Zone, $f4$: Number of Dominant Component in Lower Zone

Fig. 7.2 Schematic diagram for preselecting probable candidate keywords in a document page that are relevant to a given keyword

7.2.1.1.1 $\mathfrak{F}_1$ Extraction

To extract zonal information based feature vector $\mathfrak{F}_1 = (f1, f2, f3, f4)$ first the word images are converted to its binarized form using the method described in Chapter 2. Let, a binarized word image is represented as $B = \{f(i,j): 1 \leq i \leq H \land 1 \leq j \leq W\}$, where H and W are height and width of B respectively and$f(i,j) \in \{0,1\}$ (‘0’ and‘1’represent background and object pixels respectively). Next, B is hypothetically segmented into 3 non-overlapping horizontal regions *viz.*, upper, middle and lower zones, as shown in Fig. 7.3, for feature extraction. Algorithm 5.1, mentioned in Chapter 5, is used here to estimate the zonal boundaries *viz.*,$R1$, $R2$, $R3$ and $R4$. The position of zonal boundaries are shown in Fig. 7.3. Finally, feature values from different zones are calculated using the procedure as described below.

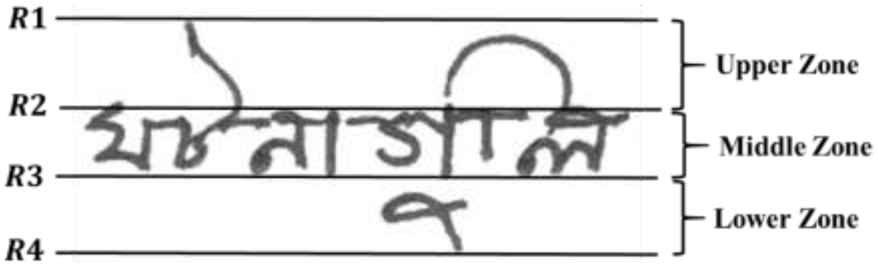


Fig. 7.3 Partitioning of a word image into three zones

To find the MTP in middle zone of a word image, number of horizontal transition points, within R2 and R3, between object and background pixels or vice versa along each row of B is calculated. Let, transition count along a horizontal row i is $\mathcal{T}_i$, where $i \in [R2, R3]$. The first feature value f1 (i.e., MTP) is estimated by

$$f1 = \frac{1}{N}\sum_{i=R2}^{i=R3} \mathcal{T}_i, \text{where } N = |\{i: \mathcal{T}_i \neq 0\}|, \forall i \in [R2, R3] \tag{7.1}$$

The second feature value (i.e., $f2$) represents the number of components (either character or character like shape) in the middle zone of a word image. To count the number of character or character like shapes, the word image is segmented vertically along *Matra* regions. For this two *fuzzy membership functions viz*. trapezoidal [226] and bell-shaped [46] are customized here for detection *Matra* region and segmentation points on *Matra* region respectively. The detail descriptions of these processes is described in Chapter 5 (sections 5.2.3 and 5.2.4).

Finally, counts of dominant CCs in upper and lower zones (i.e., f3 and 4) are estimated by eqs. 7.2 and 7.3 respectively.

$$f2 = |\{C: \theta(C) = 1\}| \tag{7.2}$$

$$f3 = |\{C: \emptyset(C) = 1\}| \tag{7.3}$$

In eqs. 7.2 and 7.3, C represents CC while $\theta(.)$ and $\emptyset(.)$ are functions that represent the belongingness of a CC in upper / lower zone respectively which are defined by

$$\theta(C) = \begin{cases} 0, if\ min\{i: f(i,j)\epsilon C\} \leq \dfrac{R1 + R2}{2} and\ max\{i: f(i,j)\epsilon C\} = R2 - 1 \\ 1, Otherwise \end{cases} \tag{7.4}$$

$$\emptyset(C) = \begin{cases} 0, if\ max\{i: f(i,j)\epsilon C\} \geq \dfrac{R3 + R4}{2} and\ min\{i: f(i,j)\epsilon C\} = R3 + 1 \\ 1, Otherwise \end{cases} \tag{7.5}$$

7.2.1.1.2 Formulation of Decision Rule

To filter out the words those are not relevant with respect to a given search word, here, a decision rule has been designed. For this, first, decision boundaries (lower and upper bounds) for each of the extracted feature values (i.e., f1, f2, f3 and f4) are estimated. These decision boundaries are set by considering mean, ($\mu_{fi}, i = 1$), mode ($m_{fi}, i = 2, 3, 4$) and standard deviation ($\sigma_{fi}, i = 1, 2, 3, 4$) of feature values extracted from reference word image samples, collected in offline mode, for each of the keyword. Let, $\mathcal{L}_{fi}$ and $\mathcal{U}_{fi}$ are the lower and upper bounds of feature value fi $(i = 1, 2, 3, 4)$ respectively which are defined by

$$\mathcal{L}_{fi} = \mu_{fi} - \sigma_{fi} \text{ , where } i = 1 \tag{7.6}$$
$$\mathcal{L}_{fi} = m_{fi} - \lfloor \sigma_{fi} \rfloor \text{ , where } i = 2, 3, 4 \tag{7.7}$$
$$\mathcal{U}_{fi} = \mu_{fi} + \lceil \sigma_{fi} \rceil \text{ , where } i = 1 \tag{7.8}$$
$$\mathcal{U}_{fi} = m_{fi} + \lceil \sigma_{fi} \rceil \text{ , where } i = 2, 3, 4 \tag{7.9}$$

Finally, a word is pre-classified as a probable candidate for the given search word by the decision rule as depicted in Fig. 7.4.

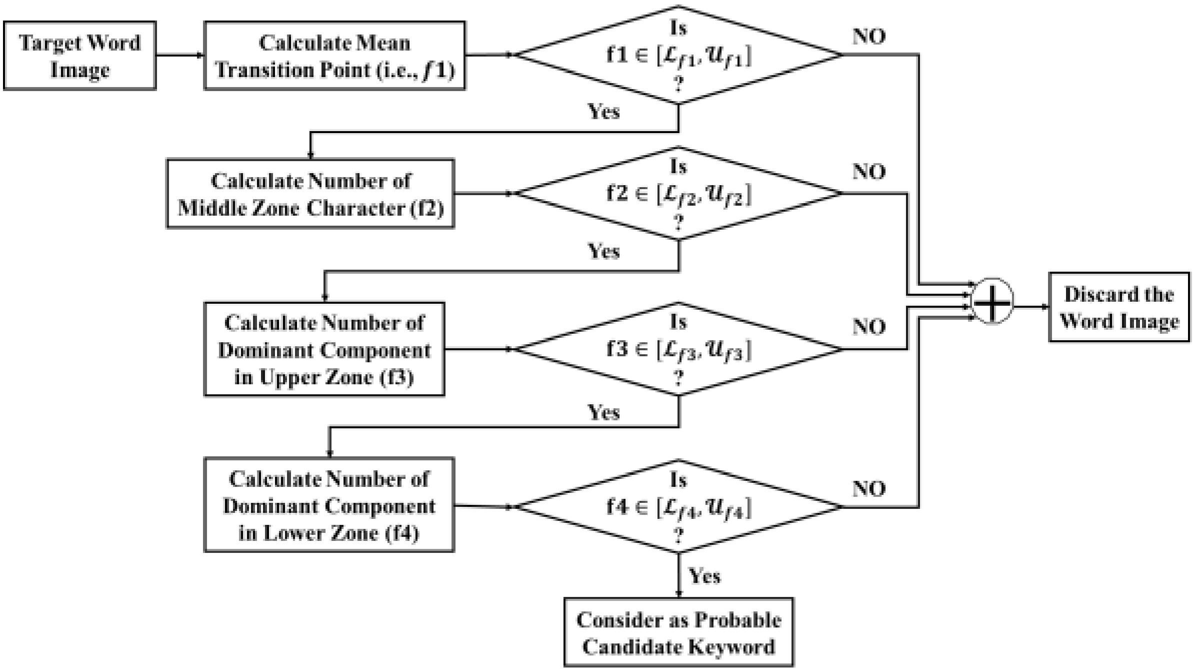

Target Word Image
Calculate Mean Transition Point (i.e., $f1$)
Is f1 ∈ [$\mathcal{L}_{f1}$, $\mathcal{U}_{f1}$] ?
NO
Yes
Calculate Number of Middle Zone Character (f2)
Is f2 ∈ [$\mathcal{L}_{f2}$, $\mathcal{U}_{f2}$] ?
NO
Yes
Calculate Number of Dominant Component in Upper Zone (f3)
Is f3 ∈ [$\mathcal{L}_{f3}$, $\mathcal{U}_{f3}$] ?
NO
Yes
Calculate Number of Dominant Component in Lower Zone (f4)
Is f4 ∈ [$\mathcal{L}_{f4}$, $\mathcal{U}_{f4}$] ?
NO
Yes
Discard the Word Image
Consider as Probable Candidate Keyword

Fig. 7.4 Diagrammatic representation of the selection process of a word image as probable candidate keyword using rule-based mechanism

7.2.1.2 Confirming a Probable Candidate Keyword as a Keyword

Confirmation of a probable candidate keyword as keyword has been done using holistic word recognition algorithm. For recognition purpose, another feature vector ($\mathfrak{F}_2$) comprising elliptical [84], topological [86] and gradient based [189] features, is extracted from reference word image samples of each the keyword. After this, a feature subset ($\mathfrak{F}_2{}'$) is selected from entire $\mathfrak{F}_2$ using memetic algorithm (MA) based hierarchical feature selection (HFS) model which has been introduced during this thesis work for recognizing handwritten *Bangla* words holistically. The detail description of this module is presented in section 6.2 of Chapter 6. Next, this selected feature vector (i.e., $\mathfrak{F}_2{}'$), are fed to an MLP-based classifier to carry out recognition and thereby a learned module is prepared. Finally, only the feature subset $\mathfrak{F}_2{}'$ is extracted from each of the candidate keywords and which is then passed through the learned module. The system provides recognition confidence (say, r) for a probable candidate keyword. Then, this candidate word is confirmed as search word if $r \geq \mathcal{T}$. $\mathcal{T}$ is a predefined threshold value which is here chosen as 0.9. The entire process of confirmation is illustrated in Fig. 7.5.

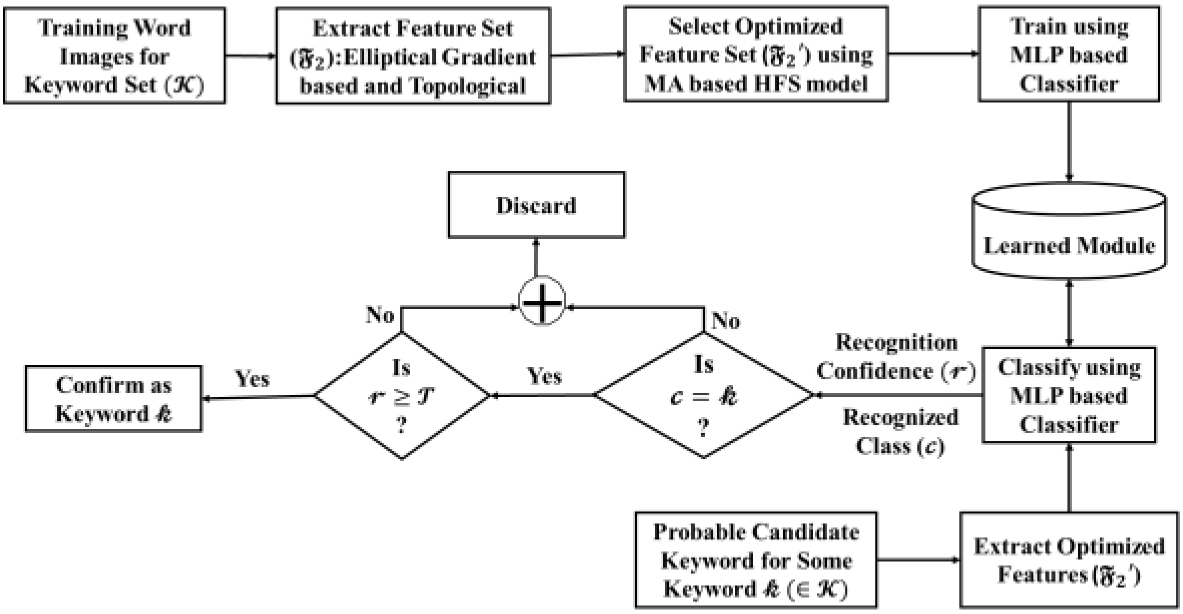

Training Word Images for Keyword Set ($\mathcal{K}$)
Extract Feature Set ($\mathfrak{F}_2$): Elliptical Gradient based and Topological
Select Optimized Feature Set ($\mathfrak{F}_2'$) using MA based HFS model
Train using MLP based Classifier
Learned Module
Discard
No
No
Is $r \geq \mathcal{T}$?
Yes
Is $c = k$?
Recognition Confidence (r)
Recognized Class (c)
Classify using MLP based Classifier
Confirm as Keyword k
Yes
Probable Candidate Keyword for Some Keyword k ($\in \mathcal{K}$)
Extract Optimized Features ($\mathfrak{F}_2'$)

Fig. 7.5 Schematic diagram for detecting instance(s) of keyword from its probable candidate keyword set

7.2.2 Document Classification

It has already been mentioned that in this chapter a handwritten document image classification technique has been introduced. For this purpose, 5 personalized categories of document page images, which are related to festival, geography, sport, technology and history, are collected. Also a set of document page images are considered here which does not fall into either of these categories. This set of images is considered as miscellaneous category. Hence, in total 6 categories of document pages are prepared during this thesis work. The detail description about the document page images are already described in Chapter 2. In addition to this a set of keywords is chosen for performing personalized classification of collected handwritten document page images. Next, these personalized keywords are searched within the document pages using the process described in the previous section. Based on the presence of the pre-set keywords, the documents are classified. For classification a feature vector, which consists of two types of features *viz.*, logarithmic term frequency and grouped term frequency, and MLP based classifier are used. These two features are explained in the following subsections.

7.2.2.1 Logarithmic Term Frequency

Term frequency and inverse document frequency (tf-idf) [227] is a well-known measure for machine encoded textual content based document clustering/ classification. While searching a keyword from such documents, the exact count of it is available since this system performs string matching i.e., ASCII value based comparison. There is no chance of generating extra/less count for the searched words. The same is not true while considering textual content in handwritten documents. Inspired by the consideration of logarithmic value for calculation of "idf", here logarithmic term frequency (ltf) is chosen in place of term frequency. The ltf is estimated as

$$ltf = \log_e(1 + tf) \qquad 7.10$$

Here tf carries the actual meaning of term frequency which is defined as

$$tf = \frac{number\ of\ occurances\ of\ a\ term\ (i.e., keyword)\ in\ a\ document}{total\ number\ of\ terms\ in\ the\ document} \qquad 7.11$$

7.2.2.2 Grouped Term Frequency

When a term (here, keyword) is searched from a handwritten document, there is no chance of retrieving all the instances. But, if a set of keywords is considered instead of single keyword then chances for retrieving these keywords as a whole will be higher. Therefore, here, another concept called group term frequency (gtf) is introduced. The value of gtf is calculated as

$$gtf = \frac{1}{n}\sum_{i=1}^{n} ltf_i \qquad 7.12$$

Here, n is the number of keywords considered by a user for some personalized document category and $ltf_i's$ ($i = 1, 2, \dots, n$) are the calculated ltf for the i^{th} keyword.

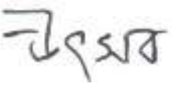
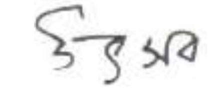
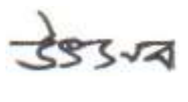
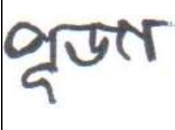
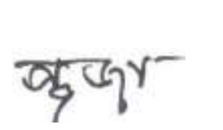
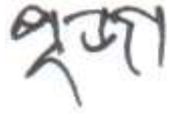
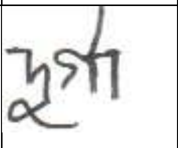
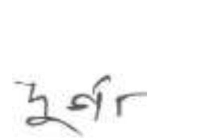
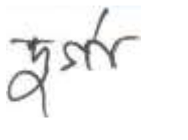
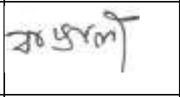
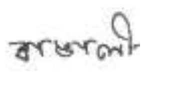
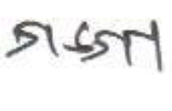
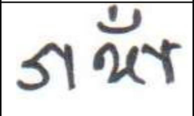
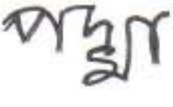
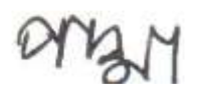
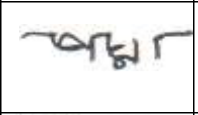
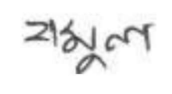
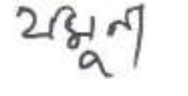
7.3 Experimental Results

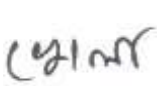
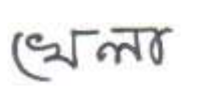
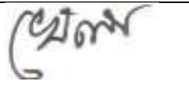
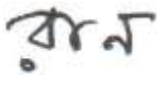
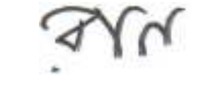
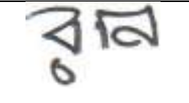
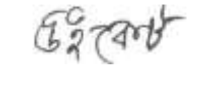
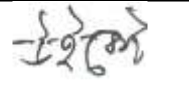
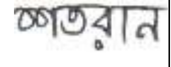
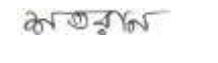
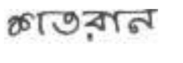
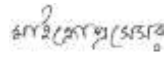
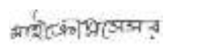
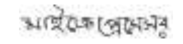
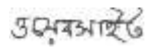
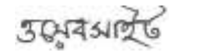
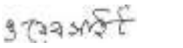
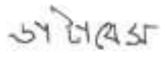
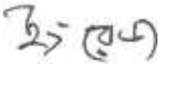
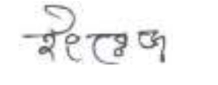
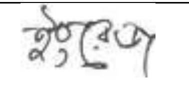
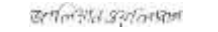
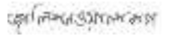
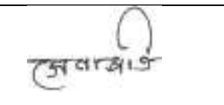
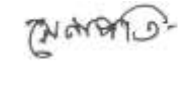
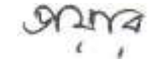
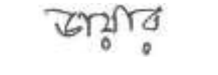
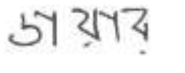
In the present work, a personalized handwritten document image classification technique is proposed. For this, first keywords present in a handwritten document image is counted and then a feature vector comprising logarithmic term frequency and grouped term frequency is designed. Finally MLP based classifier is used to classify the said categories. For the purpose of word searching, a two-stage approach, discussed earlier, has been introduced. In the following subsection, several issues related to experimental outcomes have been described.

7.3.1 Selection of Keyword Set

It has already been mentioned that 300 document page images, collected during this thesis work, are used for document classification purpose. It is noteworthy to mention that the database contains handwritten document page images that belong to 6 different categories *viz.,* festival, geographical, sport, technological, historical and none of these (i.e., miscellaneous). However, for evaluation of the proposed keyword searching technique, 60 document images (10 samples per each category) are considered randomly. In addition, a set of keywords ($\mathcal{K}$) is selected from document images that are shown in Table 7.1. More such collected word samples are provided in Table A2 of appendix. Total number of distinct query keywords as selected is 20 (4 from each of the 5 personalized categories) i.e., $|\mathcal{K}| = 20$. It is to be noted that absence of keywords in a document page is tagged the page as miscellaneous document.

Table 7.1 Instances of query keywords (KW ID stands for keyword index)

KW ID	Word	Phonology	Sample Word Images			Keyword Frequency
			1	2	3	
01	উৎসব	Utsaba				31
02	পূজা	Pūjā				38
03	দুর্গা	Durgā				29
04	বাঙালী	Bāṅālī				9
05	নদী	Nadī				65
06	গঙ্গা	Gaṅgā				20
07	পদ্মা	Padmā				11
08	যমুনা	Yamunā				9

09	খেলা	Khēlā				58
10	রান	Rāna				28
11	উইকেট	U'ikēṭa				10
12	শতরান	Śatarāna				11
13	মাইক্রোপ্র সেসর	Mā'ikrōpra sēsara				9
14	ওয়েবসা ইট	Ōẏēbasā' iṭa				10
15	ডাটাবেস	Ḍāṭābēsa				37
16	গবেষণা গার	Gabēṣaṇā gāra				9
17	ইংরেজ	Inrēja				31
18	জালিয়ানও য়ালাবাগ	Jāliẏāna' ōẏālābāga				9
19	সেনাপতি	Sēnāpati				8
20	ডায়ার	Ḍāẏāra				19

7.3.2 Preparation of Reference Keyword Sample Images

As mentioned earlier that some reference images for every keyword are collected in offline mode to formulate decision rules for performing selection of probable candidate keywords. These reference word images have also been used to obtain trained module which has been used here for confirming a probable candidate keyword either as query keyword or to reject it. For this reason 150 word images per keyword (see Table 7.1) are collected during this work. Please note that none of the authors who has written the mentioned 300 document pages has provided reference keyword images. However, the process followed to collect these samples is same as collecting city name samples which has been described in previous chapter. Therefore, a database containing $20 * 150 = 3000$ word images is prepared. It is worth mentioning that the preprocessing steps, applied on each of the collected word images, are described previously. To be specific, binarized images are prepared using the binarization technique which is described in Chapter 2 while preprocessing techniques related to feature extraction are described in Chapter 5.

7.3.3 Parameters for Detecting Preselected Candidate Words

The feature values $f1, f2$, $f3$ and $f4$ (see eqs. (7-9)) are first extracted from each of the preprocessed reference keyword images and then the bounds of the same (see eqs. (7.1-7.3)) are also calculated. These bound values for each of the query word images are shown in Table 7.2. Using these values and selection process, described in Fig. 7.4, the irrelevant target keywords with respect to a provided keyword are removed successfully. It has been observed that around 80% of the target words occur in a document page are filtered out during this stage.

Table 7.2 Search word-wise lower and upper bound values of $f1, f2$, $f3$ and $f4$ (KW ID indicates query keyword index)

KW ID	$\mathcal{L}_{fi}$				U_{fi}			
	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 1$	$i = 2$	$i = 3$	$i = 4$
01	11.91	3	0	0	14.14	5	2	1
02	9.93	2	0	0	12.98	6	0	2
03	7.42	2	0	0	9.91	6	2	2
04	15.87	4	0	0	19.09	8	2	1
05	7.51	2	0	0	9.71	4	2	1
06	10.01	2	0	0	13.10	6	1	1
07	10.37	1	0	0	13.48	5	1	2
08	11.91	3	0	0	15.15	5	1	2
09	11.72	2	0	0	14.63	6	1	2
10	8.14	2	0	0	10.01	4	1	2
11	11.72	2	0	0	16.21	8	2	2
12	16.79	3	0	0	20.94	7	1	2
13	29.41	8	0	1	35.10	14	2	3
14	21.18	7	1	0	25.03	11	3	2
15	17.47	5	0	0	20.75	9	2	2
16	20.10	5	0	0	25.33	9	1	2
17	12.74	4	0	1	16.25	8	2	3
18	38.31	11	0	2	47.47	21	2	4
19	18.95	5	0	0	22.71	9	2	2
20	13.58	5	0	1	16.53	7	1	3

7.3.4 Performance of Holistic Word Recognition

It has already been mentioned that a holistic word recognition technique has been adopted here for confirming a preselected candidate keyword as the query keyword. To prepare the required trained module for the keywords, the feature vector $\mathfrak{F}_2$, which consists of elliptical, topological and gradient based features, is extracted from all the reference word images related to keywords. Relying on the experimental outcome as obtained in section 6.4.1, length of $\mathfrak{F}_2$ is considered as 564 (i.e., 65 (elliptical feature) + 203 (topological feature) + 296 (gradient based feature)). Next, all these features are fed to the MA based HFS technique for finding optimal feature set. A similar set of experiments, described in section 6.4.3 of previous chapter, is carried out to obtain optimal feature subset (i.e., $\mathfrak{F}_2'$). The length of $\mathfrak{F}_2'$ as obtained here is 397. Finally, a 5-fold cross validation technique is performed to obtain the learned module. The recognition results have been recorded in Table 7.3. The learned module that provides best recognition accuracy is used for confirming a preselected word as search word (refer to section 7.2.1.2).

Table 7.3 Word recognition performances using 5-fold cross validation on reference word image set

Fold #	Number of training samples	Number of test samples	Recognition accuracy (in %)
1	$20 \times 120 = 2400$	$20 \times 30 = 600$	**97.67**
2			95.17
3			96.83
4			95.50
5			96.83

7.3.5 Performance of Keyword Searching

It has already been mentioned that in the present work a two-stage keyword searching mechanism has been designed. Performance of this keyword searching regulates the success of document classification. Hence, here performance of the keyword searching method is assessed on 60 document page image, a randomly selected subset of 300 document page images that have been prepared during this thesis work (see Chapter 2). The assessment of the present keyword searching mechanism is carried out in terms of recall, precision, true negative rate (TNR), predictive positive condition rate (PPCR), F-measure score and accuracy [75]. For this the 4 measures *viz.*, true positive (TP), false positive (FP), true negative (TN) and false negative (FN) are counted.

Let, $\mathcal{K}$ and N_{TW} are the set of keywords considered here for searching from the document page images and number of total target words in the entire database respectively. In addition, meaning of the counts N_{AKW}^{t}, N_{RKW}^{t} and N_{RTW}^{t} represent the number of t^{th} keyword k^{t} ($k^{t} \in \mathcal{K}$) actually present in the entire database, correctly retrieved from the entire database and actually retrieved target words from the entire database respectively. Now, TP^t, TN^t, FN^t and FP^t of t^{th} keyword k^t $(k^t \in \mathcal{K})$ are calculated as

$$TP^t = N^t_{RKW} \tag{7.13}$$

$$FP^t = N^t_{RTW} - N^t_{RKW} \tag{7.14}$$

$$FN^t = N^t_{AKW} - N^t_{RKW} \tag{7.15}$$

$$TN^t = N_{TW} + N^t_{RKW} - N^t_{AKW} - N^t_{RTW} \tag{7.16}$$

Finally the required statistic *viz.*, recall, precision, TNR, PPCR, accuracy and F-measure for t^{th} $(t \in \mathcal{K})$ keyword are calculated as below

$$recall^t = \frac{TP^t}{TP^t + FN^t} = \frac{N^t_{RKW}}{N^t_{AKW}} \tag{7.17}$$

$$precision^t = \frac{TP^t}{TP^t + FP^t} = \frac{N^t_{RKW}}{N^t_{RTW}} \tag{7.18}$$

$$TNR^t = \frac{TN^t}{TN^t + FP^t} = \frac{N_{TW} - N^t_{RTW}}{N_{TW} - N^t_{AKW}} \tag{7.19}$$

$$PPCR^t = \frac{TP^t + FP^t}{TP^t + TN^t + FP^t + FN^t} = \frac{N^t_{RTW}}{N_{TW}} \tag{7.20}$$

$$Acuracy^t = \frac{TP^t + TN^t}{TP^t + TN^t + FP^t + FN^t} \tag{7.21}$$

$$= \frac{N_{TW} + 2 \times N^t_{RKW} - N^t_{AKW} - N^t_{RTW}}{N_{TW}}$$

$$F - measure^t = 2 \times \frac{recall^t \times precision^t}{recall^t + precision^t} \tag{7.22}$$

Keyword-wise recall, precision and F-measure scores have been provided in Table 7.4. From this table it is clear that present work has achieved admissible retrieval accuracy. This table also reveals that the best recall is achieved while searching the keyword having ID 11 and overall satisfactory recall value (lowest recall value=0.7778) indicates better retrieving capability of the present work for all of the keyword. Whereas the lower precision values (≤ 0.25) are observed for 8 keywords viz. '**বাঙালী**', '**পদ্মা**', '**রান**', '**উইকেট**', '**শতরান**', '**গবেষণাগার**', '**সেনাপতি**', '**ডায়ার**'. The reason behind such low precision rate is due to character count (in number) and/or shape similarity of the target words with the keywords. On the other hand, while top 3 precision value (≥ 0.5) obtained for the keywords '**জালিয়ানওয়ালাবাগ**', '**মাইক্রোপ্রসেসর**' and '**উৎসব**' which are with more number of characters than usual or containing less occurring character. In addition higher values of TNR and accuracy and lower values of PPCR indicate its higher capability of searching corresponding keywords.

Table 7.4 Keyword-wise recall, precision TNR, PPCR, Accuracy and F-measure values. Here, KW ID stands for query keyword index

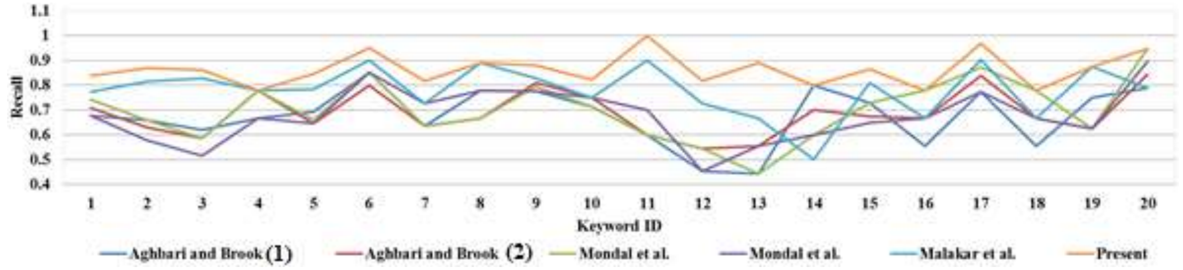
KW ID	Recall	Precision	TNR	PPCR	Accuracy	F-measure
01	0.8387	0.5098	0.9968	0.0065	0.9962	0.6341
02	0.8684	0.3367	0.9917	0.0125	0.9911	0.4853
03	0.8621	0.3049	0.9927	0.0104	0.9922	0.4505
04	0.7778	0.1842	0.9961	0.0048	0.9958	0.2979
05	0.8462	0.4104	0.9899	0.0171	0.9887	0.5527
06	0.9500	0.2676	0.9934	0.0090	0.9933	0.4176
07	0.8182	0.1875	0.9950	0.0061	0.9948	0.3051
08	0.8889	0.2963	0.9976	0.0034	0.9975	0.4445
09	0.8793	0.3018	0.9849	0.0215	0.9841	0.4494
10	0.8214	0.1345	0.9811	0.0218	0.9805	0.2312
11	**1.0000**	0.1724	0.9939	0.0074	0.9939	0.2941
12	0.8182	0.1607	0.9940	0.0071	0.9938	0.2686
13	0.8889	0.6667	0.9995	0.0015	0.9994	0.7619
14	0.8000	0.3810	0.9983	0.0027	0.9981	0.5162
15	0.8649	0.4156	0.9942	0.0098	0.9936	0.5614
16	0.7778	0.2121	0.9967	0.0042	0.9964	0.3333
17	0.9677	0.4000	0.9943	0.0095	0.9941	0.5660
18	0.7778	**0.7778**	**0.9997**	**0.0011**	**0.9995**	**0.7778**
19	0.8750	0.2000	0.9964	0.0045	0.9963	0.3256
20	0.9474	0.1818	0.9897	0.0126	0.9896	0.3051
Average	0.8634	0.3251	0.9938	0.0087	0.9934	0.4489

7.3.6 Comparison with *State-of-the-art* Word Searching Methods

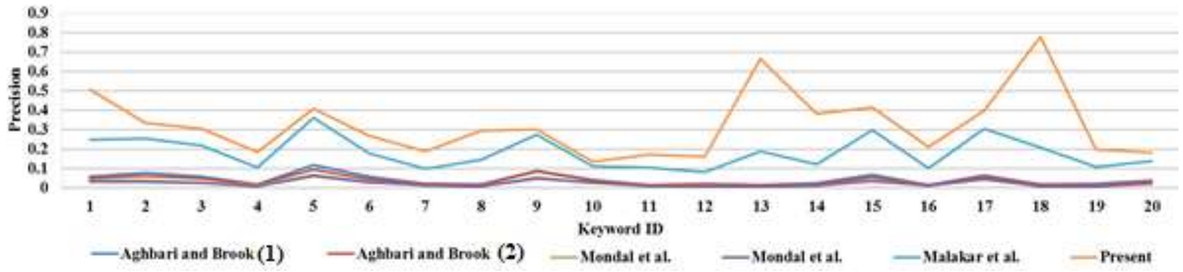
The performance of the present technique is compared with some s*tate-of-the-art* methods [68-69, 71, 218]. Aghbari and Brook [69] have used recognition based word retrieval method. In their work they have extracted several statistical features. Here, for performance comparison both of these two feature extraction methods are implemented. On the other hand, the authors in the works [68, 218] have used recognition free word searching method. In [68] they have applied FSM technique while in [218], several variations of time series matching techniques are compared. In both the cases to match two word images they have used a set of 8 feature values extracted from each column of a word image. For comparison with [218], conventional DTW method is applied here. In addition, the work described in [71] which has been introduced during this thesis work, has extracted topological features and modified HOG features for performing word searching from handwritten English document page images. The average performances of the methods compared here are recorded in Table 7.5. The search word-wise performance evaluation of different *state-of-the-art* methods along with the present one is shown in Fig. 7.6 (a-f). These results show that the recall values are very close to each other but with respect to other measures present method outperforms the others with significant margin.

Table 7.5 Average recall, precision and F-measure values for different methods

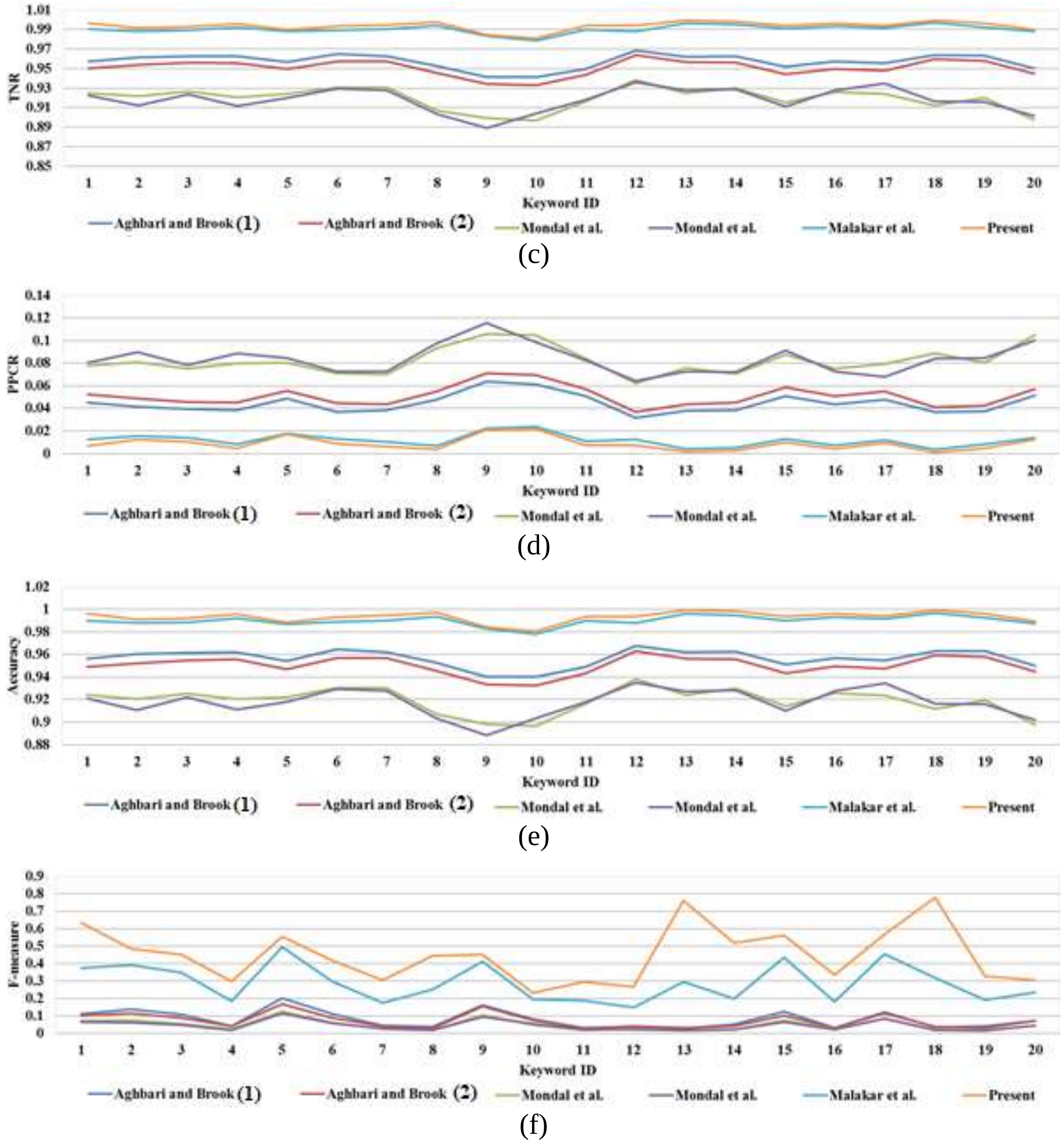
Methods	Feature extracted from	Average					
		Recall	**Precision**	**TNR**	**PPCR**	**Accuracy**	**F-measure**
Aghbari and Brook (1) [69]	Entire word	0.6761	0.0429	0.9575	0.0444	0.9567	0.0791
Aghbari and Brook (2) [69]	Connected parts of word image	0.6865	0.0375	0.9509	0.0509	0.9502	0.0698
Mondal et al. [68]	Each column of word image	0.7002	0.0244	0.9194	0.0825	0.9188	0.0465
Mondal et al. [218]	Each column of word image	0.6777	0.0229	0.9181	0.0836	0.9175	0.0436
Malakar et al. [71]	topological feature and modified histogram of oriented gradient	0.7790	0.1825	0.9905	0.0118	0.9900	0.2875
Present	Entire as well as from sub-images	**0.8634**	**0.3251**	**0.9938**	**0.0087**	**0.9934**	**0.4489**



(a)



(b)



(c)

(d)

(e)

(f)

Fig. 7.6 Keyword-wise (a) recall, (b) precision, (c) TNR, (d) PPCR, (e) Accuracy and (f) F-measure values for different methods. Horizontal axis represents indices of keywords (refer to Table 7.1) and vertical axis represents corresponding score

7.3.7 Document Classification

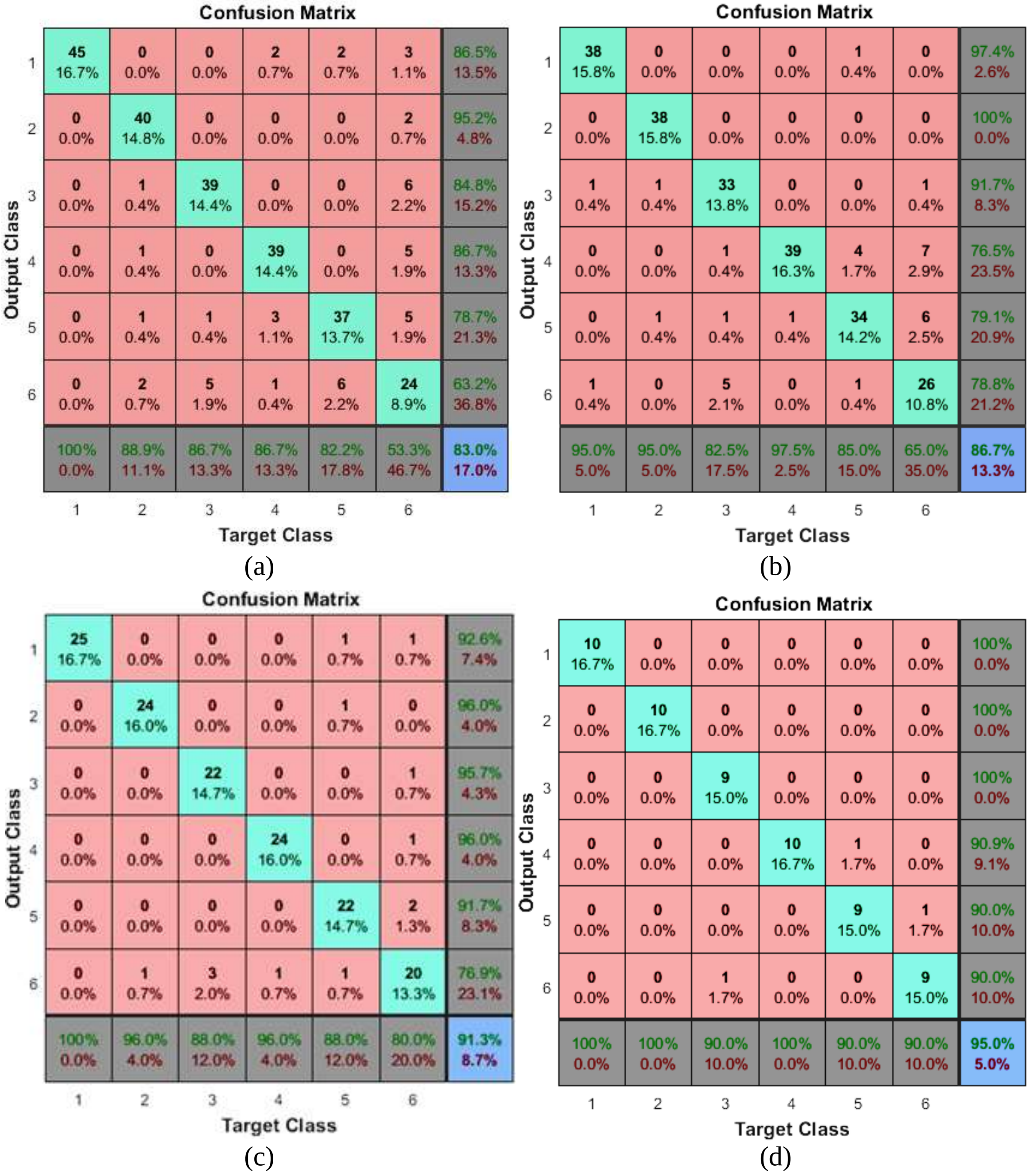
In the present work, logarithmic term frequency for each of the keywords and grouped term frequency of each of the personalized categories are considered here as feature vector. Total number of keyword used for classification purpose is 20 (4 for each of the personalized categories) and number of personalized category is 5. It is noteworthy to mention that no keyword is considered for miscellaneous category of document images. Total number of categories considered here is 6 which include 5 personalized categories and a miscellaneous category. 50 handwritten document page images for each of the said categories are collected during this thesis work (refer to Chapter 2 for further detail). Finally, the said features are extracted from each of the document images and fed to MLP based classifier. 5 set of experiments are considered here with varying number of train and test samples. The distribution of training and test data is provided in Table 7.6. It is to be noted that 20% of the training data used for validation. The classification accuracies of the said experiments have been recorded in Table 7.6. Also, the confusion matrices corresponding to the best recognition accuracy in each of the experiments are shown in Fig. 7.7(a-e). From these results, it can safely be commented that the present handwritten document image classification system performs satisfactorily. In addition, the confusion matrices indicate that in most of the cases document belonging to miscellaneous category misclassified as other classes while the document images for which the keywords are defined are well categorized.

Table 7.6 Document classification results for different experiments that are conducted using varying number of train and test samples

Experiment #	Number of			Recognition Accuracy (in %)			
	training sample per class	test sample per class	test	Best	Worst	Average	SD
01	5	45	10	82.96	74.81	79.74	2.77
02	10	40	5	86.67	80.83	84.33	2.12
03	25	25	2	91.33	91.33	91.33	0.00
04	40	10	5	95.00	91.67	94.00	1.33
05	45	5	10	100.0	93.33	96.34	1.80

7.4 Discussion

Classification/indexing of handwritten document images is an utmost necessity for their easy retrieval, searching etc. from the large archive. But, due to poor performance of existing OCR system for handwritten documents, it is till date almost impossible to devise a system with reasonable accuracy although machine encoded textual content classification is a matured research domain. Therefore, in this chapter, a keyword based handwritten document classification technique has been designed. In this method, at first, page-to-word extraction technique has been applied to extract all the words, punctuation marks etc. that present in the document page. Next, a two-stage keyword searching method is used to ensure the presence of user given keywords in a document and thereby their counts also.



Confusion Matrix

Output Class / Target Class

	1	2	3	4	5	6	
1	45 16.7%	0 0.0%	0 0.0%	2 0.7%	2 0.7%	3 1.1%	86.5% 13.5%
2	0 0.0%	40 14.8%	0 0.0%	0 0.0%	0 0.0%	2 0.7%	95.2% 4.8%
3	0 0.0%	1 0.4%	39 14.4%	0 0.0%	0 0.0%	6 2.2%	84.8% 15.2%
4	0 0.0%	1 0.4%	0 0.0%	39 14.4%	0 0.0%	5 1.9%	86.7% 13.3%
5	0 0.0%	1 0.4%	1 0.4%	3 1.1%	37 13.7%	5 1.9%	78.7% 21.3%
6	0 0.0%	2 0.7%	5 1.9%	1 0.4%	6 2.2%	24 8.9%	63.2% 36.8%
	100% 0.0%	88.9% 11.1%	86.7% 13.3%	86.7% 13.3%	82.2% 17.8%	53.3% 46.7%	83.0% 17.0%

(a)

Confusion Matrix

Output Class / Target Class

	1	2	3	4	5	6	
1	38 15.8%	0 0.0%	0 0.0%	0 0.0%	1 0.4%	0 0.0%	97.4% 2.6%
2	0 0.0%	38 15.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
3	1 0.4%	1 0.4%	33 13.8%	0 0.0%	0 0.0%	1 0.4%	91.7% 8.3%
4	0 0.0%	0 0.0%	1 0.4%	39 16.3%	4 1.7%	7 2.9%	76.5% 23.5%
5	0 0.0%	1 0.4%	1 0.4%	1 0.4%	34 14.2%	6 2.5%	79.1% 20.9%
6	1 0.4%	0 0.0%	5 2.1%	0 0.0%	1 0.4%	26 10.8%	78.8% 21.2%
	95.0% 5.0%	95.0% 5.0%	82.5% 17.5%	97.5% 2.5%	85.0% 15.0%	65.0% 35.0%	86.7% 13.3%

(b)

Confusion Matrix

Output Class / Target Class

	1	2	3	4	5	6	
1	25 16.7%	0 0.0%	0 0.0%	0 0.0%	1 0.7%	1 0.7%	92.6% 7.4%
2	0 0.0%	24 16.0%	0 0.0%	0 0.0%	1 0.7%	0 0.0%	96.0% 4.0%
3	0 0.0%	0 0.0%	22 14.7%	0 0.0%	0 0.0%	1 0.7%	95.7% 4.3%
4	0 0.0%	0 0.0%	0 0.0%	24 16.0%	0 0.0%	1 0.7%	96.0% 4.0%
5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	22 14.7%	2 1.3%	91.7% 8.3%
6	0 0.0%	1 0.7%	3 2.0%	1 0.7%	1 0.7%	20 13.3%	76.9% 23.1%
	100% 0.0%	96.0% 4.0%	88.0% 12.0%	96.0% 4.0%	88.0% 12.0%	80.0% 20.0%	91.3% 8.7%

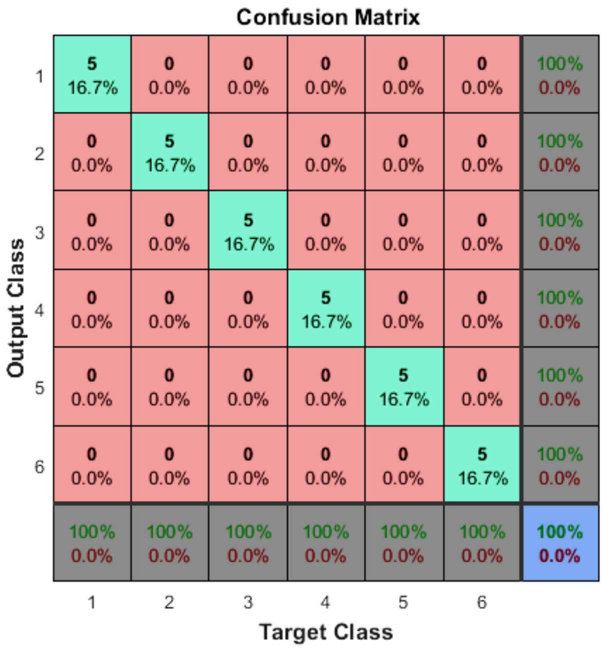
(c)

Confusion Matrix

Output Class / Target Class

	1	2	3	4	5	6	
1	10 16.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
2	0 0.0%	10 16.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
3	0 0.0%	0 0.0%	9 15.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
4	0 0.0%	0 0.0%	0 0.0%	10 16.7%	1 1.7%	0 0.0%	90.9% 9.1%
5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	9 15.0%	1 1.7%	90.0% 10.0%
6	0 0.0%	0 0.0%	1 1.7%	0 0.0%	0 0.0%	9 15.0%	90.0% 10.0%
	100% 0.0%	100% 0.0%	90.0% 10.0%	100% 0.0%	90.0% 10.0%	90.0% 10.0%	95.0% 5.0%

(d)



Confusion Matrix

Output Class \ Target Class	1	2	3	4	5	6	
1	5 16.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
2	0 0.0%	5 16.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
3	0 0.0%	0 0.0%	5 16.7%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
4	0 0.0%	0 0.0%	0 0.0%	5 16.7%	0 0.0%	0 0.0%	100% 0.0%
5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	5 16.7%	0 0.0%	100% 0.0%
6	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	5 16.7%	100% 0.0%
	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%	100% 0.0%

(e)

Fig 7.7 Confusion matrices of the best classification results for all the five mentioned experiments. Number of test document pages in (a), (b), (c), (d) and (e) are 45, 40, 25, 10 and 5 respectively

In first stage of keyword searching, irrelevant target words with respect to a query keyword are removed. During confirmation or in the second stage, all the probable candidate keywords from a document image are passed through a pre-trained module which is prepared using a holistic word recognition method. In this word recognition system, first 3 feature vectors *viz.*, elliptical, topological and gradient based are extracted from word images and then those are passed through MA based HFS model to generate an optimal feature subset which defines the keywords under consideration in a better way. Based on recognition confidence, a preselected candidate keyword is tagged as query keyword.

Finally, a document page is represented as a feature vector, which depends on number of keywords used for classification and number of personalized categories considered for documents under consideration, in the feature space. The feature values as used here are ltf and gtf. MLP based classifier is used for document classification purpose. The experimental results indicate that the proposed two-stage keyword searching method and designed handwritten document classification approach yield satisfactory performance.

Chapter 8

CONCLUSION AND FUTURE SCOPE

Before invention of modern printing technologies like printing press, typewriter, etc., people mainly used pen-and-paper for various types of documentation. A large section of the educated people still prefers to keep notes of their activities, maintain daily accounts of income/expenditure etc. in written form. Even today most of the doctors use pen-and-paper for listing their observations about their patients and prescribing medicines. All these said issues infer the presence of huge amount of handwriting documents. It is quite obvious that the number of handwritten documents will go on increasing in future due to the widespread use of handwritten documents in various educational institutions, offices, banks, hospitals and so on.

Generally, handwritten documents are prepared hurriedly. Even these are not managed properly after their creation due to lack of easily available standardized management protocol. Hence, searching some important document(s), required in a rush, from this gigantic amount of document pages becomes almost impossible in manual mode. Moreover, due to lack of proper concern, some documents get misplaced or lost. Also, these documents get degraded over the time due to lack of proper management. In addition to these, the amount of spaces these documents occupy is increasing with time. Here arises the requirement for managing these ever increasing documents automatically which is very challenging though it is a pressing need. The requirement may be fulfilled if the documents are kept in digital form or in image form with adequate indexing, which in turn helps in searching and managing the documents in much easier way.

One of the possible ways of indexing any document is through the understanding of the context of the textual content present in it. Unfortunately, analysis of textual content present in a document image is not an easy task to accomplish. It is mainly because the size of a collection which is often substantial and the current handwritten Optical Character Recognition (OCR) system works poorly when such a large lexicon size is taken into account [13-14].

Considering these facts, during the course of this thesis work, an alternative attempt is made for classifying handwritten *Bangla* documents without converting them into machine encoded form. In this context, the work presented in this thesis is mainly targeted at handwritten documents written in *Bangla* script. The reason for chosen *Bangla* script is mainly due to its richness and character shape complexity over Roman script. To be more specific, presence of compound characters, modified shapes, diacritic, *Matra* etc. makes this script rich and complex than others.

The major objective of this work is to devise an automatic system that can search a user provided keyword in handwritten *Bangla* documents and also can classify the documents into personalized categories on the basis of a set of keywords supplied. The searching of keywords is performed using a two-stage approach [71]. It is to be noted that the task has been completed by keeping the handwritten documents in image form since conversion of such documents into machine editable form is almost impossible if the status of the current handwritten *Bangla* OCR system is considered. Therefore, the present work has covered all the associated stages needed to perform such analysis. In addition, preparation of suitable databases for performing related experiments and designing of automated evaluation tools for measuring performance of page segmentation algorithms, i.e., text line (TL), word and character extraction have also been undertaken in this work.

First and foremost, a database consisting of handwritten documents belonging to any of the six personalized categories *namely* festive, geographical, sports, technological, historical and miscellaneous, is prepared. To prepare the handwritten document page image database handwriting samples are collected using the forms similar to those used in IAM database [92]. During collection of data sufficient variations in terms of writers' age, sex and educational qualification have been maintained which in turn help in adding adequate variations in writing style of the individuals. Next, the collected datasheets are scanned on a flat-bed scanner and the handwritten textual parts from the form like datasheets have been extracted programmatically. To accomplish the job, the scanned form like datasheets are first enhanced using middle of modal class (MMC) filtering technique and the these enhanced datasheets are binarized using ratio based binarization. Also, morphological close operator and a connected component (CC) based technique [21] have been used to remove unwanted clusters of object/background pixels. This preprocessing technique not only works well for the data collected for this work but also performs satisfactorily for degraded images. In the preprocessing pipeline, lastly, a Hough transform (HT) based [77] technique has been employed to correct page level skewness, if any.

It is to be noted that to search a keyword from handwritten textual content, a target word, which is either a word or a punctuation mark, is compared with the supplied keyword. Hence, segmentation of a handwritten document page into words becomes an important stage of the current work. However, extraction of words from a handwritten document page image can be performed by either of two ways: (i) first TLs are extracted from documents and then words are extracted from each of the TLs, (ii) words are directly extracted from the document page image, which is termed here as page-to-word word extraction technique. After thorough experimentation, second approach is followed here as this approach outperforms the first one.

To extract words from handwritten document images using the former approach, initially a TL extraction algorithm is devised. In this technique, a document image is partitioned into a number of vertical fragments (VFs) first and then Spiral Run Length Smearing algorithm (SRLSA) [79] is employed in each of the VFs. SRLSA merges spirally close CCs to form virtually CCs (VCCs). Next, upper and lower contours of these VCCs are analyzed to obtain line segments (LSs) in each VFs. After this, any small sized LS, which is decided by its height, is joined hypothetically with its vertically nearest LS. Finally, an inter-fragment joining protocol, which connects two vertically close LSs belonging to two adjacent VFs, is designed to identify final set of TLs.

To evaluate the present TL extraction algorithm automatically, an assessment strategy, which compares a segmented document page with its ground truth (GT) image, consisting of labeled ideal TLs, using one-to-one pixel mapping and returns evaluation results in terms of true positive (TP), false positive (FP), false negative (FN), recall, precision and F-measure scores, is introduced during the course of this thesis work. For experimentation, 300 document page images, which are prepared here, are used. The average values of the said scores, as found on the entire database, are 0.9322, 0.0178, 0.0500, 0.9808, 0.9487 and0.9633 respectively.

For extracting words from handwritten document images, two methods have been followed. The methods are (i) word extraction from isolated TLs and (ii) word extraction directly from document page. To extract words from TLs, a SRLSA based technique, described in the work [79], is used. It is to be noted that the input TLs for the system are extracted from handwritten documents using the said TL extraction technique, which means the errors occurring during the TL extraction phase will be carried forward into this technique.

In case of the second method, a CC based page-to-word extraction technique is used, which bypasses the errors occurring during TL extraction. The page-to-word extraction technique can also minimize the overall execution time. The technique identifies all the CCs present in a document image using 8 way-CC labelling (8-CCL) algorithm [33] first and then classifies the CCs into two groups, *namely*, small sized and large sized, based on their height and width. Next, the small sized CCs have been joined virtually with the closest CC in its 8-neighbours to form virtually CCs (VCCs). After this stage, the document image contains isolated components (either VCCs or large sized CCs), which either belong to same or different words. Hence, these isolated components have been considered for joining to get the final set of words.

Both the word extraction methods have been applied on 300 document page images. The previously said evaluation protocol is also used for automatic evaluation of the present algorithm. The average TP, FP, FN, recall, precision and F-measure as found using the first method are 0.8936, 0.0726, 0.0338, 0.9636, 0.9249, and 0.9439 respectively. While these scores as found using the second method are 0.9336, 0.0294, 0.0370, 0.9696, 0.9620 and 0.9651 respectively. From these results it is clear that the page-to-word extraction technique performs better than the other method on the said document page image database.

During the course of this thesis work, a modification of existing character extraction technique for handwritten *Bangla* word [150] is introduced. This algorithm is used in the first stage of the present two-stage keyword searching technique. In this technique, for estimating starting (i.e., R2) and ending (i.e., R4) rows of middle zone, a mask based technique has been introduced while detection of starting row of upper zone (i.e., R1) and ending row of lower zone (i.e., R4) are straight forward. The width of the mask is set as width of the word image while height varies with variation of vertical object and background pixel runs. Next, a *fuzzy trapezoidal membership function* and a *fuzzy bell-shaped membership function* are used for detecting *Matra* pixels and segmentation points respectively. The method also deals with under and over segmentations and loss of object pixels. Along with these, a lower zone separation technique is also introduced. For assessment of the said technique, 5000 handwritten isolated *Bangla* word images along with GT images consisting of ideally segmented word images and the above mentioned automatic evaluation protocol are used. The obtained average TP, FP, FN, recall, precision and F-measure scores are 0.8623, 0.0528, 0.0847, 0.9418, 0.9105 and 0.9212 respectively.

In the second stage of the present word searching technique, a holistic word recognition technique is used. The holistic word recognition uses three types of features, namely, *elliptical*, *gradient based* and *topological.* A hierarchical feature selection (HFS) technique [88] is applied on the extracted features for removing the redundant features. In the proposed HFS technique, all the elementary features of each type are separately optimized using Memetic Algorithm (MA) based feature selection first and then all these optimized features are concatenated to get it optimized further using the same MA based feature selection technique. For evaluating the proposed holistic word recognition system, a database containing 150 handwritten samples of each of 120 most popular city names of West Bengal, a state of India, is prepared during this work. It is worth mentioning that the database is already made available for the research community through the work [73]. The city name images are recognized using an MLP based classifier with 5-fold cross validations. The best, worst, average and standard deviation of classification accuracies as obtained here are 88.31%, 82.33%, 84.27% and 2.14% respectively (without using HFS) and 90.94%, 85.81%, 87.11% and 1.99% respectively (using HFS).

Finally, a keyword searching method is introduced here which satisfactorily searches a set of predefined keywords from a pool of handwritten document images. The search outcomes are used here for classifying the handwritten documents written in *Bangla* script. It has already been said that the underlying keyword searching technique looks for a keyword in a document using a two-stage approach involving word level matching. Hence, document images are segmented into word images using a page-to-word extraction method since it performs better than its counterpart i.e., the approach where word extraction follows TL extraction. This word extraction technique identifies words and punctuation marks as word images. This scenario turns creates a large search space while searching a keyword from document page images.

Hence, in the first stage of the keyword searching, punctuation marks and irrelevant words with respect to a given query keyword are removed from the large search space using a set of decision rules. Four zonal features viz., mean of the number of transition points in the middle zone and the numbers of middle zone characters, ascendants and descendants are used for this purpose. The remaining target words in the document page images are called as probable candidate keywords. In second stage, proposed HFS based holistic word recognition protocol has been used to search the keyword. To perform recognition based keyword searching 150 sample word images for each of the keywords (a set of 20 keywords is considered here) are collected during this work. The experimentation is conducted on 60 document pages, taking 10 from each of the abovementioned categories, of the current page level handwritten document page database. The obtained average recall, precision, true negative rate (TNR), predictive positive condition rate (PPCR), accuracy and F-measure values are 0.8634, 0.3251, 0.9938, 0.0087, 0.9934 and 0.4489 respectively.

Also, based on presence of supplied set of keywords for each of the predefined categories of documents, the pool of the document page images are classified into respective categories. For classification, a MLP based classifier and two feature attributes, called *logarithmic term frequency* and *grouped term frequency*, are used. It is to be noted that such choice of features helps in diminishing the effect of erroneous retrieval of keywords. The classification technique is evaluated on a database containing 300 handwritten *Bangla* document page images (50 images per each of the 6 predefined categories) with varying number of training and test samples. The best, worst and average classification accuracies having 5 training samples and 45 test samples per category are 82.96%, 74.81% and 79.74% respectively. However, while considering 45 training samples and 5 test samples per each category the said accuracies become 100%, 93.33% and 96.34% respectively.

Although the techniques described in this thesis work provide satisfactory results considering present *state-of-the-art* works yet, there are still some room for improvement. Possible extensions of this thesis work which can be done in future are as follows:

1. The errors occurred during extraction of TL, word and character are mostly due to presence of touching components, which causes merging of multiple TLs, words and characters into single TL, word and character respectively. Therefore, a suitable method is required for segmenting such touching components.
2. The work undertaken in this thesis has considered only handwritten *Bangla* document pages or isolated word images. In future, the present technique may be employed on other *Matra* based scripts like Devanagari, Gurumukhi and Syloti to establish the generalness of the present work.
3. It is to be noted that databases used for TL, word and character extraction include corresponding GT images. GT image preparation techniques, used here, are universal in nature (as these use one to one pixel mapping), so the same could be used for creating GT image databases for the handwritten texts written in other scripts.
4. The evaluation protocol, introduced during this thesis work, could be easily applied to create an evaluation tool for other document image processing problems include

suppression of graphics components from handwritten documents, extraction of handwritten parts from form documents, identification of regions containing keywords in segmentation free approaches.

5. Recognition of word samples has been carried out using only MLP. Hence, other recognizer like support vector machine, naïve Bayes, deep learning could be used in future for improving the recognition. In addition, probabilistic models like different Bayes models, hidden Markov model can be used.
6. Last but not the least, the document page image database can be used in future for experimenting newly designed algorithms for writer identification and writer's demographics classification. Also, the databases (document pages and isolated words) could be useful for assessment of most of stages of an OCR system.

In summary, the work presented in this thesis is a result of consistent efforts to cover almost all the major stages to build an automated system that can classify the textual contents in handwritten *Bangla* documents. Such attempt resulting into an integrated approach towards classifying handwritten *Bangla* document is unique. Also, the improved intermediate stages might aid in constructing an OCR system for handwritten *Bangla* document images in future. In addition to these, the results obtained through application of intermediate stages of the present work on freely written *Bangla* text can help in identifying the areas where more effort is needed in future. Finally, the databases developed under this work would be very beneficial to the handwritten document image processing research fraternity for testing, evaluating and comparing performances of algorithms designed for TL, word and character extraction, holistic word recognition, keyword spotting and document classification purposes.

REFERENCES

[1] Y. Hannad, I. Siddiqi, and M. E. Y. El Kettani, “Writer identification using texture descriptors of handwritten fragments,” *Expert Syst. Appl.*, vol. 47, pp. 14–22, 2016.

[2] M. N. Abdi and M. Khemakhem, “A model-based approach to offline text-independent Arabic writer identification and verification,” *Pattern Recognit.*, vol. 48, no. 5, pp. 1890–1903, 2015.

[3] S. Y. Ooi, A. B. J. Teoh, Y. H. Pang, and B. Y. Hiew, “Image-based handwritten signature verification using hybrid methods of discrete Radon transform, principal component analysis and probabilistic neural network,” *Appl. Soft Comput. J.*, vol. 40, pp. 274–282, 2016.

[4] Y. Guerbai, Y. Chibani, and B. Hadjadji, “The effective use of the one-class SVM classifier for handwritten signature verification based on writer-independent parameters,” *Pattern Recognit.*, vol. 48, no. 1, pp. 103–113, 2015.

[5] Y. Liu, Z. Yang, and L. Yang, “Online Signature Verification Based on DCT and Sparse Representation,” *IEEE Trans. Cybern.*, vol. 45, no. 11, pp. 2498–2511, 2015.

[6] I. Siddiqi, C. Djeddi, A. Raza, and L. Souici-meslati, “Automatic analysis of handwriting for gender classification,” *Pattern Anal. Appl.*, vol. 18, no. 4, pp. 887–899, 2015.

[7] C. Djeddi, S. Al-Maadeed, A. Gattal, I. Siddiqi, A. Ennaji, and H. El Abed, “ICFHR2016 competition on multi-script writer demographics classification using QUWI database,” in *Proceedings of International Conference on Frontiers in Handwriting Recognition (ICFHR)*, 2017, pp. 602–606.

[8] V. A. G. da Silva, M. Talhavini, I. C. F. Peixoto, J. J. Zacca, A. O. Maldaner, and J. W. B. Braga, “Non-destructive identification of different types and brands of blue pen inks in cursive handwriting by visible spectroscopy and PLS-DA for forensic analysis,” *Microchem. J.*, vol. 116, pp. 235–243, 2014.

[9] A. Agius, M. Morelato, S. Moret, S. Chadwick, K. Jones, R. Epple, J. Brown, and C. Roux, “Using handwriting to infer a writer’s country of origin for forensic intelligence purposes,” *Forensic Sci. Int.*, vol. 282, pp. 144–156, 2018.

[10] A. Sen, H. Shah, J. Lemos, and S. Bhattacharjee, “An Algorithm to Extract Handwriting Feature for Personality Analysis,” in *Proceedings of International Conference on Wireless*

Communication, 2018, pp. 323–329.

[11] R. Kacker and H. B. Maringanti, “Personality Analysis Through Handwriting,” *GSTF J. Comput.*, vol. 2, no. 1, 2014.

[12] A. Bandyopadhyay, B. Mukherjee, and A. Hazra, “Perception Based Decision Support System for Handwriting Behaviour Analysis,” *Procedia Comput. Sci.*, vol. 84, pp. 177–185, 2016.

[13] C. Tensmeyer, D. Saunders, and T. Martinez, “Convolutional Neural Networks for Font Classification,” *IEIE Trans. Smart Process. Comput.*, vol. 6, no. 1, pp. 53–59, 2017.

[14] D. Chakraborty, P. P. Roy, R. Saini, J. M. Alvarez, and U. Pal, “Frame selection for OCR from video stream of book flipping,” *Multimed. Tools Appl.*, vol. 77, no. 1, pp. 985–1008, 2018.

[15] S. Basu, N. Das, R. Sarkar, M. Kundu, M. Nasipuri, and D. K. Basu, “A hierarchical approach to recognition of handwritten Bangla characters,” *Pattern Recognit.*, vol. 42, no. 7, pp. 1467–1484, 2009.

[16] “Abstract of Speakers’ Strength of Languages and Mother Tongues - 2011.” [Online]. Available: http://www.censusindia.gov.in/2011Census/Language-2011/Statement-1.pdf. [Accessed: 17-Dec-2018].

[17] “List of languages by number of native speakers.” [Online]. Available: https://en.wikipedia.org/wiki/List_of_languages_by_number_of_native_speakers. [Accessed: 17-Dec-2018].

[18] “Bengali language.” [Online]. Available: http://www.newworldencyclopedia.org/entry/Bengali_language. [Accessed: 17-Dec-2018].

[19] “Bengali.” [Online]. Available: https://www.ethnologue.com/language/ben. [Accessed: 17-Dec-2018].

[20] “Babbel Magazine.” [Online]. Available: https://www.babbel.com/en/magazine/the-10-most-spoken-languages-in-the-world/. [Accessed: 17-Dec-2018].

[21] S. Malakar, D. Mohanta, R. Sarkar, N. Das, M. Nasipuri, and D. K. Basu, “A New Global Thresholding Approach for Document Image Binarization,” *Int. J. Inf. Process.*, vol. 6, no. 2, pp. 48–59, 2011.

[22] “Bengali alphabet.” [Online]. Available: https://en.wikipedia.org/wiki/Bengali_alphabet. [Accessed: 17-Dec-2018].

[23] A. Chatterjee, S. Malakar, R. Sarkar, and M. Nasipuri, “Handwritten Digit Recognition using DAISY Descriptor : A Study,” in *Proceedings of Fifth International Conference on Emerging*

Applications of Information Technology (EAIT), 2018, pp. 1–4

[23] N. Das, K. Acharya, R. Sarkar, S. Basu, M. Kundu, and M. Nasipuri, "A benchmark image database of isolated Bangla handwritten compound characters," *Int. J. Doc. Anal. Recognit.*, vol. 17, no. 4, pp. 413–431, 2014.

[24] R. Sarkar, N. Das, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, "CMATERdb1: A database of unconstrained handwritten Bangla and Bangla-English mixed script document image," *Int. J. Doc. Anal. Recognit.*, vol. 15, no. 1, pp. 71–83, 2012.

[25] B. Gatos, N. Stamatopoulos, and G. Louloudis, "ICDAR2009 handwriting segmentation contest," *Int. J. Doc. Anal. Recognit.*, vol. 14, no. 1, pp. 25–33, 2011.

[26] S. Bhowmik, S. Polley, M. G. Roushan, S. Malakar, R. Sarkar, and M. Nasipuri, "A holistic word recognition technique for handwritten Bangla words," *Int. J. Appl. Pattern Recognit.*, vol. 2, no. 2, pp. 142–159, 2015.

[27] T. A. Nodes and N. C. Gallagher, "Median Filters: Some Modifications and Their Properties," *IEEE Trans. Acoust.*, vol. 30, no. 5, pp. 739–746, 1982.

[28] A. M. Wink and J. B. T. M. Roerdink, "Denoising functional MR images : a comparison of wavelet-based denoising and Gaussian smoothing," *IEEE Trans. Med. Imaging*, vol. 23, no. 3, pp. 374–387, 2004.

[29] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Trans. Syst. Man. Cybern.*, vol. 9, no. 1, pp. 62–66, 1979.

[30] M. A. Hasan, *Introduction to digital image processing*, vol. 34. Prentice-Hall Englewood Cliffs, 2018.

[31] B. Gatos, I. Pratikakis, and S. J. Perantonis, "Adaptive degraded document image binarization," *Pattern Recognit.*, vol. 39, no. 3, pp. 317–327, 2006.

[32] S. Malakar, D. Mohanta, R. Sarkar, N. Das, M. Nasipuri, and D. K. Basu, "Binarization of the noisy document images: A new approach," in *Proceedings of International Conference on Information Processing*, 2011, vol. 157, pp. 511–520.

[33] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. 2018.

[34] L. Seong-whan and T. Y. Yan, *Advances in oriental document analysis and recognition techniques*, vol. 33. World Scientific, 1999.

[35] P. Mukhopadhyay and B. B. Chaudhuri, "A survey of Hough Transform," *Pattern Recognit.*, vol. 48, no. 3, pp. 993–1010, 2015.

[36] N. Liolios, N. Fakotakis, and G. Kokkinakis, "On the generalization of the form identification and skew detection problem," *Pattern Recognit.*, vol. 35, no. 1, pp. 253–264, 2002.

[37] J. Dong, P. Dominique, A. Krzyyzak, and C. Y. Suen, "Cursive word skew/slant corrections based on Radon transform," in *Proceedings of Eighth International Conference on Document Analysis and Recognition,* 2005, pp. 478–483.

[38] S. Malakar, R. K. Das, R. Sarkar, S. Basu, and M. Nasipuri, "Handwritten and printed word identification using gray-scale feature vector and decision tree classifier," in *Proceedings of International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA),* 2013, Procedia Technol., pp. 831–839.

[39] F. Le Bourgeois, E. Trinh, B. Allier, V. Eglin, and H. Emptoz, "Document images analysis solutions for digital libraries," in *Proceedings of First International Workshop on Document Image Analysis for Libraries,* 2004, pp. 2–24.

[40] S. Bhowmik, R. Sarkar, M. Nasipuri, and D. Doermann, "Text and non-text separation in offline document images: a survey," *Int. J. Doc. Anal. Recognit.*, vol. 21, no. 1–2, pp. 1–20, 2018.

[41] A. M. Hesham, M. A. A. Rashwan, H. M. Al-Barhamtoshy, S. M. Abdou, A. A. Badr, and I. Farag, "Arabic document layout analysis," *Pattern Anal. Appl.*, vol. 20, no. 4, pp. 1275–1287, 2017.

[42] T. Plötz and G. A. Fink, "Markov models for offline handwriting recognition: a survey," *Int. J. Doc. Anal. Recognit.*, vol. 12, no. 4, p. 269, 2009.

[43] A. Kaltenmeier, T. Caesar, J. M. Gloger, and E. Mandler, "Sophisticated topology of hidden Markov models for cursive script recognition," in *Proceedings of the Second International Conference on Document Analysis and Recognition,* 1993, pp. 139–142.

[44] A. P. Giotis, G. Sfikas, B. Gatos, and C. Nikou, "A survey of document image word spotting techniques," *Pattern Recognit.*, vol. 68, pp. 310–332, 2017.

[45] P. Wang, V. Eglin, C. Garcia, C. Largeron, J. Llados, and A. Fornes, "A coarse-to-fine word spotting approach for historical handwritten documents based on graph embedding and graph edit distance," in *Proceedings of International Conference on Pattern Recognition*, 2014, pp. 3074–3079.

[46] R. Sarkar, S. Malakar, N. Das, S. Basu, M. Kundu, and M. Nasipuri, "Word extraction and character segmentation from text lines of unconstrained handwritten Bangla document images," *J. Intell. Syst.*, vol. 20, no. 3, pp. 227–260, 2011.

[47] A. Al-dmour and F. Fraij, "Segmenting Arabic Handwritten Documents into Text lines and

Words," *Int. J. Adv. Comput. Technol.*, vol. 6, no. 3, pp. 109–119, 2014.

[48] A. Khandelwal, P. Choudhury, R. Sarkar, S. Basu, M. Nasipuri, and N. Das, "Text Line Segmentation for Unconstrained Handwritten Document Images Using Neighborhood Connected Component Analysis," in *Proceedings of International conference* on *Pattern Recognition and Machine Intelligence*, 2009, vol. 5909, pp. 369–374.

[49] G. Louloudis, B. Gatos, I. Pratikakis, and C. Halatsis, "Text line and word segmentation of handwritten documents," *Pattern Recognit.*, vol. 42, no. 12, pp. 3169–3183, 2009.

[50] R. Sarkar, N. Das, S. Basu, M. Kundu, and M. Nasipuri, "Extraction of text lines from handwritten documents using piecewise water flow technique," *J. Intell. Syst.*, vol. 23, no. 3, pp. 245–260, 2014.

[51] R. Sarkar, S. Basu, N. Das, A. F. Mollah, M. Kundu, and M. Nasipuri, "Line Extraction from Unconstraint Handwritten Document Pages using Piece-wise Water-flow Technique.," in *Proceedings of 5th Indian International Conference on Artificial Intelligence*, 2009, pp. 1861–1872.

[52] N. Stamatopoulos, G. Louloudis, and B. Gatos, "Handwriting Segmentation," *Doc. Anal. Text Recognit. Benchmarking State-of-the-art Syst.*, vol. 82, p. 29, 2018.

[53] P. K. Singh, R. Sarkar, M. Nasipuri, and D. Doermann, "Word-level script identification for handwritten Indic scripts," in *Proceedings of 13th International Conference on Document Analysis and Recognition (ICDAR)*, 2015, pp. 1106–1110.

[54] S. K. M. Obaidullah, K. C. Santosh, C. Halder, N. Das, and K. Roy, "Automatic Indic script identification from handwritten documents: page, block, line and word-level approach," *Int. J. Mach. Learn. Cybern.*, pp. 1–20, 2017.

[55] P. K. Singh, S. P. Chowdhury, S. Sinha, S. Eum, and R. Sarkar, "Page-to-word extraction from unconstrained handwritten document images," in *Proceedings of the First International Conference on Intelligent Computing and Communication*, 2017, pp. 517–525.

[56] N. Arefin, M. Hassan, M. Khaliluzzaman, and S. A. Chowdhury, "Bangla handwritten characters recognition by using distance-based segmentation and histogram oriented gradients," in *Pooceedings of Humanitarian Technology Conference (R10-HTC), 2017 IEEE Region 10*, 2017, pp. 678–681.

[57] S. Malakar, S. Halder, R. Sarkar, N. Das, S. Basu, and M. Nasipuri, "Text line extraction from handwritten document pages using spiral run length smearing algorithm," in *Proceedings of International Conference on Communications, Devices and Intelligent Systems, CODIS 2012,*

2012, pp. 616–619.

[58] P. P. Roy, A. K. Bhunia, A. Das, P. Dhar, and U. Pal, “Keyword spotting in doctor’s handwriting on medical prescriptions,” *Expert Syst. Appl.*, vol. 76, pp. 113–128, 2017.

[59] P. K. Singh, R. Sarkar, N. Das, and S. Basu, “Identification of Devnagari and Roman scripts from multi-script handwritten documents,” in *Proceedings of International Conference on Pattern Recognition and Machine Intelligence*, 2013, pp. 509–514.

[60] S. Basu, R. Sarkar, N. Das, M. Kundu, M. Nasipuri, and D. K. Basu, “A fuzzy technique for segmentation of handwritten Bangla word images,” in *Proceedings ofInternational Conference on Computing: Theory and Applications,* 2007, pp. 427–432.

[61] A. Roy, T. K. Bhowmik, S. K. Parui, and U. Roy, “A novel approach to skew detection and character segmentation for handwritten bangla words,” in *Proceedings of the Digital Imaging Computing: Techniques and Applications,* 2005, pp. 203–210.

[62] T. Bhowmik, A. Roy, and U. Roy, “Character Segmentation for Handwritten Bangla Words Using Artificial Neural Network,”in *Proceedings of International Workshop on Neural Networks Learning Document Analysis and Recognition, 2005*, pp. 28–32.

[63] S. Madhvanath, V. Govindaraju, and S. Member, “The Role of Holistic Paradigms in Handwritten Word Recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 149–164, 2001.

[64] J. A. Rodríguez-Serrano and F. Perronnin, “Handwritten word-spotting using hidden Markov models and universal vocabularies,” *Pattern Recognit.*, vol. 42, no. 9, pp. 2106–2116, 2009.

[65] K. Khurshid, C. Faure, and N. Vincent, “A novel approach for word spotting using merge-split edit distance,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2009, vol. 5702 LNCS, pp. 213–220.

[66] A. Sun, E.-P. Lim, and W.-K. Ng, “Personalized Classification for Keyword-Based Category Profiles,” in *Proceedings of 6th European Conference on Research and Advances Technology for Digital Technology,* 2002, vol. 2458, pp. 61–74.

[67] S. Srihari, C. Huang, and H. Srinivasan, “A search engine for handwritten documents,” in *Proceedings of Spie-Is&T Electronic Imaging,* 2005, vol. 5676, pp. 66--75.

[68] T. Mondal, N. Ragot, J. Y. Ramel, and U. Pal, “Flexible Sequence Matching technique: An effective learning-free approach for word spotting,” *Pattern Recognit.*, vol. 60, pp. 596–612, 2016.

[69] Z. Al Aghbari and S. Brook, “HAH manuscripts: A holistic paradigm for classifying and retrieving historical Arabic handwritten documents,” *Expert Syst. Appl.*, vol. 36, no. 8, pp. 10942–10951, 2009.

[70] M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós, “Efficient segmentation-free keyword spotting in historical document collections,” *Pattern Recognit.*, vol. 48, no. 2, pp. 545–555, 2015.

[71] S. Malakar, M. Ghosh, R. Sarkar, and M. Nasipuri, “Development of a Two-Stage Segmentation-Based Word Searching Method for Handwritten Document Images,” *J. Intell. Syst.*, 2018.

[72] L. Rothacker, M. Rusinol, and G. A. Fink, “Bag-of-features HMMs for segmentation-free word spotting in handwritten documents,” in *Proceedings of the International Conference on Document Analysis and Recognition,* 2013, pp. 1305–1309.

[73] S. Bhowmik, S. Malakar, R. Sarkar, S. Basu, M. Kundu, and M. Nasipuri, “Off-line Bangla handwritten word recognition: a holistic approach,” *Neural Comput. Appl.*, pp. 1–16, 2018.

[74] “GT Gen 1.1.rar: Ground Truth generating tool.” [Online]. Available: https://code.google.com/archive/p/cmaterdb/downloads?page=2. [Accessed: 17-Dec-2018].

[75] R. Berg, “Sensitivity and specificity.” [Online]. Available: https://en.wikipedia.org/wiki/Sensitivity_and_specificity. [Accessed: 17-Dec-2018].

[76] S. Malakar, D. Mohanta, R. Sarkar, and M. Nasipuri, “A Novel Noise-removal Technique for Document Images,” *Int. J. Comput. Commun. Technol.*, vol. 2, no. 2, pp. 120–124, 2010.

[77] S. Malakar, B. Seraogi, R. Sarkar, N. Das, S. Basu, and M. Nasipuri, “Two-stage skew correction of handwritten Bangla document images,” in *Proceedings of Third International Conference on Emerging Applications of Information Technology (EAIT),* 2012, pp. 303–306.

[78] R. Sarkar, S. Halder, S. Malakar, N. Das, S. Basu, and M. Nasipuri, “Text line extraction from handwritten document pages based on line contour estimation,” in *Proceedings of 3rd International Conference on Computing, Communication and Networking Technologies*. 2012, pp. 1–8.

[79] R. Sarkar, S. Moulik, N. Das, S. Basu, M. Nasipuri, and D. K. Basu, “Word extraction from unconstrained handwritten Bangla document images using Spiral Run Length Smearing Algorithm,” in *Proceedings of the 5th Indian International Conference on Artificial Intelligence,* 2011, pp. 32–46.

[80] D. Drivas and A. Amin, “Page segmentation and classification utilising a bottom-up approach,”

in *Proceedings of the Third International Conference on Document Analysis and Recognition,* 1995, pp. 610–614.

[81] P. K. Singh, S. Mahanta, S. Malakar, R. Sarkar, and M. Nasipuri, "Development of a page segmentation technique for Bangla documents printed in italic style," in *Proceedings of 2nd International Conference on Business and Information Management.(ICBIM),* 2014, pp. 120–125.

[82] R. Sarkar, N. Das, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, "A two-stage approach for segmentation of handwritten Bangla word images," in *Proceedings of International Conference on Frontiers in Handwriting Recognitions*, 2008, pp. 403–408.

[83] S. Malakar, P. Ghosh, R. Sarkar, N. Das, S. Basu, and M. Nasipuri, "An improved offline handwritten character segmentation algorithm for Bangla script," in *Proceedings of 2nd International Conference on Business and Information Management,* 2014, pp. 120–125.

[84] S. Bhowmik, S. Malakar, R. Sarkar, and M. Nasipuri, "Handwritten Bangla word recognition using elliptical features," in *Proceedings of 6th International Conference on Computational Intelligence and Communication Networks,* 2014, pp. 257–261.

[85] S. Bhowmik, M. G. Roushan, S. Polley, S. Malakar, R. Sarkar, and M. Nasipuri, "Handwritten Bangla Word Recognition using HOG Descriptor," in *Proceedings of Fourth International Conference on Emerging Applications of Information Technology*, 2014, pp. 193-197.

[86] S. Malakar, P. Sharma, P. K. Singh, M. Das, R. Sarkar, and M. Nasipuri, "A Holistic Approach for Handwritten Hindi Word Recognition," *Int. J. Comput. Vis. Image Process.*, vol. 7, no. 1, pp. 59–78, 2017.

[87] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *Proceedings of IEEE Computer Society Conference oComputer Vision and Pattern Recognition, 2005,* 2005, pp. 886–893.

[88] S. Malakar, M. Ghosh, S. Bhowmik, R. Sarkar, and M. Nasipuri, "A GA based Hierarchical Feature Selection Approach for Handwritten Word Recognition," *Neural Comput. Appl., Preprint*

[89] Z. Zhu, Y. Ong, and M. Dash, "Wrapper-Filter Feature Selection Algorithm Using A Memetic Framework," *IEEE Trans. Syst. Man, Cybern. Part B*, vol. 37, no. 1, pp. 1–19, 2007.

[90] "CMATERdb." [Online]. Available: http://www.cmaterju.org/cmaterdb.htm. [Accessed: 17-Dec-2018].

[91] "Indian Script Character Databases." [Online]. Available:

http://www.isical.ac.in/~ujjwal/download/database.html. [Accessed: 17-Dec-2018].

[92] “IAM Handwriting Database.” [Online]. Available: http://www.fki.inf.unibe.ch/databases/iam-handwriting-database. [Accessed: 17-Dec-2018].

[93] “Public Domain OCR.” [Online]. Available: https://www.nist.gov/services-resources/software/public-domain-ocr. [Accessed: 17-Dec-2018].

[94] R. Casey, D. Ferguson, K. Mohiuddin, and E. Walach, “Intelligent forms processing system,” *Mach. Vis. Appl.*, vol. 5, no. 3, pp. 143–155, 1992.

[95] “The EMNIST Dataset.” [Online]. Available: https://www.nist.gov/itl/iad/image-group/emnist-dataset. [Accessed: 17-Dec-2018].

[96] “THE MNIST DATABASE of handwritten digits.” [Online]. Available: http://yann.lecun.com/exdb/mnist/. [Accessed: 17-Dec-2018].

[97] “QUWI database.” [Online]. Available: http://handwriting.qu.edu.qa/dataset/. [Accessed: 17-Dec-2018].

[98] “ICDAR2017 Competitions.” [Online]. Available: http://u-pat.org/ICDAR2017/program_competitions.php. [Accessed: 17-Dec-2018].

[99] “ICFHR 2018 Competitions.” [Online]. Available: http://icfhr2018.org/competitions.html. [Accessed: 17-Dec-2018].

[100] “The Arabic Handwritten Digits Databases.” [Online]. Available: http://datacenter.aucegypt.edu/shazeem/. [Accessed: 17-Dec-2018].

[101] “Devnagari-database.” [Online]. Available: https://code.google.com/archive/p/devnagari-database/. [Accessed: 17-Dec-2018].

[102] K. Bache and M. Lichman, “UCI machine learning repository,” *UCI machine learning repository*. 2013.

[103] S. H. Cha and S. N. Srihari, “A priori algorithm for sub-category classification analysis of handwriting,” in *Proceedings of Sixth International Conference on Document Analysis and Recognition*, 2001, pp. 1022–1025.

[104] V. Bouletreau, “Synthetic parameters for handwriting classification,” in *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, 1997, pp. 102–106.

[105] “rgb2gray.” [Online]. Available: http://in.mathworks.com/help/matlab/ref/rgb2gray.html?requestedDomain=true. [Accessed: 17-Dec-2018].

[106] G. V. Borisenko and A. M. Denisov, “Nonlinear source in diffusion filtering methods for image processing,” *Comput. Math. Math. Phys.*, vol. 47, no. 10, pp. 1631–1635, 2007.

[107] F. Russo and G. Ramponi, “A fuzzy filter for images corrupted by impulse noise,” *IEEE Signal Process. Lett.*, vol. 3, no. 6, pp. 168–170, 1996.

[108] G. Qiu, “An improved recursive median filtering scheme for image processing,” *IEEE Trans. Image Process.*, vol. 5, no. 4, pp. 646–648, 1996.

[109] G. R. Arce and R. E. Foster, “Detail-Preserving Ranked-Order Based Filters For Image Processing,” *IEEE Trans. Acoust.*, vol. 37, no. 1, pp. 83–98, 1989.

[110] S. Arora, J. Acharya, A. Verma, and P. K. Panigrahi, “Multilevel thresholding for image segmentation through a fast statistical recursive algorithm,” *Pattern Recognit. Lett.*, vol. 29, no. 2, pp. 119–125, 2008.

[111] L. Zhang, X. Zhan, and X. R. Zhang, “Fingerprint Image Binarization Algorithm Based on Information Entropy,” *Comput. Syst. Appl.*, vol. 6, p. 37, 2010.

[112] A. Z. Arifin and A. Asano, “Image segmentation by histogram thresholding using hierarchical cluster analysis,” *Pattern Recognit. Lett.*, vol. 27, no. 13, pp. 1515–1521, 2006.

[113] A. Pérez and R. C. Gonzalez, “An Iterative Thresholding Algorithm for Image Segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9, no. 6, pp. 742–751, 1987.

[114] S. Ghosh and S. Bag, “An improvement on thinning to handle characters with noisy contour,” in Proceedings of *4th National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics*, 2013, pp. 1–4.

[115] L. Di Stefano and A. Bulgarelli, “A Simple and Efficient Connected Components Labeling Algorithm,” in *Proceedings of 10th International Conference on Image Analysis and Processing*, 1999, pp. 322–327.

[116] S. K. Mamatha Hosalli Ramappa, “Skew Detection, Correction and Segmentation of Handwritten Kannada Document,” *Int. J. Adv. Sci. Technol.*, vol. 48, pp. 71–88, 2012.

[117] B. Yu and A. K. Jain, “A robust and fast skew detection algorithm for generic documents,” *Pattern Recognit.*, vol. 29, no. 10, pp. 1599–1629, 1996.

[118] A. Papandreou and B. Gatos, “A novel skew detection technique based on vertical projections,” in *Proceedings of the International Conference on Document Analysis and Recognition,* 2011, pp. 384–388.

[119] A. K. Das and B. Chanda, “A fast algorithm for skew detection of document images using

morphology," *Int. J. Doc. Anal. Recognit.*, vol. 4, no. 2, pp. 109–114, 2001.

[120] A. Antonacopoulos, "Local skew angle estimation from background space in text regions," in *Proceedings of the International Conference on Document Analysis and Recognition*, 1997

[121] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 29, no. 6, pp. 1153–1160, 1981.

[122] "Inter quartile Range," *Flashcards*. [Online]. Available: https://en.wikipedia.org/wiki/Interquartile_range. [Accessed: 17-Dec-2018].

[123] L. Likforman-Sulem, A. Hanimyan, and C. Faure, "A Hough based algorithm for extracting text lines in handwritten documents," in *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 1995, pp. 774–777.

[124] Y. Pu and Z. Shi, "A natural learning algorithm based on hough transform for text lines extraction in handwritten documents," *Proc. 6th Int. Work. Front. Handwrit. Recognit.*, pp. 637–646, 1998.

[125] S. J. Ha, B. Jin, and N. I. Cho, "Fast text line extraction in document images," *Proceedings of 19th IEEE International Conference on Image Processing*, 2012, pp. 797–800.

[126] F. M. Wahl, K. Y. Wong, and R. G. Casey, "Block segmentation and text extraction in mixed text/image documents," *Comput. Graph. Image Process.*, vol. 20, no. 4, pp. 375–390, 1982.

[127] Z. Shi and V. Govindaraju, "Line separation for complex document images using fuzzy run length," in *Proceedings of First International Workshop on Document Image Analysis for Libraries*, 2004, pp. 306–212.

[128] P. P. Roy, U. Pal, and J. Lladós, "Morphology based handwritten line segmentation using foreground and background information," in *Proceedings of International Conference on Frontiers in Handwriting Recognition*, 2008, pp. 241–246.

[129] F. Yin and C.-L. Liu, "Handwritten Chinese text line segmentation by clustering with distance metric learning," *Pattern Recognit.*, vol. 42, no. 12, pp. 3146–3157, 2009.

[130] X. Du, W. Pan, and T. D. Bui, "Text line segmentation in handwritten documents using Mumford-Shah model," *Pattern Recognit.*, vol. 42, no. 12, pp. 3136–3145, 2009.

[131] Y. Li, Y. Zheng, D. Doermann, and S. Jaeger, "Script-independent text line segmentation in freestyle handwritten documents," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 8, pp. 1313–1329, 2008.

[132] S. Basu, C. Chaudhury, M. Kundu, M. Nasipuri, and D. K. Basu, "Text Line Extraction from

Multi Skewed Handwritten Documents," *Pattern Recognition,* vol. 40, no. 6, pp. 1825 – 1839, 2007.

[133] A. Alaei, U. Pal, and P. Nagabhushan, "A new scheme for unconstrained handwritten text line segmentation," *Pattern Recognit.*, vol. 44, no. 4, pp. 917–928, 2011.

[134] G. Louloudis, B. Gatos, I. Pratikakis, and C. Halatsis, "Text line detection in handwritten documents," *Pattern Recognit.,* vol. 41, no. 12, pp. 3758--3772, 2008.

[135] L. Wang, W. Fan, J. Sun, S. Naoi, and T. Hiroshi, "Text line extraction in document images," in *Proceedings of 13th International Conference on Document Analysis and Recognition,* 2015, pp. 191–195.

[136] "Digital differential analyzer (graphics algorithm)." [Online]. Available: https://en.wikipedia.org/wiki/Digital_differential_analyzer_(graphics_algorithm). [Accessed: 17-Dec-2018].

[137] M. Liwicki, M. Scherz, and H. Bunke, "Word extraction from on-line handwritten text lines," in *Proceedings of International Conference on Pattern Recognition*, 2006, pp. 929–933.

[138] "IAM On-Line Handwriting Database," 2005. [Online]. Available: http://www.fki.inf.unibe.ch/databases/iam-on-line-handwriting-database. [Accessed: 17-Dec-2018].

[139] V. Papavassiliou, T. Stafylakis, V. Katsouros, and G. Carayannis, "Handwritten document image segmentation into text lines and words," *Pattern Recognit.*, vol. 43, no. 1, pp. 369–377, 2010.

[140] "ICDAR 2007 Handwriting Segmentation Competition." [Online]. Available: http://www.icdar2007.org/competition.html. [Accessed: 17-Dec-2018].

[141] F. Kurniawan, A. R. Khan, and D. Mohamad, "Contour Vs Non-Contour based Word Segmentation from Handwritten Text Lines: an Experimental Analysis," *Int. J. Digit. Content Technol. its Appl.*, vol. 3, no. 2, pp. 127–131, 2009.

[142] N. Manmatha, R and Srimal, "Scale Space Technique for Word Segmentation in Handwritten Documents," in *Proceedings of the Second International Conference on Scale-Space Theories in Computer Vision*, 1999, pp. 22–33.

[143] K. Khurshid, C. Faure, and N. Vincent, "Fusion of word spotting and spatial information for figure caption retrieval in historical document images," in *Proceedings of 10th International Conference on Document Analysis and Recognition,* 2009, pp. 266–270.

[144] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," in *Proceedings of the*

Alvey Vision Conference, 1988, pp. 147-151.

[145] M. Ester and H. Kriegel, “Density-based spatial clustering of applications with noise (DBSCAN),” in Proceedings of *International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 1–5.

[146] “Euclidean Distance.” [Online]. Available: http://link.springer.com/10.1007/978-0-387-35973-1_373. [Accessed: 17-Dec-2018].

[147] B. B. Chaudhuri and U. Pal, “A complete printed Bangla OCR system,” *Pattern Recognit.*, vol. 31, no. 5, pp. 531–549, 1998.

[148] A. Bishnu and B. B. Chaudhuri, “Segmentation of Bangla handwritten text into characters by recursive contour following,” in *Proceedings of the International Conference on Document Analysis and Recognition,* 1999, pp. 402–405.

[149] U. Pal and S. Datta, “Segmentation of Bangla unconstrained handwritten text,” in *Proceedings of the International Conference on Document Analysis and Recognition,* 2003, pp. 1128–1132.

[150] R. Sarkar, S. Malakar, N. Das, S. Basu, and M. Nasipuri, “A Script Independent Technique for Extraction of Characters from Handwritten Word Images,” *Int. J. Comput. Appl.*, vol. 1, no. 23, pp. 83–88, 2010.

[151] R. Sarkar, B. Sen, N. Das, and S. Basu, “Handwritten Devanagari Script Segmentation : A Non-linear Fuzzy Approach,” *arXiv Prepr. arXiv1501.05472*, no. Cd, 2008.

[152] V. P. Dhaka and M. K. Sharma, “An efficient segmentation technique for Devanagari offline handwritten scripts using the Feedforward Neural Network,” *Neural Comput. Appl.*, vol. 26, no. 8, pp. 1881–1893, 2015.

[153] D. V. Sharma and G. S. Lehal, “An iterative algorithm for segmentation of isolated handwritten words in Gurmukhi script,” in *Proceedings of International Conference on Pattern Recognition,* 2006, pp. 1022–1025.

[154] M. Kumar, M. K. Jindal, and R. K. Sharma, “Segmentation of Isolated and Touching Characters in Offline Handwritten Gurmukhi Script Recognition,” *Int. J. Inf. Technol. Comput. Sci.*, vol. 6, no. 2, pp. 58–63, 2014.

[155] J. Sas and U. Markowska-Kaczmar, “Semi-supervised handwritten word segmentation using character samples similarity maximization and evolutionary algorithm,” in *Proceedings of 6th International Conference on Computer Information Systems and Industrial Management Applications,* 2007, pp. 316–321.

[156] M. K. Sharma and V. P. Dhaka, “Segmentation of english Offline handwritten cursive scripts

using a feedforward neural network," *Neural Comput. Appl.*, vol. 27, no. 5, pp. 1369–1379, 2016.

[157] T. Yamaguchi, S. Tsuruoka, T. Yoshikawa, T. Shinogi, E. Makimoto, H. Ogata, and M. Shridhar, "A segmentation system for touching handwritten Japanese characters," in *Proceedings Eighth International Workshop on Frontiers in Handwriting Recognition*, 2002, pp. 407–412.

[158] J. Tan, J. H. Lai, C. D. Wang, W. X. Wang, and X. X. Zuo, "A new handwritten character segmentation method based on nonlinear clustering," *Neurocomputing*, vol. 89, pp. 213–219, 2012.

[159] A. Lawgali, "A Survey on Arabic Character Recognition," *Int. J. Doc. Anal. Recognit.*, vol. 8, no. 2, pp. 401–426, 2015.

[160] N. Dave, "Segmentation methods for hand written character recognition," *Int. J. signal Process. image Process. pattern Recognit.*, vol. 8, no. 4, pp. 155–164, 2015.

[161] "Paschimbanga Bangla Akademi." [Online]. Available: https://en.wikipedia.org/wiki/Paschimbanga_Bangla_Akademi. [Accessed: 17-Dec-2018].

[162] U. Pal, K. Roy, and F. Kimura, "A lexicon-driven handwritten city-name recognition scheme for Indian postal automation," *IEICE Trans. Inf. Syst.*, vol. E92–D, no. 5, pp. 1146–1158, 2009.

[163] U. Pal, R. K. Roy, and F. Kimura, "Multi-lingual city name recognition for Indian postal automation," in *Proceedings of International Workshop on Frontiers in Handwriting Recognition,* 2012, pp. 169–173.

[164] M. Morita, R. Sabourn, A. El Yacoubi, F. Bortolozzi, and C. Y. Suen, "Handwritten Month Word Recognition on Brazilian Bank Checks," in *Proceedings of Sixth International Conference on Document Analysis and Recognition,* 2001, pp. 972–976.

[165] A. Namane, A. Guessoum, and P. Meyrueis, "New holistic handwritten word recognition and its application to French legal amount," in *Proceedings of International Conference on Pattern Recognition and Image Analysis*, 2005, pp. 654–663.

[166] S. Singh, T. Kariveda, J. Das Gupta, and K. Bhattacharya, "Handwritten words recognition for legal amounts of bank cheques in English script," in *Proceedings of 8th International Conference on Advances in Pattern Recognition*, 2015, pp. 1–5.

[167] T. Watanabe, Q. Luo, and N. Sugie, "Layout recognition of multi-kinds of table-form documents," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 4, pp. 432–445, 1995.

[168] C. A. Peanho, H. Stagni, and F. S. C. da Silva, "Semantic information extraction from images

of complex documents," *Appl. Intell.*, vol. 37, no. 4, pp. 543–557, 2012.

[169] R. A. Luciano Jr and L. Luciano, "System and method for processing a multiple prescription order." Google Patents, 18-Jul-2017.

[170] G. Kirn and V. Govindaraju, "A lexicon driven approach to handwritten word recognition for real-time applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 4, pp. 366–379, 1997.

[171] S. Edelman, T. Flash, and S. Ullman, "Reading cursive handwriting by alignment of letter prototypes," *Int. J. Comput. Vis.*, vol. 5, no. 3, pp. 303–331, 1990.

[172] A. Vinciarelli, S. Bengio, and H. Bunke, "Offline recognition of unconstrained handwritten texts using HMMs and statistical language models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 709–720, 2004.

[173] E. K. Warrington and T. Shallice, "Word-form dyslexia.," *Brain*, vol. 103, no. 1, pp. 99–112, 1980.

[174] E. Ishidera, S. M. Lucas, and A. C. Downton, "Top-down likelihood word image generation model for holistic word recognition," in *Proceedings of International Workshop on Document Analysis Systems*, 2002, pp. 82–94.

[175] P. P. Roy, A. K. Bhunia, A. Das, P. Dey, and U. Pal, "HMM-based Indic handwritten word recognition using zone segmentation," *Pattern Recognit.*, vol. 60, pp. 1057–1075, 2016.

[176] A. Broumandnia, J. Shanbehzadeh, and M. Rezakhah Varnoosfaderani, "Persian/arabic handwritten word recognition using M-band packet wavelet transform," *Image Vis. Comput.*, vol. 26, no. 6, pp. 829–842, 2008.

[177] B. El Qacimy, M. A. Kerroum, and A. Hammouch, "Word-based Arabic handwritten recognition using SVM classifier with a reject option," in *International Conference on Intelligent Systems Design and Applications*, 2016, pp. 64–68.

[178] Z. Tamen, H. Drias, and D. Boughaci, "An efficient multiple classifier system for Arabic handwritten words recognition," *Pattern Recognit. Lett.*, vol. 93, pp. 123–132, 2017.

[179] S. Madhvanath, "Holistic verification of handwritten phrases," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 12, pp. 1344–1356, 1999.

[180] J. Dasgupta, K. Bhattacharya, and B. Chanda, "A holistic approach for Off-line handwritten cursive word recognition using directional feature based on Arnold transform," *Pattern Recognit. Lett.*, vol. 79, pp. 73–79, 2016.

[181] R. Jayadevan, S. R. Kolhe, P. M. Patil, and U. Pal, “Database development and recognition of handwritten Devanagari legal amount words,” in *Proceedings of the International Conference on Document Analysis and Recognition,* 2011, pp. 304–308.

[182] J. Ruiz-Pinales, R. Jaime-Rivas, and M. J. Castro-Bleda, “Holistic cursive word recognition based on perceptual features,” *Pattern Recognit. Lett.*, vol. 28, no. 13, pp. 1600–1609, 2007.

[183] J. J. De Oliveira, C. O. A. De Freitas, J. M. De Carvalho, and R. Sabourin, “Handwritten word recognition using multi-view analysis,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics),* 2009, pp. 371–378.

[184] M. Dehghan, K. Faez, M. Ahmadi, and M. Shridhar, “Handwritten Farsi (Arabic) word recognition: A holistic approach using discrete HMM,” *Pattern Recognit.*, vol. 34, no. 5, pp. 1057–1065, 2001.

[185] S. K. Parui and B. Shaw, “Offline Handwritten Devanagari Word Recognition : An HMM Based Approach,” in *Proceedings of International Conference on Information Technology*, 2007, pp. 528–535.

[186] P. P. Roy, P. Dey, S. Roy, U. Pal, and F. Kimura, “A Novel Approach of Bangla Handwritten Text Recognition Using HMM,” in *Proceedings of International Conference on Frontiers in Handwriting Recognition,* 2014, pp. 661–666.

[187] T. K. Bhowmik, S. K. Parui, and U. Roy, “Discriminative HMM training with GA for handwritten word recognition,” in *Proceedings of 19th International Conference on Pattern Recognition,* 2008, pp. 1–4.

[188] C. Adak, B. B. Chaudhuri, and M. Blumenstein, “Offline cursive Bengali word recognition using CNNs with a recurrent model,” in *Proceedings of International Conference on Frontiers in Handwriting Recognition,* 2017, pp. 429–434.

[189] S. Barua, S. Malakar, S. Bhowmik, R. Sarkar, and M. Nasipuri, “Bangla handwritten city name recognition using gradient-based feature,” in *Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications,* Springer, 2017, pp. 343–352.

[190] M. Ghosh, S. Malakar, S. Bhowmik, R. Sarkar, and M. Nasipuri, “Memetic algorithm based feature selection for handwritten city name recognition,” in *Proceedings of International Conference on Computational Intelligence, Communications, and Business Analytics*, 2017, pp. 599–613.

[191] S. Sahoo, S. K. Nandi, S. Barua, Pallavi, S. Malakar, and R. Sarkar, “Handwritten Bangla city

name recognition using shape-context feature," in *Proceedings of the 6th International Conference on Frontiers in Intelligent Computing: Theory and Applications*, 2018, pp. 451–460.

[192] S. Sahoo, S. K. Nandi, S. Barua, Pallavi, S. Bhowmik, S. Malakar, and R. Sarkar, "Handwritten Bangla word recognition using negative refraction based shape transformation," *J. Intell. Fuzzy Syst.*, vol. 35, no. 2, pp. 1765–1777, 2018.

[193] M. Ghosh, S. Malakar, S. Bhowmik, R. Sarkar, and M. Nasipuri, "Feature Selection for Handwritten Word Recognition Using Memetic Algorithm," in *Advances in Intelligent Computing*, Springer, 2019, pp. 103–124.

[194] X. Chen, Y. S. Ong, M. H. Lim, and K. C. Tan, "A multi-facet survey on memetic computation," *IEEE Trans. Evol. Comput.*, vol. 15, no. 5, pp. 591–607, 2011.

[195] "CMATERdb2.1.2." [Online]. Available: https://drive.google.com/file/d/0B8rZngAQdufXemZmYlI2M2xwdXc/view. [Accessed: 17-Dec-2018].

[196] "Feature selection." [Online]. Available: https://en.wikipedia.org/wiki/Feature_selection. [Accessed: 17-Dec-2018].

[197] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of ReliefF and RReliefF," *Mach. Learn.*, vol. 53, no. 1–2, pp. 23–69, 2003.

[198] "Memetic algorithm." [Online]. Available: https://en.wikipedia.org/wiki/Memetic_algorithm. [Accessed: 17-Dec-2018].

[199] Y. S. Ong and A. J. Keane, "Meta-Lamarckian learning in memetic algorithms.pdf," *IEEE Trans. Evol. Comput.*, vol. 8, no. 2, pp. 99–110, 2004.

[200] V. Ho-Huu, T. Nguyen-Thoi, T. Truong-Khac, L. Le-Anh, and T. Vo-Duy, "An improved differential evolution based on roulette wheel selection for shape and size optimization of truss structures with frequency constraints," *Neural Comput. Appl.*, vol. 29, no. 1, pp. 167–185, 2018.

[201] M. Pechwitz, S. S. Maddouri, and V. Märgner, "IFN/ENIT-database of handwritten Arabic words," in *Proceedings of Colloque International francophone sur l'écrit et le document*, 2002, vol. 2, pp. 127–136.

[202] "Centre for Pattern Recognition and Machine Intelligence." [Online]. Available: http://www.concordia.ca/research/cenparmi.html. [Accessed: 17-Dec-2018].

[203] M. Zimmermann and H. Bunke, "Automatic segmentation of the IAM off-line database for handwritten English text," in *Proceedings of Object recognition supported by user interaction for service robots*, 2002, vol. 4, pp. 35–39.

[204] A. K. Bhunia, A. Das, P. P. Roy, and U. Pal, “A comparative study of features for handwritten Bangla text recognition,” in *Proceedings of the International Conference on Document Analysis and Recognition,* 2015, pp. 636–640.

[205] Y. Zhang, M. Chen, D. Huang, D. Wu, and Y. Li, “iDoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization,” *Futur. Gener. Comput. Syst.*, vol. 66, pp. 30–35, 2017.

[206] “Document classification.” [Online]. Available: https://en.wikipedia.org/wiki/Document_classification. [Accessed: 17-Dec-2018].

[207] S. S. Kang, “Keyword based document clustering,” in *Proceedings of the sixth international workshop on Information retrieval with Asian languages*, 2003, pp. 132–137.

[208] W. Xu, X. Liu, and Y. Gong, “Document clustering based on non-negative matrix factorization,” in *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, 2003, pp. 267–273.

[209] H. Cao, A. Bhardwaj, and V. Govindaraju, “A probabilistic method for keyword retrieval in handwritten document images,” *Pattern Recognit.*, vol. 42, no. 12, pp. 3374–3382, 2009.

[210] A. Tarafdar, U. Pal, J. Y. Ramel, N. Ragot, and B. B. Chaudhuri, “Word spotting in Bangla and English graphical documents,” in *Proceedings of International Conference on Pattern Recognition*, 2014, pp. 3044–3049.

[211] L. Rothacker, D. Fisseler, G. G. W. Müller, F. Weichert, and G. A. Fink, “Retrieving cuneiform structures in a segmentation-free word spotting framework,” in *Proceedings of the 3rd International Workshop on Historical Document Imaging and Processing*, 2015, pp. 129–136.

[212] R. Pintus, Y. Yang, E. Gobbetti, and H. Rushmeier, “An automatic word-spotting framework for medieval manuscripts,” in *Proceedings of Digital Heritage International Congress, Digital Heritage*, 2015, pp. 5–12.

[213] A. Fischer, A. Keller, V. Frinken, and H. Bunke, “Lexicon Free Handwritten Word Spotting using Character HMMs,” *Pattern Recognit. Lett.*, vol. 33, no. 7, pp. 934–942, 2012.

[214] Y. Liang, M. C. Fairhurst, and R. M. Guest, “A synthesised word approach to word retrieval in handwritten documents,” *Pattern Recognit.*, vol. 45, no. 12, pp. 4225–4236, 2012.

[215] V. Frinken, A. Fischer, M. Baumgartner, and H. Bunke, “Keyword spotting for self-training of BLSTM NN based handwriting recognition systems,” *Pattern Recognit.*, vol. 47, no. 3, pp. 1073–1082, 2014.

[216] W. Pantke, M. Dennhardt, D. Fecker, V. Märgner, and T. Fingscheidt, “An Historical

Handwritten Arabic Dataset for Segmentation-Free Word Spotting - HADARA80P," in *Proceedings of International Conference on Frontiers in Handwriting Recognition,* 2014, pp. 15–20.

[217] L. Rothacker and G. A. Fink, "Segmentation-free query-by-string word spotting with Bag-of-Features HMMs," in *Proceedings of the International Conference on Document Analysis and Recognition,* 2015, pp. 661–665.

[218] T. Mondal, N. Ragot, J. yves Ramel, and U. Pal, "Comparative study of conventional time series matching techniques for word spotting," *Pattern Recognit.*, vol. 73, pp. 47–64, 2018.

[219] H. Cao and V. Govindaraju, "Template-free word spotting in low-quality manuscripts," in *Proceedings of International Conference on Advances in Pattern Recognition*, World Scientific, 2007, pp. 1–5.

[220] S. Sudholt and G. A. Fink, "PHOCNet : A Deep Convolutional Neural Network for Word Spotting in Handwritten Documents," in *Proceedings 15th International Conference on of Frontiers in Handwriting Recognition*, 2016, pp. 277–282.

[221] R. Saabni and A. Bronstein, "Fast keyword searching using 'boostmap' based embedding," in *Proceedings of International Conference on Frontiers in Handwriting Recognition,* 2012, pp. 734–739.

[222] P. Riba, J. Llados, and A. Fornes, "Handwritten word spotting by inexact matching of grapheme graphs," in *Proceedings of the International Conference on Document Analysis and Recognition,* 2015, pp. 781–785.

[223] P. Wang, V. Eglin, C. Garcia, C. Largeron, J. Llados, and A. Fornes, "A novel learning-free word spotting approach based on graph representation," in *Proceedings of 11th IAPR International Workshop on Document Analysis Systems,* 2014, pp. 207–211.

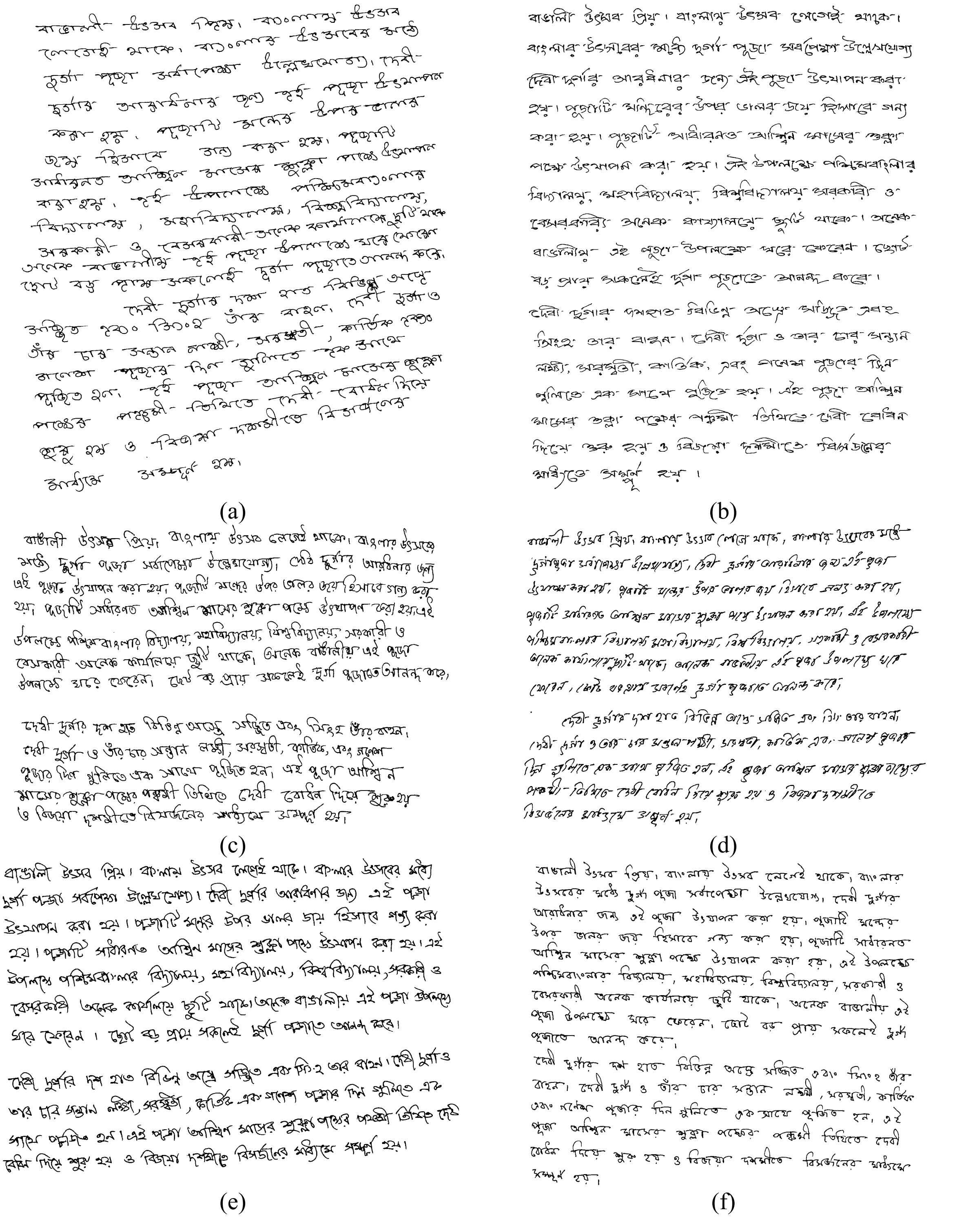
[224] P. P. Roy, J. Ramel, and N. Ragot, "Word Retrieval in Historical Document Using Character-Primitives," in *Proceedings of International Conference on Document Analysis and Recognition,* 2011, pp. 678–682.

[225] M. Khayyat, L. Lam, and C. Y. Suen, "Learning-based word spotting system for Arabic handwritten documents," *Pattern Recognit.*, vol. 47, no. 3, pp. 1021–1030, 2014.

[226] P. K. Singh, R. Sarkar, N. Das, S. Basu, M. Kundu, and M. Nasipuri, "Benchmark databases of handwritten Bangla-Roman and Devanagari-Roman mixed-script document images," *Multimed. Tools Appl.*, pp. 1–33, 2017.

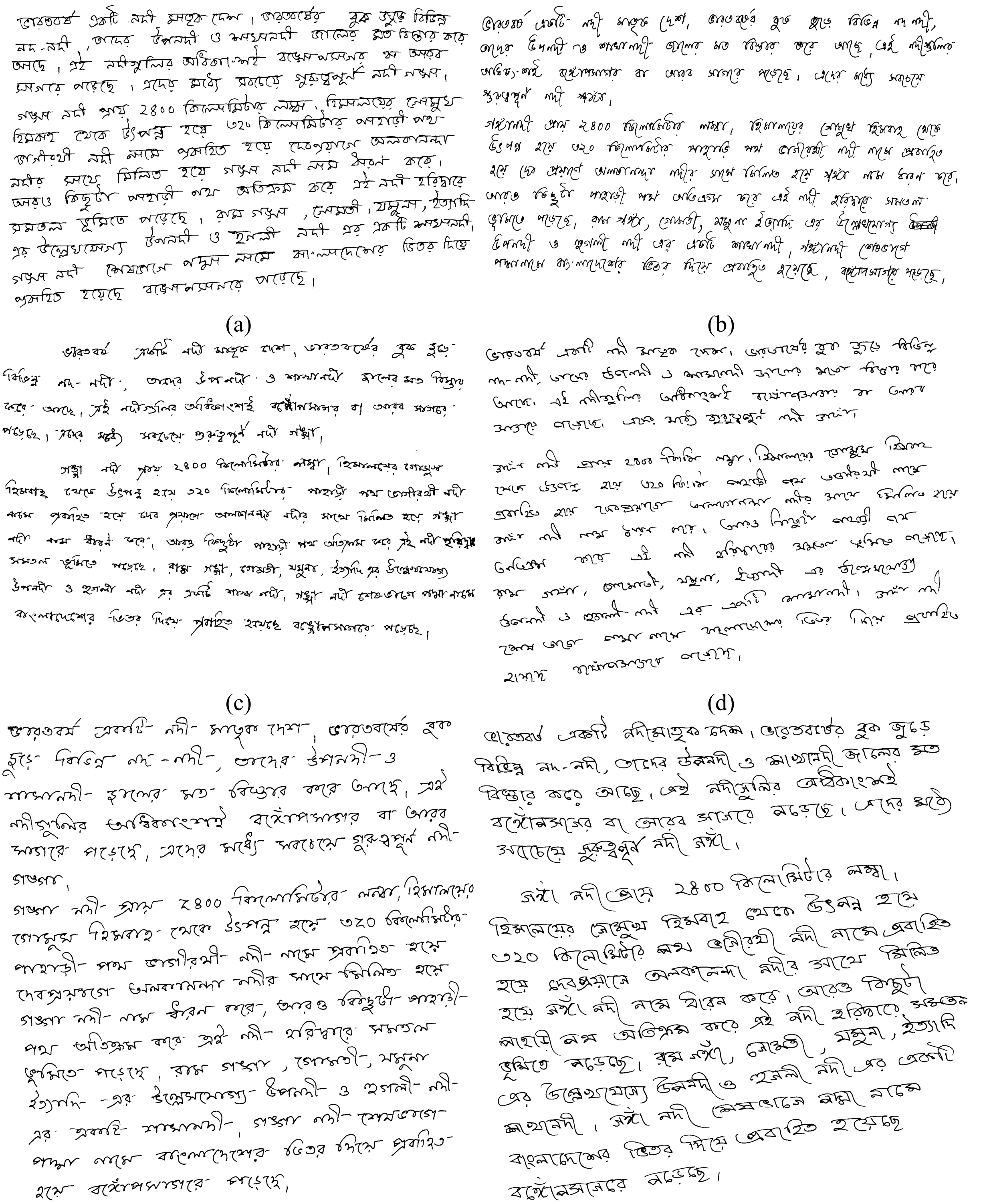
[227] "tf–idf." [Online]. Available: https://en.wikipedia.org/wiki/Tf–idf. [Accessed: 17-Dec-2018].

Appendix



(a) (b) (c) (d) (e) (f)

Fig. A1 Six samples of collected document page images of festival category (refer to Chapter 2)



ভারতবর্ষ একটি নদী মাতৃক দেশ। ভারতবর্ষের বুক জুড়ে বিভিন্ন
নদ-নদী, তাদের উপনদী ও শাখানদী জালের মত বিস্তার করে
আছে। এই নদীগুলির অধিকাংশই বঙ্গোপসাগর বা আরব
সাগরে পড়েছে। এদের মধ্যে সবচেয়ে গুরুত্বপূর্ণ নদী গঙ্গা।
গঙ্গা নদী প্রায় ২৪০০ কিলোমিটার লম্বা। হিমালয়ের গোমুখ
হিমবাহ থেকে উৎপন্ন হয়ে ৩২০ কিলোমিটার পাহাড়ী পথ
ভাগীরথী নদী নামে প্রবাহিত হয়ে দেবপ্রয়াগে অলকানন্দা
নদীর সাথে মিলিত হয়ে গঙ্গা নদী নাম ধারণ করে।
আরও কিছুটা পাহাড়ী পথ অতিক্রম করে এই নদী হরিদ্বারে
সমতল ভূমিতে পড়েছে। রাম গঙ্গা, গোমতী, যমুনা, ইত্যাদি
এর উল্লেখযোগ্য উপনদী ও হুগলী নদী এর একটি শাখানদী।
গঙ্গা নদী শেষভাগে পদ্মা নামে বাংলাদেশের ভিতর দিয়ে
প্রবাহিত হয়েছে বঙ্গোপসাগরে পড়েছে।

(a)

ভারতবর্ষ একটি নদী মাতৃক দেশ, ভারতবর্ষের বুক জুড়ে বিভিন্ন নদ নদী,
তাদের উপনদী ও শাখানদী জালের মত বিস্তার করে আছে, এই নদীগুলির
অধিকাংশই বঙ্গোপসাগর বা আরব সাগরে পড়েছে। এদের মধ্যে সবচেয়ে
গুরুত্বপূর্ণ নদী গঙ্গা,
গঙ্গানদী প্রায় ২৪০০ কিলোমিটার লম্বা, হিমালয়ের গোমুখ হিমবাহ থেকে
উৎপন্ন হয়ে ৩২০ কিলোমিটার পাহাড়ি পথ ভাগীরথী নদী নামে প্রবাহিত
হয়ে দেবপ্রয়াগে অলকানন্দা নদীর সাথে মিলিত হয়ে গঙ্গা নাম ধারণ করে,
আরও কিছুটা পাহাড়ী পথ অতিক্রম করে এই নদী হরিদ্বারে সমতল
ভূমিতে পড়েছে, রাম গঙ্গা, গোমতী, যমুনা ইত্যাদি এর উল্লেখযোগ্য ~~উপনদী~~
উপনদী ও হুগলী নদী এর একটি শাখানদী, গঙ্গানদী শেষভাগে
পদ্মানামে বাংলাদেশের ভিতর দিয়ে প্রবাহিত হয়েছে, বঙ্গোপসাগরে পড়েছে,

(b)

ভারতবর্ষ একটি নদী মাতৃক দেশ, ভারতবর্ষের বুক জুড়ে
বিভিন্ন নদ-নদী, তাদের উপনদী ও শাখানদী জালের মত বিস্তার
করে আছে, এই নদীগুলির অধিকাংশই বঙ্গোপসাগর বা আরব সাগরে
পড়েছে। এদের মধ্যে সবচেয়ে গুরুত্বপূর্ণ নদী গঙ্গা,
গঙ্গা নদী প্রায় ২৪০০ কিলোমিটার লম্বা, হিমালয়ের গোমুখ
হিমবাহ থেকে উৎপন্ন হয়ে ৩২০ কিলোমিটার পাহাড়ী পথ ভাগীরথী নদী
নামে প্রবাহিত হয়ে দেবপ্রয়াগে অলকানন্দা নদীর সাথে মিলিত হয়ে গঙ্গা
নদী নাম ধারণ করে, আরও কিছুটা পাহাড়ী পথ অতিক্রম করে এই নদী হরিদ্বারে
সমতল ভূমিতে পড়েছে, রাম গঙ্গা, গোমতী, যমুনা, ইত্যাদি এর উল্লেখযোগ্য
উপনদী ও হুগলী নদী এর একটি শাখা নদী, গঙ্গা নদী শেষভাগে পদ্মা নামে
বাংলাদেশের ভিতর দিয়ে প্রবাহিত হয়েছে বঙ্গোপসাগরে পড়েছে,

(c)

ভারতবর্ষ একটি নদী মাতৃক দেশ। ভারতবর্ষের বুক জুড়ে বিভিন্ন
নদ-নদী, তাদের উপনদী ও শাখানদী জালের মত বিস্তার করে
আছে। এই নদীগুলির অধিকাংশই বঙ্গোপসাগর বা আরব
সাগরে পড়েছে। এদের মধ্যে গুরুত্বপূর্ণ নদী গঙ্গা।
গঙ্গা নদী প্রায় ২৪০০ কিঃমিঃ লম্বা। হিমালয়ের গোমুখ হিমবাহ
থেকে উৎপন্ন হয়ে ৩২০ কিঃমিঃ পাহাড়ী পথ ভাগীরথী নামে
প্রবাহিত হয়ে দেবপ্রয়াগে অলকানন্দা নদীর সাথে মিলিত হয়ে
গঙ্গা নদী নাম ধারণ করে। আরও কিছুটা পাহাড়ী পথ
অতিক্রম করে এই নদী হরিদ্বারে সমতল ভূমিতে পড়েছে।
রাম গঙ্গা, গোমতী, যমুনা, ইত্যাদি এর উল্লেখযোগ্য
উপনদী ও হুগলী নদী এর একটি শাখানদী। গঙ্গা নদী
শেষ ভাগে পদ্মা নামে বাংলাদেশের ভিতর দিয়ে প্রবাহিত
বঙ্গোপসাগরে পড়েছে।
হয়েছে

(d)

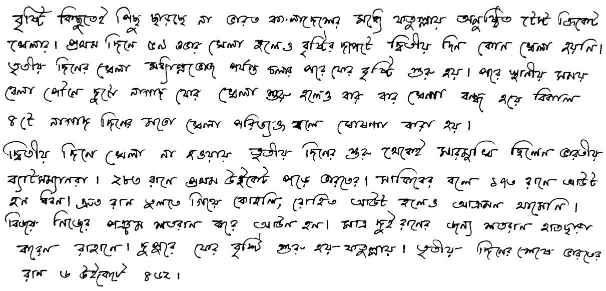
ভারতবর্ষ একটি নদী মাতৃক দেশ, ভারতবর্ষের বুক
জুড়ে বিভিন্ন নদ-নদী, তাদের উপনদী ও
শাখানদী জালের মত বিস্তার করে আছে, এই
নদীগুলির অধিকাংশই বঙ্গোপসাগর বা আরব
সাগরে পড়েছে, এদের মধ্যে সবচেয়ে গুরুত্বপূর্ণ নদী
গঙ্গা,
গঙ্গা নদী প্রায় ২৪০০ কিলোমিটার লম্বা, হিমালয়ের
গোমুখ হিমবাহ থেকে উৎপন্ন হয়ে ৩২০ কিলোমিটার
পাহাড়ী পথ ভাগীরথী নদী নামে প্রবাহিত হয়ে
দেবপ্রয়াগে অলকানন্দা নদীর সাথে মিলিত হয়ে
গঙ্গা নদী নাম ধারণ করে, আরও কিছুটা পাহাড়ী
পথ অতিক্রম করে এই নদী হরিদ্বারে সমতলে
ভূমিতে পড়েছে, রাম গঙ্গা, গোমতী, যমুনা
ইত্যাদি এর উল্লেখযোগ্য উপনদী ও হুগলী নদী
এর একটি শাখানদী, গঙ্গা নদী শেষভাগে
পদ্মা নামে বাংলাদেশের ভিতর দিয়ে প্রবাহিত
হয়ে বঙ্গোপসাগরে পড়েছে,

(e)

ভারতবর্ষ একটি নদীমাতৃক দেশ। ভারতবর্ষের বুক জুড়ে
বিভিন্ন নদ-নদী, তাদের উপনদী ও শাখানদী জালের মত
বিস্তার করে আছে। এই নদীগুলির অধিকাংশই
বঙ্গোপসাগর বা আরব সাগরে পড়েছে। এদের মধ্যে
সবচেয়ে গুরুত্বপূর্ণ নদী গঙ্গা।
গঙ্গা নদী প্রায় ২৪০০ কিলোমিটার লম্বা।
হিমালয়ের গোমুখ হিমবাহ থেকে উৎপন্ন হয়ে
৩২০ কিলোমিটার পথ ভাগীরথী নদী নামে প্রবাহিত
হয়ে দেবপ্রয়াগে অলকানন্দা নদীর সাথে মিলিত
হয়ে গঙ্গা নদী নামে ধারণ করে। আরও কিছুটা
পাহাড়ী পথ অতিক্রম করে এই নদী হরিদ্বারে সমতল ইত্যাদি
ভূমিতে পড়েছে। রাম গঙ্গা, গোমতী, যমুনা,
এর উল্লেখযোগ্য উপনদী ও হুগলী নদী এর একটি
শাখানদী। গঙ্গা নদী শেষভাগে পদ্মা নামে
বাংলাদেশের ভিতর দিয়ে প্রবাহিত হয়েছে
বঙ্গোপসাগরে পড়েছে।

(f)

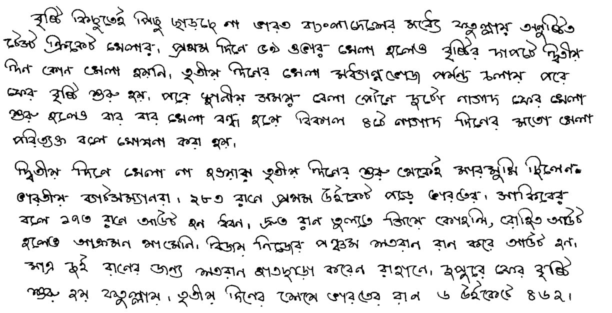
Fig. A2 Six samples of collected document page images of geographical category (refer to Chapter 2)



বৃষ্টি কিছুতেই পিছু ছাড়ছে না ভারত বাংলাদেশের মধ্যে ফতুল্লায় অনুষ্ঠিত টেস্ট ক্রিকেট
খেলার। প্রথম দিনে ৫৭ ওভার খেলা হলেও বৃষ্টির দাপটে দ্বিতীয় দিন কোন খেলা হয়নি।
তৃতীয় দিনের খেলা মধ্যাহ্নভোজ পর্যন্ত চলার পরে ফের বৃষ্টি শুরু হয়। পরে স্থানীয় সময়
বেলা পৌনে দুটো নাগাদ ফের খেলা শুরু হলেও বার বার খেলা বন্ধ হয়ে বিকাল
৪টে নাগাদ দিনের মতো খেলা পরিত্যক্ত বলে ঘোষণা করা হয়।

দ্বিতীয় দিনে খেলা না হওয়ায় তৃতীয় দিনের শুরু থেকেই মারমুখি ছিলেন ভারতীয়
ব্যাটসম্যানরা। ২৮৩ রানে প্রথম উইকেট পড়ে ভারতের। সাকিবের বলে ১৭৩ রানে আউট
হন ধবন। দ্রুত রান তুলতে গিয়ে কোহলি, রোহিত আউট হলেও আক্রমণ থামেনি।
বিজয় নিজের পঞ্চম শতরান করে আউট হন। মাত্র দুই রানের জন্য শতরান হাতছাড়া
করেন রাহানে। দুপুরে ফের বৃষ্টি শুরু হয় ফতুল্লায়। তৃতীয় দিনের শেষে ভারতের
রান ৬ উইকেটে ৪৬২।

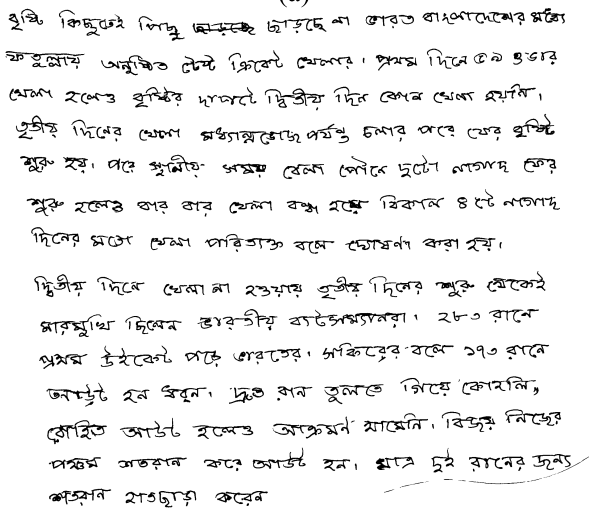
(a)



বৃষ্টি কিছুতেই পিছু ছাড়ছে না ভারত বাংলাদেশের মধ্যে ফতুল্লায় অনুষ্ঠিত
টেস্ট ক্রিকেট খেলার, প্রথম দিনে ৫৭ ওভার খেলা হলেও বৃষ্টির দাপটে দ্বিতীয়
দিন কোন খেলা হয়নি, তৃতীয় দিনের খেলা মধ্যাহ্নভোজ পর্যন্ত চলার পরে
ফের বৃষ্টি শুরু হয়, পরে স্থানীয় সময় বেলা পৌনে দুটো নাগাদ ফের খেলা
শুরু হলেও বার বার খেলা বন্ধ হয়ে বিকাল ৪টে নাগাদ দিনের মতো খেলা
পরিত্যক্ত বলে ঘোষণা করা হয়,

দ্বিতীয় দিনে খেলা না হওয়ায় তৃতীয় দিনের শুরু থেকেই মারমুখি ছিলেন
ভারতীয় ব্যাটসম্যানরা। ২৮৩ রানে প্রথম উইকেট পড়ে ভারতের। সাকিবের
বলে ১৭৩ রানে আউট হন ধবন, দ্রুত রান তুলতে গিয়ে কোহলি, রোহিত আউট
হলেও আক্রমণ থামেনি, বিজয় নিজের পঞ্চম শতরান রান করে আউট হন,
মাত্র দুই রানের জন্য শতরান হাতছাড়া করেন রাহানে, দুপুরে ফের বৃষ্টি
শুরু হয় ফতুল্লায়, তৃতীয় দিনের শেষে ভারতের রান ৬ উইকেটে ৪৬২।

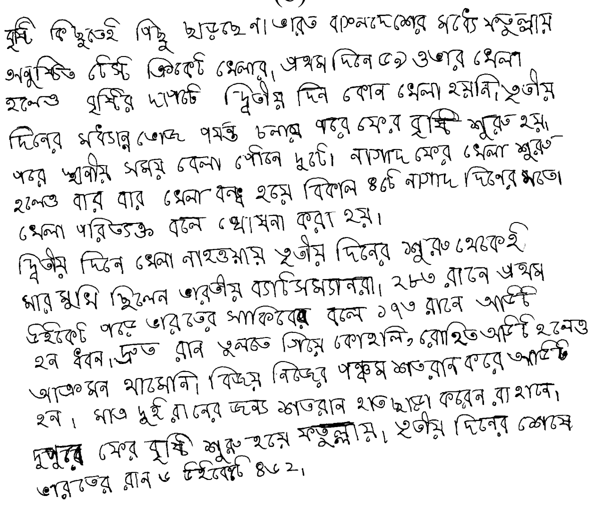
(b)



বৃষ্টি কিছুতেই পিছু ~~ছাড়ছে~~ ছাড়ছে না ভারত বাংলাদেশের মধ্যে
ফতুল্লায় অনুষ্ঠিত টেস্ট ক্রিকেট খেলার। প্রথম দিনে ৫৭ ওভার
খেলা হলেও বৃষ্টির দাপটে দ্বিতীয় দিন কোন খেলা হয়নি।
তৃতীয় দিনের খেলা মধ্যাহ্নভোজ পর্যন্ত চলার পরে ফের বৃষ্টি
শুরু হয়। পরে স্থানীয় সময় বেলা পৌনে দুটো নাগাদ ফের
শুরু হলেও বার বার খেলা বন্ধ হয়ে বিকাল ৪টে নাগাদ
দিনের মতো খেলা পরিত্যক্ত বলে ঘোষণা করা হয়।

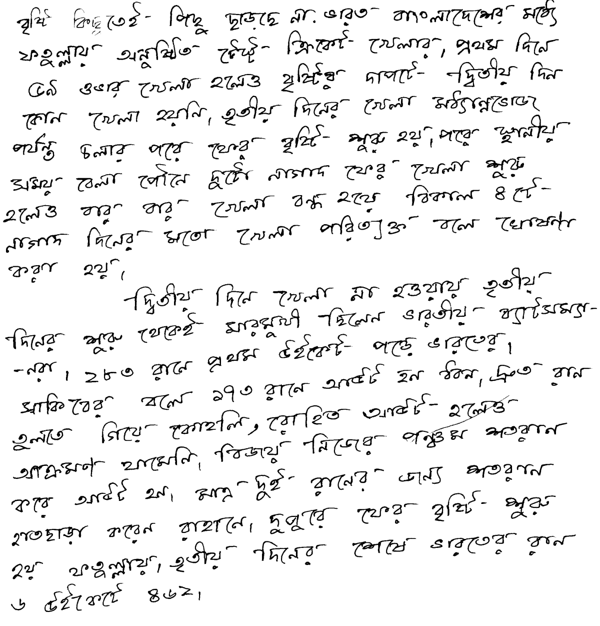
দ্বিতীয় দিনে খেলা না হওয়ায় তৃতীয় দিনের শুরু থেকেই
মারমুখি ছিলেন ভারতীয় ব্যাটসম্যানরা। ২৮৩ রানে
প্রথম উইকেট পড়ে ভারতের। সাকিবের বলে ১৭৩ রানে
আউট হন ধবন। দ্রুত রান তুলতে গিয়ে কোহলি,
রোহিত আউট হলেও আক্রমণ থামেনি। বিজয় নিজের
পঞ্চম শতরান করে আউট হন। মাত্র দুই রানের জন্য
শতরান হাতছাড়া করেন

(c)



বৃষ্টি কিছুতেই পিছু ছাড়ছে না। ভারত বাংলাদেশের মধ্যে ফতুল্লায়
অনুষ্ঠিত টেস্ট ক্রিকেট খেলার, প্রথম দিনে ৫৭ ওভার খেলা
হলেও বৃষ্টির দাপটে দ্বিতীয় দিন কোন খেলা হয়নি, তৃতীয়
দিনের মধ্যাহ্নভোজ পর্যন্ত চলার পরে ফের বৃষ্টি শুরু হয়।
পরে স্থানীয় সময় বেলা পৌনে দুটো। নাগাদ ফের খেলা শুরু
হলেও বার বার খেলা বন্ধ হয়ে বিকাল ৪টে নাগাদ দিনের মতো
খেলা পরিত্যক্ত বলে ঘোষণা করা হয়।
দ্বিতীয় দিনে খেলা না হওয়ায় তৃতীয় দিনের শুরু থেকেই
মারমুখি ছিলেন ভারতীয় ব্যাটসম্যানরা। ২৮৩ রানে প্রথম
উইকেট পড়ে ভারতের সাকিবের বলে ১৭৩ রানে আউট
হন ধবন, দ্রুত রান তুলতে গিয়ে কোহলি, রোহিত আউট হলেও
আক্রমণ থামেনি। বিজয় নিজের পঞ্চম শতরান করে আউট
হন। মাত্র দুই রানের জন্য শতরান হাতছাড়া করেন রাহানে,
দুপুরে ফের বৃষ্টি শুরু হয়ে ফতুল্লায়। তৃতীয় দিনের শেষে
ভারতের রান ৬ উইকেটে ৪৬২,

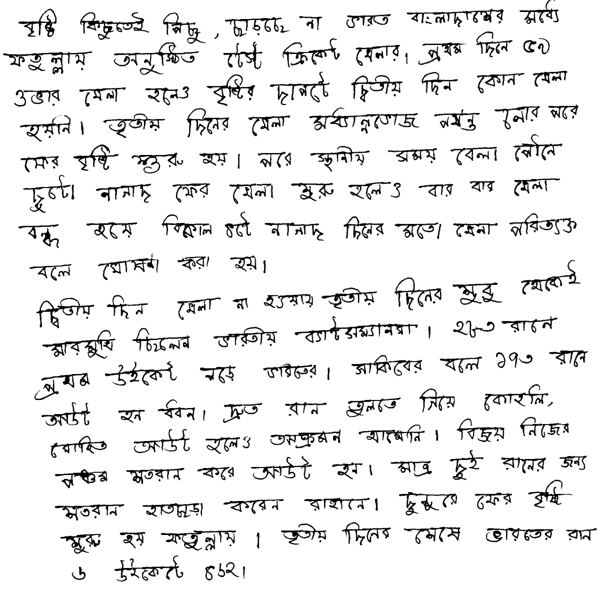
(d)



বৃষ্টি কিছুতেই পিছু ছাড়ছে না. ভারত বাংলাদেশের মধ্যে
ফতুল্লায় অনুষ্ঠিত টেস্ট ক্রিকেট খেলার, প্রথম দিনে
৫৭ ওভার খেলা হলেও বৃষ্টির দাপটে দ্বিতীয় দিন
কোন খেলা হয়নি, তৃতীয় দিনের খেলা মধ্যাহ্নভোজ
পর্যন্ত চলার পরে ফের বৃষ্টি শুরু হয়, পরে স্থানীয়
সময় বেলা পৌনে দুটো নাগাদ ফের খেলা শুরু
হলেও বার বার খেলা বন্ধ হয়ে বিকাল ৪টে
নাগাদ দিনের মতো খেলা পরিত্যক্ত বলে ঘোষণা
করা হয়,

দ্বিতীয় দিনে খেলা না হওয়ায় তৃতীয়
দিনের শুরু থেকেই মারমুখী ছিলেন ভারতীয় ব্যাটসম্যা-
-নরা। ২৮৩ রানে প্রথম উইকেট পড়ে ভারতের।
সাকিবের বলে ১৭৩ রানে আউট হন ধবন, দ্রুত রান
তুলতে গিয়ে কোহলি, রোহিত আউট হলেও
আক্রমণ থামেনি, বিজয় নিজের পঞ্চম শতরান
করে আউট হন, মাত্র দুই রানের জন্য শতরান
হাতছাড়া করেন রাহানে, দুপুরে ফের বৃষ্টি শুরু
হয় ফতুল্লায়, তৃতীয় দিনের শেষে ভারতের রান
৬ উইকেটে ৪৬২।

(e)

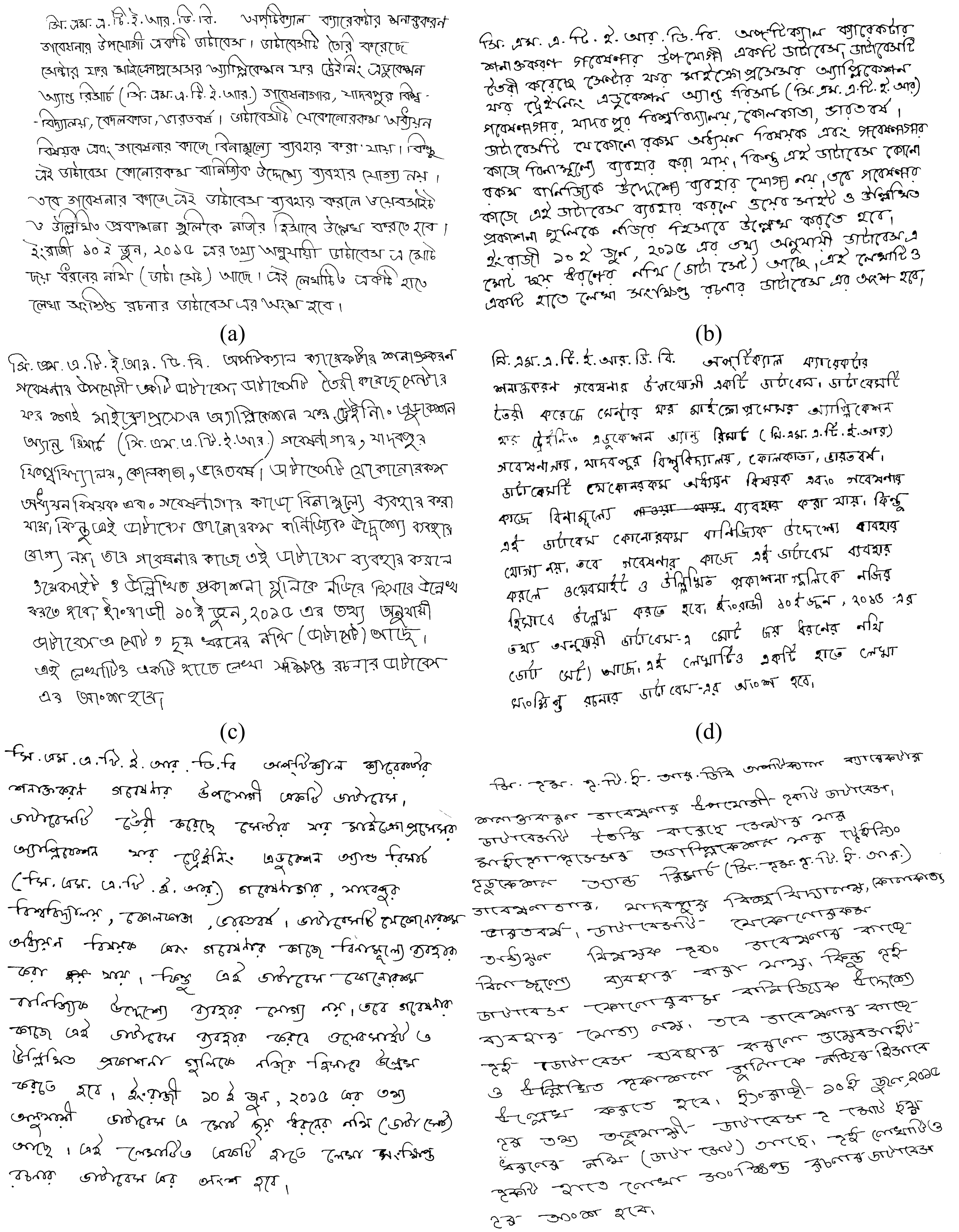


বৃষ্টি কিছুতেই পিছু, ছাড়ছে না ভারত বাংলাদেশের মধ্যে
ফতুল্লায় অনুষ্ঠিত টেস্ট ক্রিকেট খেলার, প্রথম দিনে ৫৭
ওভার খেলা হলেও বৃষ্টির দাপটে দ্বিতীয় দিন কোন খেলা
হয়নি। তৃতীয় দিনের খেলা মধ্যাহ্নভোজ পর্যন্ত চলার পরে
ফের বৃষ্টি শুরু হয়। পরে স্থানীয় সময় বেলা পৌনে
দুটো। নাগাদ ফের খেলা শুরু হলেও বার বার খেলা
বন্ধ হয়ে বিকাল ৪টে নাগাদ দিনের মতো খেলা পরিত্যক্ত
বলে ঘোষণা করা হয়।

দ্বিতীয় দিন খেলা না হওয়ায় তৃতীয় দিনের শুরু থেকেই
মারমুখি ছিলেন ভারতীয় ব্যাটসম্যানরা। ২৮৩ রানে
প্রথম উইকেট পড়ে ভারতের। সাকিবের বলে ১৭৩ রানে
আউট হন ধবন। দ্রুত রান তুলতে গিয়ে কোহলি,
রোহিত আউট হলেও আক্রমণ থামেনি। বিজয় নিজের
পঞ্চম শতরান করে আউট হন। মাত্র দুই রানের জন্য
শতরান হাতছাড়া করেন রাহানে। দুপুরে ফের বৃষ্টি
শুরু হয় ফতুল্লায়। তৃতীয় দিনের শেষে ভারতের রান
৬ উইকেটে ৪৬২।

(f)

Fig. A3 Six samples of collected document page images of sport category (refer to Chapter 2)



সি.এম.এ.টি.ই.আর.ডি.বি. অপটিক্যাল ক্যারেক্টার শনাক্তকরণ গবেষনার উপযোগী একটি ডাটাবেস। ডাটাবেসটি তৈরি করেছে সেন্টার ফর মাইক্রোপ্রসেসর অ্যাপ্লিকেশন ফর ট্রেইনিং এডুকেশন অ্যান্ড রিসার্চ (সি.এম.এ.টি.ই.আর.) গবেষনাগার, যাদবপুর বিশ্ববিদ্যালয়, কোলকাতা, ভারতবর্ষ। ডাটাবেসটি যেকোনোরকম অধ্যয়ন বিষয়ক এবং গবেষনার কাজে বিনামূল্যে ব্যবহার করা যায়। কিন্তু এই ডাটাবেস কোনোরকম বানিজ্যিক উদ্দেশ্যে ব্যবহার যোগ্য নয়। তবে গবেষনার কাজে এই ডাটাবেস ব্যবহার করলে ওয়েবসাইট ও উল্লিখিত প্রকাশনা গুলিকে নজির হিসাবে উল্লেখ করতে হবে। ইংরাজী ১০ই জুন, ২০১৫ এর তথ্য অনুযায়ী ডাটাবেস এ মোট ছয় ধরনের নথি (ডাটা সেট) আছে। এই লেখাটিও একটি হাতে লেখা সংক্ষিপ্ত রচনার ডাটাবেস এর অংশ হবে।

(a)

সি. এম. এ. টি. ই. আর. ডি. বি. অপটিক্যাল ক্যারেক্টার শনাক্তকরণ গবেষণার উপযোগী একটি ডাটাবেস, ডাটাবেসটি তৈরী করেছে সেন্টার ফর মাইক্রোপ্রসেসর অ্যাপ্লিকেশন ফর ট্রেইনিং এডুকেশন অ্যান্ড রিসার্চ (সি.এম.এ.টি.ই.আর) গবেষণাগার, যাদবপুর বিশ্ববিদ্যালয়, কোলকাতা, ভারতবর্ষ। ডাটাবেসটি যে কোনো রকম অধ্যয়ন বিষয়ক এবং গবেষণাগার কাজে বিনামূল্যে ব্যবহার করা যায়, কিন্তু এই ডাটাবেস কোনো রকম বানিজ্যিক উদ্দেশ্যে ব্যবহার যোগ্য নয়, তবে গবেষণার কাজে এই ডাটাবেস ব্যবহার করলে ওয়েবসাইট ও উল্লিখিত প্রকাশনা গুলিকে নজির হিসাবে উল্লেখ করতে হবে, ইংরাজী ১০ই জুন, ২০১৫ এর তথ্য অনুযায়ী ডাটাবেস এ মোট ছয় ধরনের নথি (ডাটা সেট) আছে, এই লেখাটিও একটি হাতে লেখা সংক্ষিপ্ত রচনার ডাটাবেস এর অংশ হবে,

(b)

সি.এম.এ.টি.ই.আর.ডি.বি. অপটিক্যাল ক্যারেক্টার শনাক্তকরণ গবেষনার উপযোগী একটি ডাটাবেস, ডাটাবেসটি তৈরী করেছে সেন্টার ফর মাই মাইক্রোপ্রসেসর অ্যাপ্লিকেশান ফর ট্রেইনিং এডুকেশান অ্যান্ড রিসার্চ (সি.এম.এ.টি.ই.আর.) গবেষনাগার, যাদবপুর বিশ্ববিদ্যালয়, কোলকাতা, ভারতবর্ষ। ডাটাবেসটি যেকোনোরকম অধ্যয়ন বিষয়ক এবং গবেষনাগার কাজে বিনামূল্যে ব্যবহার করা যায়, কিন্তু এই ডাটাবেস কোনোরকম বানিজ্যিক উদ্দেশ্যে ব্যবহার যোগ্য নয়, তবে গবেষনার কাজে এই ডাটাবেস ব্যবহার করলে ওয়েবসাইট ও উল্লিখিত প্রকাশনা গুলিকে নজিরে হিসাবে উল্লেখ করতে হবে, ইংরাজী ১০ই জুন, ২০১৫ এর তথ্য অনুযায়ী ডাটাবেসে মোট ৬ ছয় ধরনের নথি (ডাটাসেট) আছে। এই লেখাটিও একটি হাতে লেখা সংক্ষিপ্ত রচনার ডাটাবেস এর অংশ হবে,

(c)

সি.এম.এ.টি.ই.আর.ডি.বি. অপটিক্যাল ক্যারেক্টার শনাক্তকরণ গবেষনার উপযোগী একটি ডাটাবেস। ডাটাবেসটি তৈরী করেছে সেন্টার ফর মাইক্রোপ্রসেসর অ্যাপ্লিকেশন ফর ট্রেইনিং এডুকেশন অ্যান্ড রিসার্চ (সি.এম.এ.টি.ই.আর) গবেষনাগার, যাদবপুর বিশ্ববিদ্যালয়, কোলকাতা, ভারতবর্ষ। ডাটাবেসটি যেকোনরকম অধ্যয়ন বিষয়ক এবং গবেষনার কাজে বিনামূল্যে ~~পাওয়া যায়~~ ব্যবহার করা যায়। কিন্তু এই ডাটাবেস কোনোরকম বানিজ্যিক উদ্দেশ্যে ব্যবহার যোগ্য নয়। তবে গবেষনার কাজে এই ডাটাবেস ব্যবহার করলে ওয়েবসাইট ও উল্লিখিত প্রকাশনা গুলিকে নজির হিসাবে উল্লেখ করতে হবে। ইংরাজী ১০ই জুন, ২০১৫-এর তথ্য অনুযায়ী ডাটাবেস-এ মোট ছয় ধরনের নথি (ডাটা সেট) আছে। এই লেখাটিও একটি হাতে লেখা সংক্ষিপ্ত রচনার ডাটাবেস-এর অংশ হবে।

(d)

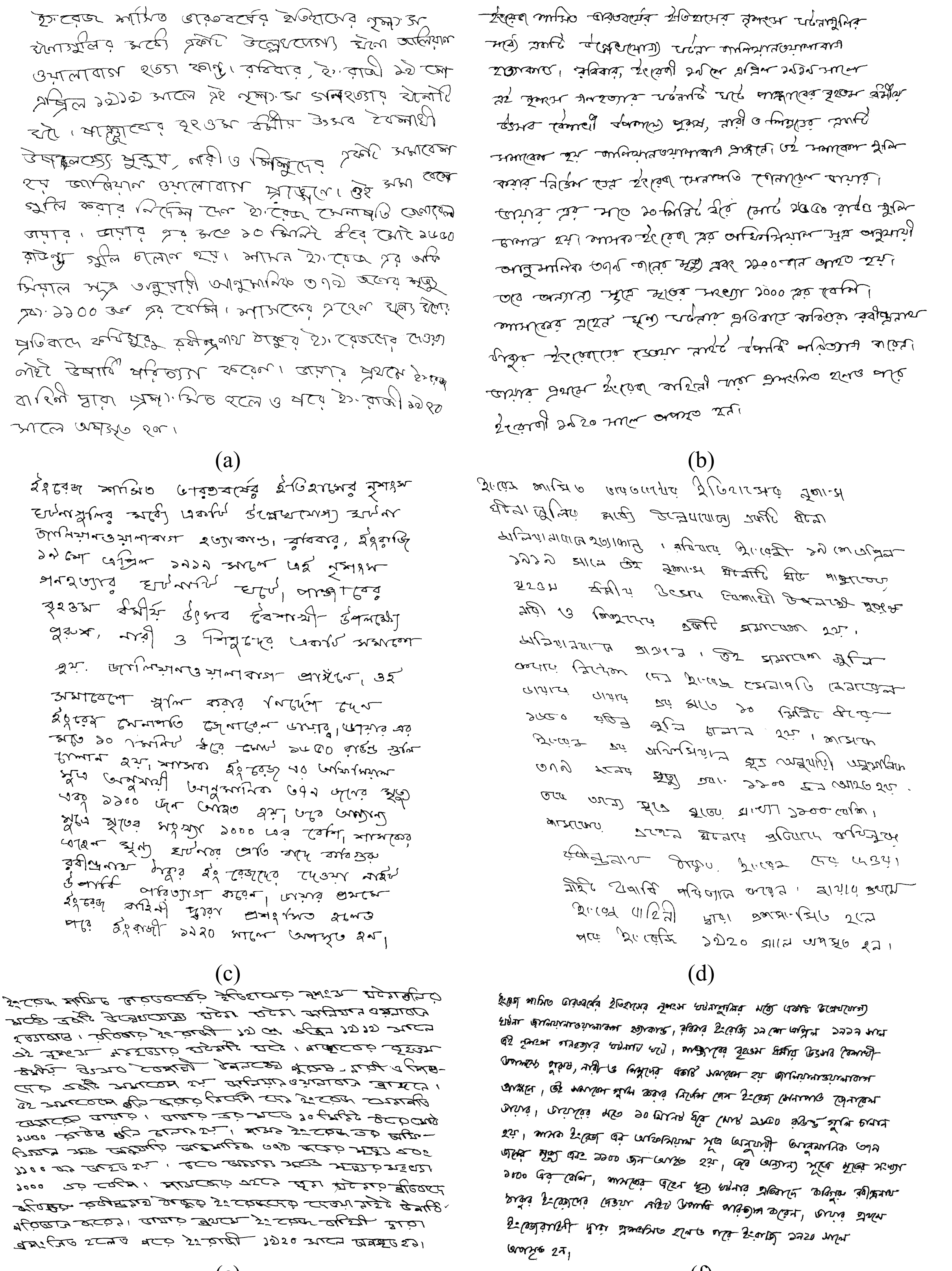
সি.এম.এ.টি.ই.আর.ডি.বি অপটিক্যাল ক্যারেক্টার শনাক্তকরণ গবেষনার উপযোগী একটি ডাটাবেস, ডাটাবেসটি তৈরী করেছে সেন্টার ফর মাইক্রোপ্রসেসর অ্যাপ্লিকেশন ফর ট্রেইনিং এডুকেশন অ্যান্ড রিসার্চ (সি.এম.এ.টি.ই.আর) গবেষনাগার, যাদবপুর বিশ্ববিদ্যালয়, কোলকাতা, ভারতবর্ষ। ডাটাবেসটি যেকোনোরকম অধ্যয়ন বিষয়ক এবং গবেষনার কাজে বিনামূল্যে ব্যবহার করা যায়। কিন্তু এই ডাটাবেস কোনোরকম বানিজ্যিক উদ্দেশ্যে ব্যবহার যোগ্য নয়, তবে গবেষনার কাজে এই ডাটাবেস ব্যবহার করলে ওয়েবসাইট ও উল্লিখিত প্রকাশনা গুলিকে নজির হিসাবে উল্লেখ করতে হবে। ইংরাজী ১০ই জুন, ২০১৫ এর তথ্য অনুযায়ী ডাটাবেস এ মোট ছয় ধরনের নথি (ডাটা সেট) আছে। এই লেখাটিও একটি হাতে লেখা সংক্ষিপ্ত রচনার ডাটাবেস এর অংশ হবে।

(e)

সি.এম.এ.টি.ই.আর.ডিবি অপটিক্যাল ক্যারেক্টার শনাক্তকরণ গবেষনার উপযোগী একটি ডাটাবেস। ডাটাবেসটি তৈরি করেছে সেন্টার ফর মাইক্রোপ্রসেসর অ্যাপ্লিকেশন ফর ট্রেইনিং এডুকেশন অ্যান্ড রিসার্চ (সি.এম.এ.টি.ই.আর.) গবেষনাগার, যাদবপুর বিশ্ববিদ্যালয়, কোলকাতা, ভারতবর্ষ। ডাটাবেসটি যেকোনোরকম অধ্যয়ন বিষয়ক এবং গবেষনার কাজে বিনামূল্যে ব্যবহার করা যায়। কিন্তু এই ডাটাবেস কোনোরকম বানিজ্যিক উদ্দেশ্যে ব্যবহার যোগ্য নয়। তবে গবেষনার কাজে এই ডাটাবেস ব্যবহার করলে ওয়েবসাইট ও উল্লিখিত প্রকাশনা গুলিকে নজির হিসাবে উল্লেখ করতে হবে। ইংরাজী ১০ই জুন, ২০১৫ এর তথ্য অনুযায়ী ডাটাবেস এ মোট ছয় ধরনের নথি (ডাটা সেট) আছে। এই লেখাটিও একটি হাতে লেখা সংক্ষিপ্ত রচনার ডাটাবেস এর অংশ হবে।

(f)

Fig. A4 Six samples of collected document page images of technological category (refer to Chapter 2)



ইংরেজ শাসিত ভারতবর্ষের ইতিহাসের নৃশংস
ঘটনাগুলির মধ্যে একটি উল্লেখযোগ্য ঘটনা জালিয়ান
ওয়ালাবাগ হত্যা কাণ্ড। রবিবার, ইংরাজী ১৩ সে
এপ্রিল ১৯১৯ সালে এই নৃশংস গণহত্যার ঘটনাটি
ঘটে। পাঞ্জাবের বৃহত্তম বর্ষীয় উৎসব বৈশাখী
উপলক্ষ্যে পুরুষ, নারী ও শিশুদের একটি সমাবেশ
হয় জালিয়ান ওয়ালাবাগ প্রাঙ্গণে। ওই সমাবেশে
গুলি করার নির্দেশ দেন ইংরেজ সেনাপতি জেনারেল
ডায়ার। ডায়ার এর মতে ১০ মিনিট ধরে মোট ১৬৫০
রাউন্ড গুলি চালান হয়। শাসক ইংরেজ এর অফি
সিয়াল সূত্র অনুযায়ী আনুমানিক ৩৭৯ জনের মৃত্যু
এবং ১১০০ জন এর বেশি। শাসকের এহেন ঘৃণ্য ঘটনার
প্রতিবাদে কবিগুরু রবীন্দ্রনাথ ঠাকুর ইংরেজদের দেওয়া
নাইট উপাধি পরিত্যাগ করেন। ডায়ার প্রথমে ইংরেজ
বাহিনী দ্বারা প্রশংসিত হলেও পরে ইংরাজী ১৯২০
সালে অপসৃত হন।

(a)

ইংরেজ শাসিত ভারতবর্ষের ইতিহাসের নৃশংস ঘটনাগুলির
মধ্যে একটি উল্লেখযোগ্য ঘটনা জালিয়ানওয়ালাবাগ
হত্যাকাণ্ড। রবিবার, ইংরেজী ১৩ই সে এপ্রিল ১৯১৯ সালে
এই নৃশংস গণহত্যার ঘটনাটি ঘটে পাঞ্জাবের বৃহত্তম বর্ষীয়
উৎসব বৈশাখী উপলক্ষে পুরুষ, নারী ও শিশুদের একটি
সমাবেশ হয় জালিয়ানওয়ালাবাগ প্রাঙ্গণে। ওই সমাবেশে গুলি
করার নির্দেশ দেন ইংরেজ সেনাপতি জেনারেল ডায়ার।
ডায়ার এর মতে ১০ মিনিট ধরে মোট ১৬৫০ রাউন্ড গুলি
চালানো হয়। শাসক ইংরেজ এর অফিসিয়াল সূত্র অনুযায়ী
আনুমানিক ৩৭৯ জনের মৃত্যু এবং ১১০০ জন আহত হয়।
তবে অন্যান্য সূত্রে মৃতের সংখ্যা ১০০০ এর বেশি।
শাসকের এহেন ঘৃণ্য ঘটনার প্রতিবাদে কবিগুরু রবীন্দ্রনাথ
ঠাকুর ইংরেজদের দেওয়া নাইট উপাধি পরিত্যাগ করেন।
ডায়ার প্রথমে ইংরেজ বাহিনী দ্বারা প্রশংসিত হলেও পরে
ইংরেজী ১৯২০ সালে অপসৃত হন।

(b)

ইংরেজ শাসিত ভারতবর্ষের ইতিহাসের নৃশংস
ঘটনাগুলির মধ্যে একটি উল্লেখযোগ্য ঘটনা
জালিয়ানওয়ালাবাগ হত্যাকাণ্ড। রবিবার, ইংরাজি
১৩ সে এপ্রিল ১৯১৯ সালে এই নৃশংস
গণহত্যার ঘটনাটি ঘটে, পাঞ্জাবের
বৃহত্তম বর্ষীয় উৎসব বৈশাখী উপলক্ষ্যে
পুরুষ, নারী ও শিশুদের একটি সমাবেশ
হয় জালিয়ানওয়ালাবাগ প্রাঙ্গণে, ওই
সমাবেশে গুলি করার নির্দেশ দেন
ইংরেজ সেনাপতি জেনারেল ডায়ার, ডায়ার এর
মতে ১০ মিনিট ধরে মোট ১৬৫০ রাউন্ড গুলি
চালান হয়, শাসক ইংরেজ এর অফিসিয়াল
সূত্র অনুযায়ী আনুমানিক ৩৭৯ জনের মৃত্যু
এবং ১১০০ জন আহত হয়, তবে অন্যান্য
সূত্রে মৃতের সংখ্যা ১০০০ এর বেশি, শাসকের
এহেন ঘৃণ্য ঘটনার প্রতি বাদে কবিগুরু
রবীন্দ্রনাথ ঠাকুর ইংরেজদের দেওয়া নাইট
উপাধি পরিত্যাগ করেন, ডায়ার প্রথমে
ইংরেজ বাহিনী দ্বারা প্রশংসিত হলেও
পরে ইংরাজী ১৯২০ সালে অপসৃত হন,

(c)

ইংরেজ শাসিত ভারতবর্ষের ইতিহাসের নৃশংস
ঘটনাগুলির মধ্যে উল্লেখযোগ্য একটি ঘটনা
জালিয়ানওয়ালাবাগ হত্যাকাণ্ড। রবিবারে ইংরেজী ১৩ সে এপ্রিল
১৯১৯ সালে এই নৃশংস ঘটনাটি ঘটে পাঞ্জাবের
বৃহত্তম বর্ষীয় উৎসব বৈশাখী উপলক্ষ্যে পুরুষ
নারী ও শিশুদের একটি সমাবেশ হয়,
জালিয়ানওয়ালাবাগ প্রাঙ্গণে। ওই সমাবেশে গুলি
করার নির্দেশ দেন ইংরেজ সেনাপতি জেনারেল
ডায়ার ডায়ার এর মতে ১০ মিনিট ধরে
১৬৫০ রাউন্ড গুলি চালান হয়। শাসক
ইংরেজ এর অফিসিয়াল সূত্র অনুযায়ী আনুমানিক
৩৭৯ জনের মৃত্যু এবং ১১০০ জন আহত হয়.
তবে অন্য সূত্রে মৃতের সংখ্যা ১০০০ বেশি,
শাসকের এহেন ঘটনার প্রতিবাদে কবিগুরু
রবীন্দ্রনাথ ঠাকুর ইংরেজ দের দেওয়া
নাইট উপাধি পরিত্যাগ করেন। ডায়ার প্রথমে
ইংরেজ বাহিনী দ্বারা প্রশংসিত হলে
পরে ইংরেজি ১৯২০ সালে অপসৃত হন।

(d)

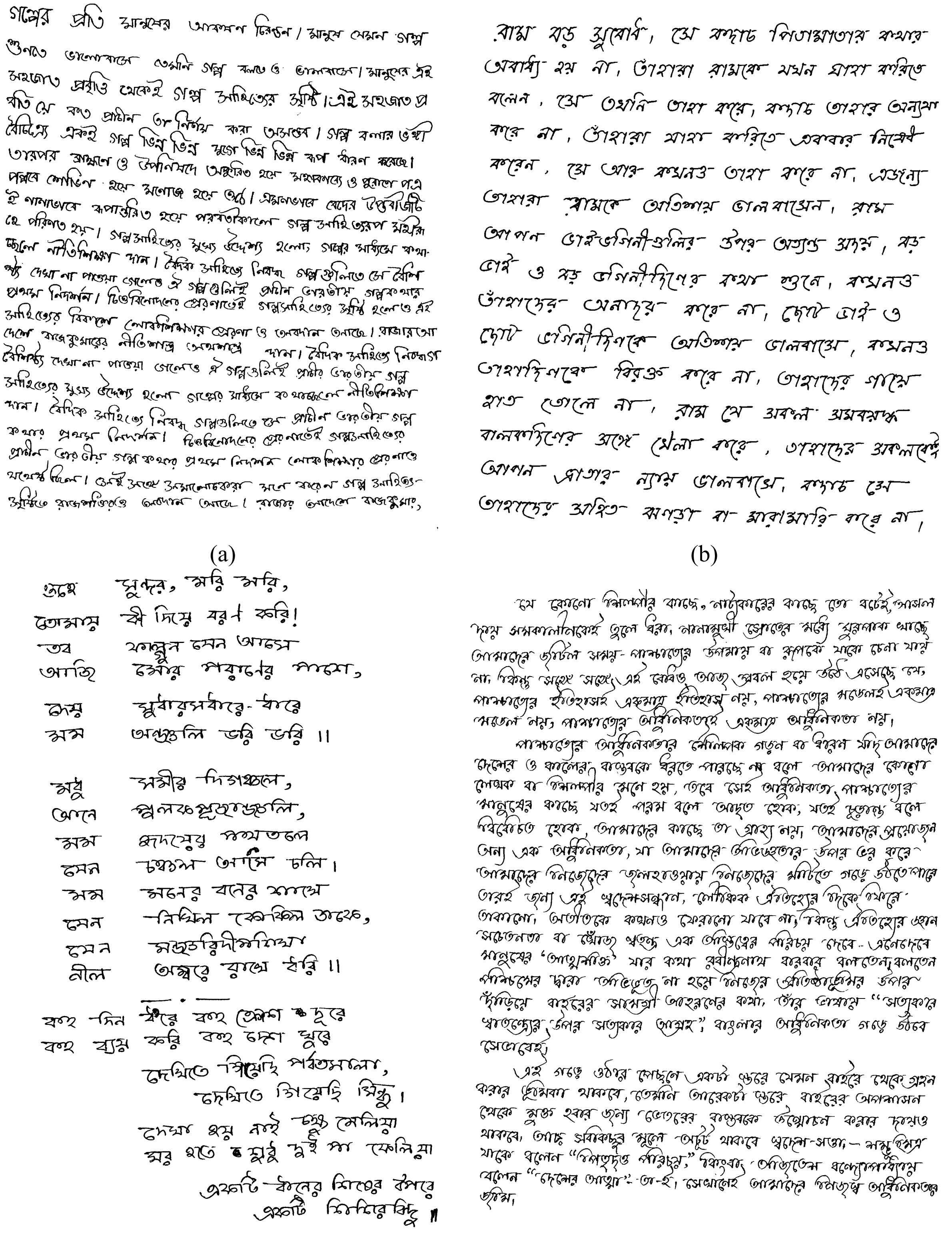
ইংরেজ শাসিত ভারতবর্ষের ইতিহাসের নৃশংস ঘটনাগুলির
মধ্যে একটি উল্লেখযোগ্য ঘটনা ঘটনা জালিয়ানওয়ালাবাগ
হত্যাকাণ্ড। রবিবার ইংরাজী ১৩ সে এপ্রিল ১৯১৯ সালে
এই নৃশংস গণহত্যার ঘটনাটি ঘটে। পাঞ্জাবের বৃহত্তম
বর্ষীয় উৎসব বৈশাখী উপলক্ষ্যে পুরুষ, নারী ও শিশু-
দের একটি সমাবেশ হয় জালিয়ানওয়ালাবাগ প্রাঙ্গণে।
ওই সমাবেশে গুলি করার নির্দেশ দেন ইংরেজ সেনাপতি
জেনারেল ডায়ার। ডায়ার এর মতে ১০ মিনিট ধরে মোট
১৬৫০ রাউন্ড গুলি চালান হয়। শাসক ইংরেজ এর অফি-
সিয়াল সূত্র অনুযায়ী আনুমানিক ৩৭৯ জনের মৃত্যু এবং
১১০০ জন আহত হয়। তবে অন্যান্য সূত্রে মৃতের সংখ্যা
১০০০ এর বেশি। শাসকের এহেন ঘৃণ্য ঘটনার প্রতিবাদে
কবিগুরু রবীন্দ্রনাথ ঠাকুর ইংরেজদের দেওয়া নাইট উপাধি
পরিত্যাগ করেন। ডায়ার প্রথমে ইংরেজ বাহিনী দ্বারা
প্রশংসিত হলেও পরে ইংরাজী ১৯২০ সালে অপসৃত হন।

(e)

ইংরেজ শাসিত ভারতবর্ষের ইতিহাসের নৃশংস ঘটনাগুলির মধ্যে একটি উল্লেখযোগ্য
ঘটনা জালিয়ানওয়ালাবাগ হত্যাকাণ্ড, রবিবার ইংরেজি ১৩ সে এপ্রিল ১৯১৯ সালে
এই নৃশংস গণহত্যার ঘটনাটি ঘটে, পাঞ্জাবের বৃহত্তম বর্ষীয় উৎসব বৈশাখী
উপলক্ষে পুরুষ, নারী ও শিশুদের একটি সমাবেশ হয় জালিয়ানওয়ালাবাগ
প্রাঙ্গণে, ওই সমাবেশে গুলি করার নির্দেশ দেন ইংরেজ সেনাপতি জেনারেল
ডায়ার, ডায়ারের মতে ১০ মিনিট ধরে মোট ১৬৫০ রাউন্ড গুলি চালান
হয়, শাসক ইংরেজ এর অফিসিয়াল সূত্র অনুযায়ী আনুমানিক ৩৭৯
জনের মৃত্যু এবং ১১০০ জন আহত হয়, তবে অন্যান্য সূত্রে মৃতের সংখ্যা
১০০০ এর বেশি, শাসকের এহেন ঘৃণ্য ঘটনার প্রতিবাদে কবিগুরু রবীন্দ্রনাথ
ঠাকুর ইংরেজদের দেওয়া নাইট উপাধি পরিত্যাগ করেন, ডায়ার প্রথমে
ইংরেজবাহিনী দ্বারা প্রশংসিত হলেও পরে ইংরাজি ১৯২০ সালে
অপসৃত হন,

(f)

Fig. A5 Six samples of collected document page images of historical category (refer to Chapter 2)



গল্পের প্রতি মানুষের আকর্ষণ চিরন্তন। মানুষ যেমন গল্প
শুনতে ভালোবাসে তেমনি গল্প বলতেও ভালবাসে। মানুষের এই
সহজাত প্রবৃত্তি থেকেই গল্প সাহিত্যের সৃষ্টি। এই সহজাত প্র
বৃত্তি যে কত প্রাচীন তা নির্ণয় করা অসম্ভব। গল্প বলার ভঙ্গী
বৈচিত্র্যে একই গল্প ভিন্ন ভিন্ন যুগে ভিন্ন ভিন্ন রূপ ধারণ করেছে।
তারপর ব্রাহ্মণে ও উপনিষদে অঙ্কুরিত হয়ে মহাকাব্যে ও পুরাণে পত্র
পল্লবে শোভিত হয়ে মনোজ্ঞ হয়ে ওঠে। এমনভাবে বেদের উপ্তবীজটি
ই নানাভাবে রূপান্তরিত হয়ে পরবর্তীকালে গল্প সাহিত্যরূপ মহীরু
হে পরিণত হয়। গল্পসাহিত্যের মুখ্য উদ্দেশ্য হলো গল্পের মাধ্যমে কথা-
চ্ছলে নীতিশিক্ষা দান। বৈদিক সাহিত্যে নিবদ্ধ গল্পগুলিতে সে বৈশি
ষ্ট্য দেখা না পাওয়া গেলেও ঐ গল্পগুলিই প্রাচীন ভারতীয় গল্পকথার
প্রথম নিদর্শন। চিত্তবিনোদনের প্রেরণাতেই গল্পসাহিত্যের সৃষ্টি হলেও এই
সাহিত্যের বিকাশে লোকশিক্ষার প্রেরণা ও অবদান আছে। রাজার আ
দেশে রাজকুমারের নীতিশাস্ত্র অর্থশাস্ত্রে দান। বৈদিক সাহিত্যে নিবদ্ধ
বৈশিষ্ট্য দেখা না পাওয়া গেলেও ঐ গল্পগুলিই প্রাচীন ভারতীয় গল্প
সাহিত্যের মুখ্য উদ্দেশ্য হলো গল্পের মাধ্যমে কথাচ্ছলে নীতিশিক্ষা
দান। বৈদিক সাহিত্যে নিবদ্ধ গল্পগুলিতে যে প্রাচীন ভারতীয় গল্প
কথার প্রথম নিদর্শন। চিত্তবিনোদনের প্রেরণাতেই গল্পসাহিত্যের
প্রাচীন ভারতীয় গল্প কথার প্রথম নিদর্শন লোকশিক্ষার প্রেরণাও
যথেষ্ট ছিল। তাই অনেক সমালোচকরা মনে করেন গল্প সাহিত্য-
সৃষ্টিতে রাজশক্তিরও অবদান আছে। রাজার আদেশে রাজকুমার,

(a)

রাম বড় সুবোধ, সে কদাচ পিতামাতার কথার
অবাধ্য হয় না, তাঁহারা রামকে যখন যাহা করিতে
বলেন, সে তখনি তাহা করে, কদাচ তাহার অন্যথা
করে না, তাঁহারা যাহা করিতে একবার নিষেধ
করেন, সে আর কখনও তাহা করে না, এজন্যে
তাহারা রামকে অতিশয় ভালবাসেন, রাম
আপন ভাই-ভগিনীগুলির উপর অত্যন্ত সদয়, বড়
ভাই ও বড় ভগিনীদিগের কথা শুনে, কখনও
তাঁহাদের অনাদর করে না, ছোট ভাই ও
ছোট ভগিনীদিগকে অতিশয় ভালবাসে, কখনও
তাহাদিগকে বিরক্ত করে না, তাহাদের গায়ে
হাত তোলে না, রাম যে সকল সমবয়স্ক
বালকদিগের সঙ্গে খেলা করে, তাহাদের সকলকেই
আপন ভ্রাতার ন্যায় ভালবাসে, কদাচ কেন
তাহাদের সহিত ঝগড়া বা মারামারি করে না,

(b)

ওহে সুন্দর, মরি মরি,
তোমায় কী দিয়ে বরণ করি!
তব ফাল্গুন যেন আসে
আজি মোর পরানের পাশে,
দেয় সুধারসধারে-ধারে
মম অঞ্জলি ভরি ভরি ॥

মধু সমীর দিগঞ্চলে,
আনে পুলকপূজাঞ্জলি,
মম হৃদয়ের পথতলে
যেন চঞ্চল আসে চলি।
মম মনের বনের শাখে
যেন নিখিল কোকিল ডাকে,
যেন মঞ্জরিদীপশিখা
নীল অম্বরে রাখে ধরি ॥

বহু দিন ধরে বহু ক্রোশ দূরে
বহু ব্যয় করি বহু দেশ ঘুরে
দেখিতে গিয়েছি পর্বতমালা,
দেখিতে গিয়েছি সিন্ধু।
দেখা হয় নাই চক্ষু মেলিয়া
ঘর হতে শুধু দুই পা ফেলিয়া
একটি ধানের শিষের উপরে
একটি শিশিরবিন্দু ॥

(c)

যে কোনো শিল্পীর কাছে, নাট্যকারের কাছে তো বটেই, আসল
দায় সমকালীনকেই তুলে ধরা, নানামুখী স্রোতের মধ্যে ঘুরপাক খাচ্ছে
আমাদের জটিল সময়- পাশ্চাত্যের উপমায় বা রূপকে যাকে চেনা যায়
না, কিন্তু সত্ত্বে সত্ত্বে, এই বোধও আজ প্রবল হয়ে উঠে এসেছে যে,
পাশ্চাত্যের ইতিহাসই একমাত্র ইতিহাস নয়, পাশ্চাত্যের মডেলই একমাত্র
মডেল নয়, পাশ্চাত্যের আধুনিকতাই একমাত্র আধুনিকতা নয়,
পাশ্চাত্যের আধুনিকতার মৌলিক গড়ন বা বিন্যাস যদি আমাদের
দেশের ও কালের বাস্তবকে ধরতে পারছে না বলে আমাদের কোনো
লেখক বা শিল্পীর মনে হয়, তবে সেই আধুনিকতা, পাশ্চাত্যের
মানুষের কাছে যতই পরম বলে আদৃত হোক, যতই চূড়ান্ত বলে
বিবেচিত হোক, আমাদের কাছে তা গ্রাহ্য নয়, আমাদের প্রয়োজন
অন্য এক আধুনিকতা, যা আমাদের অভিজ্ঞতার উপর ভর করে
আমাদের নিজেদের জলহাওয়ায় নিজেদের মাটিতে গড়ে উঠতে পারে
তারই জন্য এই স্বদেশসন্ধান, লৌকিক ঐতিহ্যের দিকে ফিরে
তাকানো, অতীতকে কখনও ফেরানো যাবে না, কিন্তু ঐতিহ্যের জ্ঞান
সচেতনতা বা খোঁজ স্বতন্ত্র এক অস্তিত্বের পরিচয় দেবে- এনে দেবে
মানুষের 'আত্মশক্তি' যার কথা রবীন্দ্রনাথ বারবার বলতেন, বলতেন
পশ্চিমের দ্বারা অভিভূত না হয়ে নিজের প্রতিষ্ঠাভূমির উপর
দাঁড়িয়ে বাইরের সামগ্রী আহরণের কথা, তাঁর ভাষায় "সত্যকার
স্বাতন্ত্র্যের উপর সত্যকার আগ্রহ", বাংলার আধুনিকতা গড়ে উঠবে
সেখানেই,

এই গড়ে ওঠার পেছনে একটা স্তরে যেমন বাইরে থেকে গ্রহণ
করার ভূমিকা থাকবে, তেমনি আরেকটা স্তরে বাইরের অপশাসন
থেকে মুক্ত হবার জন্য ভেতরের বাস্তবকে উন্মোচন করার দায়ও
থাকবে, আর সবকিছুর মূলে অটুট থাকবে স্বদেশ-সত্তা, — শম্ভু মিত্র
যাকে বলেন "নিজস্ব পরিচয়," কিংবা অজিতেশ বন্দ্যোপাধ্যায়
বলেন "দেশের আত্মা"- তা-ই, সেখানেই আমাদের নিজস্ব আধুনিকতার
জমি,

(d)

Fig. A6 Four samples of collected document page images of miscellaneous category (refer to Chapter 2)

Table A1 Sample isolated handwritten word images used for evaluating present character extraction technique (refer to Chapter 5)

অংশ	অতিশয়	অভিকাঙ্ক্ষা	অপসৃত	অভি
আধাআধি	আনুমানিক	আমার	আলেখিনা	আরব
ইংরাজী	ইংরেজ	ইজেরা	ইন্দুর -	ইমারত
ঈগল	ঈর্ষা	ঈমান	ঈপ্সিকা	ঈপ্সিতা
উৎসব	উথালপাথাল	উপাধি	উপযোগী	উষ্ণ
ঊর্মিলা	উলূক	ঊষা	ঊষসী	উলূপী
ঋগ্বেদ	ঋজুরেখা	ঋণী	ঋতু	ঋতুরাজ
এই	এক	একযালি	এযাবৎ	একলা
ঐকতান	ঐকার	ঐকাহিক	ঐশী	ঐহিক
ওঠান	ওদিক-	ওপার	ওপড়ানো	ওরাও - ওঠাও
ঔৎপাতিক	ঔৎসুক	ঔদরিক	ঔদার্য	ঔষধি
কখন	কুকুর	কণভোজা	কাঁথা	কৈলাস
খাওয়া	খিটমিট	খবর	খ্যাতিমান	খেলা
গগন	গৈত্রে	গোগ্রাসি	গবেষণা	গর্জন
ঘটনা	ঘুম	ঘর্মল	ঘড়িয়াল	ঘষণা
চাঁদ	চামচিকে	চালু	চশমা	চৈতালী-

ছিটকানি	ছাই	ছটফটানি	ছুরি	ছুঁয়ে
জগৎ	জাদু	জহারক	জীবৎ	জমিয়ে
ঝকঝক	ঝাঁজালো	ঝাল	ঝমঝম	ঝরা
টুকটুকে	টাঁকশাল	টাঙিয়ে	টোপর	টলটল
ঠুংরি	ঠিকুজি	ঠাকুরমা	ঠেঙিয়ে	ঠোঙা
ডুগডুগি	ডুলি	ডাকসাইটে	ডিঙিয়ে	ডোরা
ঢংঢং	ঢাকঢোল	ঢেউ	ঢোঁড়া	ঢেঁকিশাল
তন্দ্রা	তাহাদের	তৃতীয়	তরণী	তুলে
থানকুনি	থুতনি	থরথর	থামল	থোড়
দুইভাই	দর্জি	দীপাবলী	দুপুর	দুর্গা
ধৈর্য	ধারালো	ধূধূ	ধর্মীয়	ধুলো
নলকূপ	নারায়ণ	নারী	নিমেষে	নর্মদা
পাখি	পাগলামো	পাহাড়ী	পিঁপড়ে	প্রমুখ
ফাঁদ	ফিঙে	ফস্ ফস্	ফুল	ফসকে
বাংলা	বাতেলা	বাবু	বিজয়	বিসর্জনে
ভাই	ভিতর	ভূভারতে	ভূমি	ভৌমিক

মানুষ	মায়মুখি	মিনিট	মৌসুমী	মোচ
যাদুঘর	যমুনা	যক্ষা	যূথিকা	যৌবন
রুধির	রবিবার	রামায়ণ	রসাল	রমণী
লুকোচুরি	লাঙল	লাম্বর	লাটসাহেব	লেখাপড়া
শাখানদী	শামুক	শাসক	শিশু	শৌখিন
ষটক	ষড়ানন	ষাঁড়ষাঁড়ি	ষাঁড়	ষোড়শী
সাগর	সিংহ	সেনাপতি	সৈকত	সময়
হাঙর	হাতঘড়ি	হিমবাহ	হৈমবতী	হঠাৎ

Table A2 Five instances of each keyword used for classification of handwritten document pages (refer to Chapter 7). In this table KW ID indicates keyword index number

KW #	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
01	উৎসব	উৎসব	উৎসব	উৎসব	উৎসব
02	পূজা	পূজা	পূজা	পূজা	পূজা
03	দুর্গা	দুর্গা	দুর্গা	দুর্গা	দুর্গা
04	বাঙালী	বাঙালী	বাঙালী	বাঙালী	বাঙালী

05	নদী	নদী	নদী	নদী	নদী
06	গঙ্গা	গঙ্গা	গঙ্গা	গঙ্গা	গঙ্গা
07	পদ্মা	পদ্মা	পদ্মা	পদ্মা	পদ্মা
08	যমুনা	যমুনা	যমুনা	যমুনা	যমুনা
09	খেলা	খেলা	খেলা	খেলা	খেলা
10	রান	রান	রান	রান	রান
11	উইকেট	উইকেট	উইকেট	উইকেট	উইকেট
12	শতরান	শতরান	শতরান	শতরান	শতরান
13	মাইক্রোপ্রসেসর	মাইক্রোপ্রসেসর	মাইক্রোপ্রসেসর	মাইক্রোপ্রসেসর	মাইক্রোপ্রসেসর
14	ওয়েবসাইট	ওয়েবসাইট	ওয়েবসাইট	ওয়েব সাইট	ওয়েবসাইট
15	ডাটাবেস	ডাটাবেস	ডাটাবেস	ডাটাবেস	ডাটাবেস
16	প্রকাশনা	প্রকাশনা	প্রকাশনা	প্রকাশনা	প্রকাশনা
17	ইংরেজ	ইংরেজ	ইংরেজ	ইংরেজ	ইংরেজ
18	জালিয়ানওয়ালাবাগ	জালিয়ানওয়ালাবাগ	জালিয়ানওয়ালাবাগ	জালিয়ান ওয়ালাবাগ	জালিয়ানওয়ালাবাগ
19	সেনাপতি	সেনাপতি	সেনাপতি	সেনাপতি	সেনাপতি
20	ডায়ার	ডায়ার	ডায়ার	ডায়ার	ডায়ার

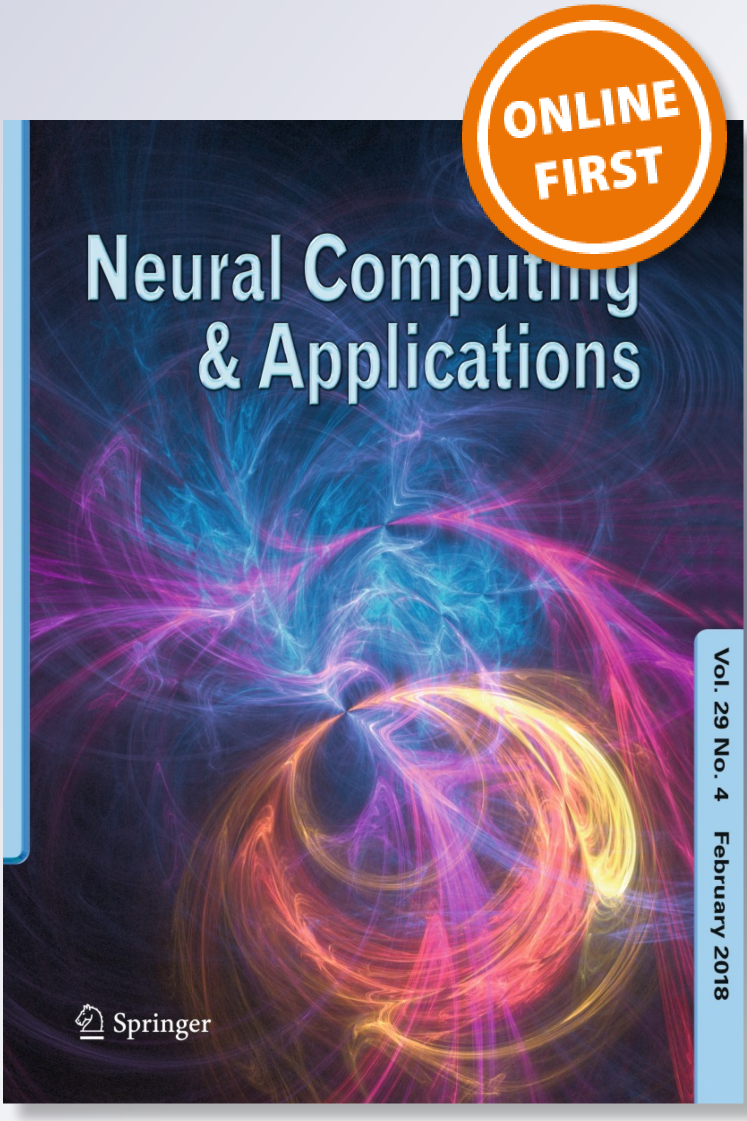
Off-line Bangla handwritten word recognition: a holistic approach

Showmik Bhowmik, Samir Malakar, Ram Sarkar, Subhadip Basu, Mahantapas Kundu & Mita Nasipuri

Neural Computing and Applications

ISSN 0941-0643

Neural Comput & Applic
DOI 10.1007/s00521-018-3389-1



Neural Computing and Applications
https://doi.org/10.1007/s00521-018-3389-1

ORIGINAL ARTICLE

Off-line Bangla handwritten word recognition: a holistic approach

Showmik Bhowmik[1] · Samir Malakar[2] · Ram Sarkar[1] · Subhadip Basu[1] · Mahantapas Kundu[1] · Mita Nasipuri[1]

Received: 9 September 2017 / Accepted: 21 February 2018

Abstract

Due to the cursive nature, segmentation of handwritten Bangla words into characters and also recognition of the same sometimes become a very challenging problem to the researchers. Presence of comparatively large character set along with modifiers, ascendants, descendants, and compound characters makes the segmentation task more complex. As holistic method avoids such character-level segmentation, it is generally useful for the recognition of words written in any such complex scripts. In the present work, a holistic handwritten word recognition method is developed using a feature descriptor, designed by combining different Elliptical, Tetragonal and Vertical pixel density histogram-based features. Recognition process is carried out separately using two classifiers, *namely* multi-layer perceptron (MLP) and support vector machine (SVM). For evaluation of the proposed method, a database of 18,000 handwritten Bangla word images, having 120 word classes, is prepared. The proposed system performs comparatively better with SVM than MLP for the prepared dataset. It has achieved 83.64% accuracy at best case and 79.38% accuracy on an average using fivefold cross-validation. The current method has also outperformed some recently reported holistic word recognition technique tested on the developed dataset. In addition to that the database, prepared in this work, is made freely available to fill the absence of a publicly available standard database for holistic Bangla word recognition.

Keywords Holistic word recognition · Handwritten word · Tetragonal feature · Elliptical feature · Vertical pixel density histogram-based feature · Bangla script

1 Introduction

Automatic recognition of handwritten text is one of the most popular areas of research in the domain of document image processing [1, 2]. The reason of its popularity lies in its wide range of applications in human society which include postal automation [3, 4], bank check processing [5, 6], form processing [7, 8], etc. Major difficulty in recognizing the handwritten text is mainly due to the varying writing styles of individuals. Even the script in which the text is written can pose additional challenges. For example, some Indic scripts like Devanagari and Bangla comprise a considerably larger character set in comparison with Roman/Latin script. Chinese, Japanese and Korean scripts also have large character sets but in these scripts characters appear isolated in the text, whereas Indic scripts are very often written in cursive manner. Therefore, development of a comprehensive and accurate handwritten text recognition system in Indic script is difficult and needs more attention from the researchers [9].

In the literature, plenty of work can be found for the recognition of words written in Arabic [10–12], Chinese/Japanese [13–15] and Roman [16–18] scripts. But in comparison with that very few attempts have been made for the recognition of words written in Bangla script. With more than 200 million speakers, Bangla is the seventh most spoken language in the world [19]. It is also the second most popular official language (out of 23 official languages) in India and the national language of Bangladesh. Besides Bangla language, Bangla script is also used to write other languages like Assamese and Manipuri. Although a significant number of work have been reported

✉ Samir Malakar
samirmalakar@ieee.org

[1] Department of Computer Science and Engineering, Jadavpur University, Kolkata, India

[2] Department of Computer Science, Asutosh College, Kolkata, India

Published online: 03 March 2018

for recognition of handwritten Bangla isolated characters [20, 21] and digits [22], very few pieces of work are there for Bangla handwritten word recognition (HWR) and even, existence of standard databases of handwritten Bangla words is also scarce.

To bridge this research gap, in this work, a novel HWR system of Bangla script is developed. For the evaluation of this system, a first-of-its-kind open-access handwritten Bangla word image database, consisting of 18,000 city name images of 120 different classes, is also prepared, which is another major contribution of this paper. The paper is organized as follows: Sect. 2 reviews the related work. Section 3 presents the description of the proposed work including database preparation, preprocessing and feature extraction. Experimental results are discussed in Sect. 4, and finally, Sect. 5 concludes the paper.

2 Related work

Handwritten text recognition including word recognition is generally performed either in online mode [23, 24] or in off-line mode [25, 26]. In online mode, some digital input devices like tabs or I-pads are used as writing medium instead of paper, and the recognition is performed simultaneously while the writing progresses. But in off-line mode, writing medium is a passive surface such as paper and simultaneous recognition is not possible. In online mode, availability of information like positional information of strokes, direction of strokes and their order [26] makes the recognition process relatively less complex in comparison with text written in off-line mode.

In the literature, two different approaches are followed to handle the problem of off-line HWR, *namely* (1) analytical approach [27, 28] and (2) holistic approach [29]. In analytical approach, a word is initially segmented into sub-units called characters and then each sub-unit is recognized sequentially in order to identify the whole word. Most of the earlier word recognition solutions have been developed based on this approach. The major problem with this approach is finding the proper segmentation points, which becomes more challenging when the handwriting is cursive in nature. On the other hand, holistic word recognition considers a word as a single and indivisible unit and thus extracts information from the whole word to recognize it. In this way, this approach avoids the segmentation issue. According to the authors in [29], holistic approach may succeed even when the writing is too poor for the identification of individual character boundary from the word, but it preserves the overall shape. It is worth mentioning here that use of this approach is restricted to the problems with fixed or limited lexicon. Besides that, this approach can also be used for the reduction in the lexicon in large vocabulary problems [29, 30]. There are many contemporary work available in the literature, where holistic approach has been followed to recognize handwritten words. For example Dasgupta et al. [26] have applied Arnold transform followed by Hough transform on the word images to get distribution of stroke orientation. Based on this information, they have recognized the word images holistically. For evaluation of their work, they have used CENPARMI database of legal amount written in English with 32 different word classes. Malakar et al. [31] have extracted several topological features either from the entire word image or from the hypothetically segmented sub-regions of a word image to recognize it holistically. They have used a database of handwritten Hindi word images having 33 different classes to assess the performance of their method. Tamen et al. [32] have described the word images at feature space using Chebyshev moments and some statistical and contour-based features. For classification, they have used a multi-classifier environment. The method is evaluated using a database of handwritten Arabic words having 21 different word classes, which is basically a subset of IFN/ENIT [33] database.

Similar to other languages, most of the earlier work on Bangla HWR are purely segmentation based [3, 4]. As mentioned previously, the major problem with this approach is finding proper segmentation points. Due to the presence of noise, touching character(s), uneven space among the characters in a word, these techniques may end up with over- or under-segmented characters. To overcome this problem, character-based Hidden Markov Model (HMM) is used in [34, 35]. In these methods, a two-stage recognition scheme is introduced for words written in Bangla. Initially, a given word image is segmented into three zones, viz., upper, lower and middle. The character components in the segmented zones are recognized separately, and the zone recognition results are combined to generate the final word recognition score. The character-based HMM is used to recognize the middle zone. The problem of presegmentation of characters from the word images is solved to some extent by this approach, but it introduces zone-level segmentation which may also become very much error prone for unconstrained handwritten words.

However, in the literature, few attempts can be found toward the development of a holistic Bangla HWR scheme. Some of those have followed a lexicon-based holistic approach [36–38], whereas others have focused on employing feature descriptor for that purpose. In [39], Histogram of Oriented Gradient (HOG) feature descriptor is used for holistic handwritten Bangla word recognition, whereas in [40, 41] Elliptical and convex hull-based features are used, respectively, for the same. Recently, in [42, 43], gradient-based feature descriptor is used for holistic Bangla HWR. In the later one, a memetic algorithm-based feature selection technique is also proposed to enhance the recognition result. However, most of these methods are evaluated on a very small dataset, so the robustness of these methods cannot be assured.

3 Proposed work

In the present work, a novel shape-based feature descriptor called *Tetragonal feature* is used along with *Vertical pixel density histogram-based feature* and our previously introduced *Elliptical feature* [40] to capture the shape or geometric nature of a handwritten word image in order to recognize it in a holistic manner. Recognition process separately employs two well-known classifiers, viz., MLP [44] and SVM [45]. Each step of the proposed work is elaborated in the subsequent sections.

3.1 Database description

One of the main reasons for the slow progress of research on HWR for regional languages is the unavailability of suitable databases. As holistic HWR systems are generally developed for specific applications, these mainly deal with limited lexicons. To carry out training and evaluation of such systems, some handwritten word databases like IFN/ENIT (in Arabic script) [33] and CENPERMI (in Roman script) [46] are made available to the research community either on-demand basis or through subscription charges. However, no such standard database comprising Bangla handwritten words is available in the literature, which can be used to train and evaluate a holistic HWR system.

To address this need, in the present work, a database of 18,000 handwritten Bangla word images, written by around 300 different native writers belonging to different age groups, sex and educational backgrounds, is prepared. Word images in the current database contain the names of 120 different cities in West Bengal, a state of India, with 150 samples for each city name. The city names listed in Table 1 are chosen based on their population and the literacy rate. The present database includes almost all urban regions of West Bengal. Handwritten words in this work were collected in A4 size datasheets containing a grid of 10 rows and 3–5 columns depending upon the word length. The writers were asked to write each city name inside the rectangular boxes only. Such datasheets are then scanned using a flat-bed scanner with a resolution of 300 dpi and are stored as 24 bit BMP image file. From each such image of the datasheets, handwritten word images are cropped automatically. More detail description regarding the height, width, number of ascendants and descendants of the word images present in each word class is given in a file named "ReadMeCMATERdb2.1.2.pdf" along with the database. This information will help the researchers to get an idea about how much intra- and/or inter-class shape and size variation their method may tolerate. More information regarding the samples of the present dataset is given in Tables 2 and 3.

Database nomenclature is an important task for future reference by the research community. The database prepared in this work is given the name "CMATERdb2.1.2" based on our predefined naming convention [47]. One of the main contributions of the current work is that the database is made freely available to the entire research community of the said domain. To access the database refer to the link given in [48].

3.2 Preprocessing

Before feature extraction, all the word images are smoothed using disk filter with radius 4 to remove noise and then binarized using global histogram-based Otsu method [49]. After that, two morphological operations, viz., erosion and dilation [50] are performed on the binarized word images using a 3×3 structuring element to remove the isolated foreground[1] pixels as well as to smooth the contours of the object regions. These operations also help in filling up the holes, created during binarization, within the object regions.

3.3 Feature extraction

The proposed feature descriptor is designed with the intension of capturing shape information of a given word image. In the field of object recognition, shape information of an object is heavily relied upon for the discrimination purpose [51]. But most of these shape descriptors are computationally expensive and require extra memory space for further processing. Keeping that in mind, in the present work, shape information of a handwritten word is obtained in terms of *local distribution of foreground pixels* along with *presence* and *position of ascendant(s) and descendent (s)*. For that, from a given word image several *Elliptical* [40], *Tetragonal* and *Vertical pixel density histogram-based* features are extracted. The proposed descriptor is not only simple to compute but also effective in recognition of large number of handwritten word classes which are truly complex in nature.

[1] By foreground pixels in a word image, we mean object pixels only and the rest of the pixels are considered as representing the background. In this paper, we have followed this convention.

Table 1 City names, used in the database, written in Bangla along with their class numbers

Class #	Name	Class #	Name	Class #	Name	Class #	Name
1	আলিপুর	31	বনগাঁ	61	এগরা	91	কোন্নগর
2	বালুরঘাট	32	বানপুর	62	ফুলিয়া	92	কুলটি
3	বাঁকুড়া	33	বাঁশবেড়িয়া	63	গঙ্গারামপুর	93	লালগোলা
4	বারাসাত	34	বাঁশড়া	64	গারুলিয়া	94	মধ্যমগ্রাম
5	বর্ধমান	35	ব্যারাকপুর	65	গায়েশপুর	95	মহেশতলা
6	বহরমপুর	36	বরানগর	66	ঘাটাল	96	মেমারি
7	চুঁচুড়া	37	বারুইপুর	67	গোবরডাঙ্গা	97	মুর্শিদাবাদ
8	কোচবিহার	38	বসিরহাট	68	গুসকরা	98	নবদ্বীপ
9	দার্জিলিং	39	বেলডাঙা	69	হাবড়া	99	নৈহাটী
10	ইংলিশবাজার	40	বেলঘরিয়া	70	হলদিয়া	100	নলহাটি
11	হাওড়া	41	ভাটপাড়া	71	হালিশহর	101	ঔরঙ্গাবাদ
12	জলপাইগুড়ি	42	বিরাটি	72	হিজুলি	102	পানিহাটি
13	কলকাতা	43	বীরনাগার	73	ইছাপুর	103	পলাশী
14	কৃষ্ণনগর	44	বিষ্ণুপুর	74	ইসলামপুর	104	রামপুরহাট
15	মালদা	45	বোলপুর	75	জামুরিয়া	105	রানাঘাট
16	মেদিনীপুর	46	বজবজ	76	জঙ্গীপুর	106	রিষড়া
17	পুরুলিয়া	47	চাকদহ	77	ঝাড়গ্রাম	107	সাঁইথিয়া
18	রায়গঞ্জ	48	চাঁপদানি	78	কাজোড়া	108	শান্তিপুর
19	সিউড়ি	49	চন্দননগর	79	কালনা	109	শিবপুর
20	তমলুক	50	চিত্তরঞ্জন	80	কল্যাণী	110	শিলিগুড়ি
21	আগারপাড়া	51	দাঁইহাট	81	কামারহাটী	111	শ্যামনগর
22	আজিমগঞ্জ	52	ডালখোলা	82	কাঁচরাপাড়া	112	সোদপুর
23	আরামবাগ	53	ডানকুনি	83	কান্দি	113	সোনামুখী
24	আসানসোল	54	ধুলিয়ান	84	কাঁকশা	114	সোনারপুর
25	অশোকনগর	55	ধূপগুড়ি	85	কাঁথি	115	শ্রীরামপুর
26	বাদকুল্লা	56	দিনহাটা	86	করিমপুর	116	তারকেশ্বর
27	বৈদ্যবাটী	57	ডোমকল	87	কাটোয়া	117	টিটাগড়
28	বলরামপুর	58	দুবরাজপুর	88	খড়্গপুর	118	উখরা
29	বালি	59	দমদম	89	খড়দহ	119	উলুবেড়িয়া
30	ব্যান্ডেল	60	দুর্গাপুর	90	কোলাঘাট	120	উত্তরপাড়া

3.3.1 Elliptical features

An ellipse is a curved line forming a closed loop, where the sum of the distances from two points (foci) to every other points on the curve is constant. Ellipse can also be defined parametrically as well as nonparametrically.

The parametric definition of an ellipse is,

$$X = C_x + a\cos(t) \tag{1}$$

$$Y = C_y + b\sin(t) \tag{2}$$

where $\left(C_x, C_y\right)$ is the coordinate of the center of ellipse, a is a constant representing the length of its radius along X-axis., b is a constant representing the length of its radius along Y-axis and t is a parameter such that $0 < t < 2\pi$. Equation of an ellipse in nonparametric or Cartesian form is defined as,

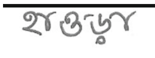
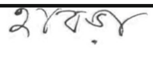
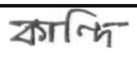
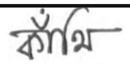
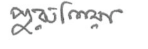
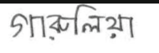
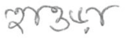
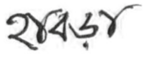
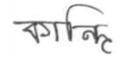
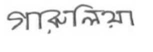
$$\frac{(X - C_x)^2}{a^2} + \frac{\left(Y - C_y\right)^2}{b^2} = 1 \tag{3}$$

The nonparametric equation of the ellipse is used here for feature computation.

3.3.1.1 Fitting of ellipses on the word image In this work, some hypothetical concentric ellipses are first fitted on a word image. To decide the center of these ellipses, the center of gravity of a word image is computed as,

$$C_x = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{4}$$

Table 2 Sample word classes having high shape similarity

হাওড়া	হাবড়া	কান্দি	কাঁথি	পুরুলিয়া	গারুলিয়া
					
					

$$C_y = \frac{1}{N}\sum_{i=1}^{N} y_i \tag{5}$$

where (x_i, y_i) is the coordinate of the ith foreground pixel and N represents the total number of foreground pixels. The constants a and b are computed as follows,

$$a = \min\{(C_x - X_L), (X_R - C_x)\} \tag{6}$$

$$b = \min\{(C_y - Y_B), (Y_T - C_y)\} \tag{7}$$

where (X_L, Y_B) and (X_R, Y_T) are the coordinates of the top-left and bottom-right corners of the minimum bounding box of a word image, respectively (see Fig. 1).

While fitting a hypothetical ellipse within a boundary of a word image, our objective is to take most of the foreground pixels inside the ellipse. For that reason, the center of gravity of a word image is considered as the center of the ellipse, as it always resides in the dense foreground region of the image. Now as the center of gravity for a word image may always not be the geometric center of its bounding box, choosing maximum value instead of minimum in Eqs. (6) and (7) may cause a situation, when a portion of the hypothetical ellipse may lie outside the image bounding box. That empty portion of the ellipse will not provide any useful shape information of that word.

3.3.1.2 Computation of feature values Elliptical features over an entire word image are computed in following two ways:

Concentric ellipses Initially, three hypothetical concentric ellipses are considered on a given word image (see Fig. 2). Then from those three ellipses four Elliptical features are computed. Let us assume that the outermost ellipse is marked as first, the next ellipse is marked as second and the innermost ellipse is marked as third. Radii of the ellipses are computed as follows,

$$a_i = \frac{a}{2^{(i-1)}} \tag{8}$$

$$b_i = \frac{b}{2^{(i-1)}} \tag{9}$$

where $i = 1, 2, 3$.

Here a_i and b_i represent the parameters a and b of ith ellipse. From each of the ellipses, following *four* feature values representing the *number of foreground pixels* over different local regions of the word image are computed. The local regions considered here are: (i) outside the first ellipse (but within the minimum bounding box of the word image), (ii) *between first and second ellipses*, (iii) *between second and third ellipses* and (iv) *inside the third ellipse*. As different words have different shapes, there may be noticeable differences in the foreground pixel distribution at various local regions created by those concentric ellipses. Our endeavor is to use these differences which eventually help classifier in recognizing the word images.

Outermost ellipse After computation of the feature values from three hypothetical concentric ellipses, outermost ellipse is considered for computation of another *five* feature values. These are (i) *number of foreground pixels on boundary of the ellipse*, (ii) *number of foreground pixels along the axis parallel to X-axis*, (iii) *number of foreground pixels along the axis parallel to Y-axis*, (iv) *ratio of foreground pixels and background pixels inside the ellipse and* (v) *ratio of foreground pixels inside and outside of the ellipse* (not exceeding the minimum bounding box of the word).

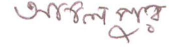
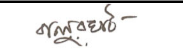
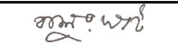
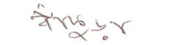
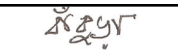
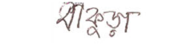
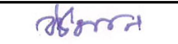
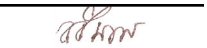
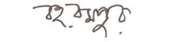
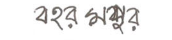
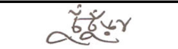
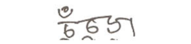
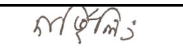
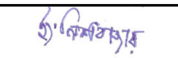
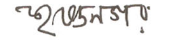
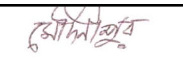
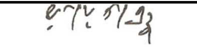
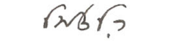
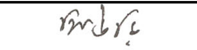
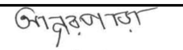
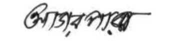
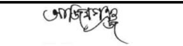
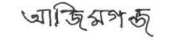
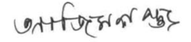
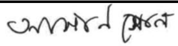
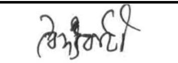
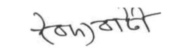
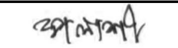
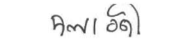
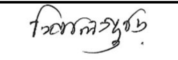
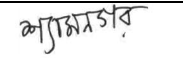
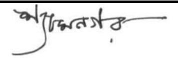
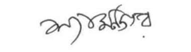
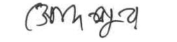
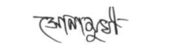
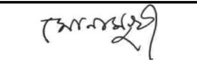
To acquire more information about the shape of a word image, region inside the outermost ellipse is divided into four sub-regions depending on the center and two foci points of the ellipse as shown in Fig. 3. The foci points $(C_x + \delta, C_y)$ and $(C_x - \delta, C_y)$ for the outermost ellipse are computed using the foci distance δ computed as follows,

$$\delta = \sqrt{|a^2 - b^2|} \tag{10}$$

Finally, *the number of foreground pixels* inside each of these four sub-regions is computed. Thus, in total, *nine* feature values are computed from the outermost ellipse.

From Eq. 10, it is clear that computation of foci points requires the values of a and b, which are directly related to width and height of a given word image, respectively. Widths of the word images vary from one class to another depending on the number of characters in a word and the writing styles of various writers. Thus the value of a would also vary over the different word classes. Similarly, heights

Table 3 Examples of some cursive word samples reflecting the shape variation present in the prepared database

Sl. #	Word Class	Sample 1	Sample 2	Sample 3
1	আলিপুর			
2	বালুরঘাট			
3	বাঁকুড়া			
4	বারাসাত			
5	বর্ধমান			
6	বহরমপুর			
7	চুঁচুড়া			
8	দার্জিলিং			
9	ইংলিশবাজার			
10	জলপাইগুড়ি			
11	কলকাতা			
12	কৃষ্ণনগর			
13	মেদিনীপুর			
14	পুরুলিয়া			
15	রায়গঞ্জ			
16	সিউড়ি			
17	আগারপাড়া			
18	আজিমগঞ্জ			
19	আসানসোল			
20	অশোকনগর			
21	বৈদ্যবাটী			
22	পলাশী			
23	শান্তিপুর			
24	শিলিগুড়ি			
25	শ্যামনগর			
26	সোদপুর			
27	সোনামুখী			

of the word images differ from one word class to another because of the same logic and also due to the presence of ascendants and descendants. So, the value of *b* also varies for different word types. As the values of these two constants get changed over the word classes, the location of the foci points would also vary accordingly. That means the pixel distribution at various local regions created by those foci points would also differ significantly from one word class to another, which helps during classification.

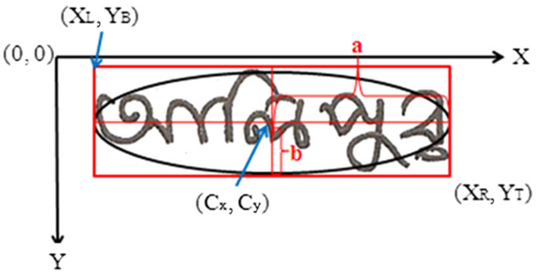


Fig. 1 Illustration of outermost hypothetical ellipse fitted on a sample word image and the related parameters. Here (C_x, C_y) represents the center of gravity of the word image and the center of the hypothetical ellipse. (X_L, Y_B) and (X_R, Y_T) represent the coordinates of the upper-left and lower-right corner of the bounding box of the word image, respectively. a and b are the length of radius along X and Y-axes, respectively

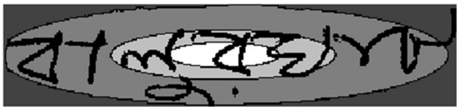


Fig. 2 Three concentric ellipses fit on a word image

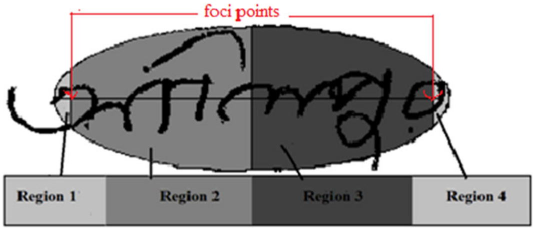


Fig. 3 Illustration of four sub-regions generated based on the foci points by considering the outermost ellipse

Therefore, from each word image, 13 (i.e., 4 from concentric ellipses and 9 from outermost Ellipse) global feature values are extracted. To get additional local information, a word image is further divided into four small sub-parts depending on the center of the ellipse and then from each sub-part, same set of feature values, as mentioned earlier, is computed. Therefore, in total 65 (i.e., 5 × 13) Elliptical features are computed from a particular word image (see Fig. 4). All these feature values are suitably normalized before feeding them to classifier.

3.3.2 Tetragonal features

In Euclidian plane geometry, a tetragon can be defined as a polygon with four sides and four corners. It can be concave or convex as well. In the present method, two convex tetragons are considered, the first one, whose two corner points (upper and lower) vary from one word type to another, is named as *Flexible Tetragon* (see Fig. 5a) and the second one, whose all four corner points are fixed or rigid, is named as *Rigid Tetragon* (see Fig. 5b). The main motive behind using *Flexible Tetragon* is to capture the geometrical dissimilarity among the words belonging to different classes, whereas *Rigid Tetragon* is used to estimate the variation present in the local pixel distribution of a word images, inside a fixed shaped region.

3.3.2.1 Flexible Tetragon Before extraction of the Tetragonal features, a Flexible Tetragon for a given word image is drawn hypothetically. Four vertices p_1, p_2, p_3 and p_4 of the hypothetical tetragon are estimated as follows. Point p_1 is the location of first foreground pixel from left on the upper boundary of the bounding box of a word image. Similarly p_3 corresponds to the first foreground pixel from left on the lower boundary, whereas, p_2 and p_4 are mid-points on the left and right sides of the bounding box,

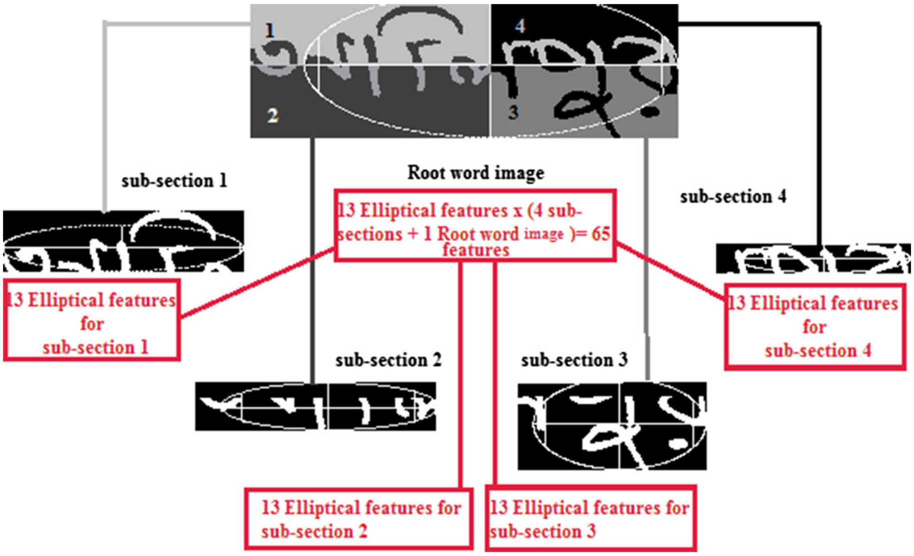

1
4
2
3
sub-section 1
Root word image
13 Elliptical features x (4 sub-sections + 1 Root word image)= 65 features
sub-section 4
13 Elliptical features for sub-section 1
13 Elliptical features for sub-section 4
sub-section 2
sub-section 3
13 Elliptical features for sub-section 2
13 Elliptical features for sub-section 3

Fig. 4 Illustration of local elliptical feature extraction process from a given word image

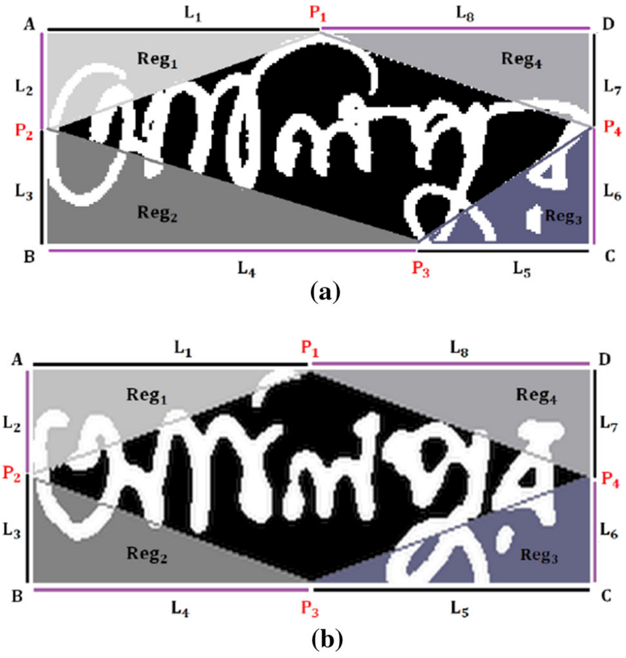


Fig. 5 Illustration of hypothetical tetragons, drawn on a given word image. Here P_i represents the ith corner of a hypothetical tetragon. L_is are the sides representing bases and heights of the right-angled triangles created by a hypothetical tetragon with the word boundary. Reg_i is the region, within the word boundary covered by the ith triangle. **a** Flexible Tetragon. **b** Rigid Tetragon

respectively. After the estimation of those four points, four hypothetical lines are constructed from p_1 to p_2, p_2 to p_3, p_3 to p_4 and p_4 to p_1. Figure 5a shows a hypothetical Flexible Tetragon on a given word image.

Feature computation using Flexible Tetragon After designing the hypothetical convex tetragon, two types of features are computed, *namely* (1) *Pixel distribution* and (2) *Geometric*.

Pixel distribution features A hypothetical tetragon divides the minimum bounding box of a word image into five sub-regions. This includes four sub-regions, created outside the tetragon named as Reg_1, Reg_2, Reg_3, Reg_4 and one sub-region inside the tetragon (see Fig. 5a). From this tetragon, following six Pixel distribution features are computed: (i) *Number of foreground pixels in Reg₁*, (ii) *Number of foreground pixels in Reg₂*, (iii) *Number of foreground pixels in Reg₃*,(iv) *Number of foreground pixels in Reg₄*, (v) *Number of foreground pixels inside the tetragon* and (vi) *Number of foreground pixels along the boundary of the tetragon*. All these feature values are normalized by the total number of foreground pixels in a given word image.

Geometric features Following two types of geometric features are considered for the present work: (i) *Area-based* and (ii) *Angle-based*.

(i) Area-based features

In this work, eight Area-based features are computed. Out of that four feature values are computed by estimating areas of the four outer sub-regions created by the tetragon and the minimum bounding box of a word image, which are basically right-angled triangles (see Fig. 5a) and rest of the feature values are computed by considering the ratio of the area of each outer region and the area of the inside region.

Area of a right-angled triangle can be computed by the following equation,

$$\mathrm{Area} = \frac{1}{2} \times \mathrm{base} \times \mathrm{height} \tag{11}$$

In this work, area of a sub-region is computed as follows,

$$\mathrm{Area}(\mathrm{Reg}_i) = \frac{1}{2} \times L_{2\times i} \times L_{((2\times i)-1)} \tag{12}$$

where $i = 1, 2, 3, 4$. Here L_is are the sides representing bases and heights of the right-angled triangles. Area of the inside region is computed as follows,

$$\text{Area (inside region of a tetragon)}$$

$$= (H_{\mathrm{BB}} \times W_{\mathrm{BB}}) - \sum_{i=1}^{4} \mathrm{Area}(\mathrm{Reg}_i) \tag{13}$$

where H_{BB} and W_{BB} represent the height and width of a word bounding box.

First four Area-based feature values are normalized by the area of the minimum bounding box of the word.

(ii) Angle-based features

Here also, four different feature values are computed by estimating the angles created at the four corners of a hypothetical tetragon (see Fig. 5a). These angles are named as $\angle p1p2p3$, $\angle p2p3p4$, $\angle p3p4p1$ and $\angle p4p1p2$ which are computed as follows,

$$\angle p1p2p3 = 180^\circ - (\angle Ap2p1 + \angle Bp2p3) \tag{14}$$

$$\angle p2p3p4 = 180^\circ - (\angle p2p3B + \angle p4p3C) \tag{15}$$

$$\angle p3p4p1 = 180^\circ - (\angle p3p4C + \angle p1p4D) \tag{16}$$

$$\angle p4p1p2 = 180^\circ - (\angle p4p1D + \angle p2p1A) \tag{17}$$

To solve Eq. 14, first, the values of $\angle Ap2p1$ and $\angle Bp2p3$ are computed as follows,

$$\angle Ap2p1 = \arctan\left(\frac{L_1}{L_2}\right) \tag{18}$$

$$\angle Bp2p3 = \arctan\left(\frac{L_4}{L_3}\right) \tag{19}$$

In a similar way, Eqs. 15, 16 and 17 are computed. These feature values are normalized by dividing them with 360, as the sum of four interior angles of a tetragon is 360°.

Therefore, from each word image 18 (i.e., 6 + 12) features are extracted globally. To get local information, a given word image is further divided into four sub-regions depending on the center of gravity of the image and from each sub-region same set of features are computed. Thus, in total 90 (i.e., 18 × (4 local + 1 global)) features are estimated from a given word image. Figure 6 shows how the geometrical nature of the Flexible Tetragon gets changed over different classes of words.

As presence and position of the ascendants and descendants in a word play a very significant role in defining its shape, the above-mentioned geometrical properties of the hypothetical Flexible Tetragon reflect the shape information almost accurately. In addition to that they also vary over words from different word classes (see Fig. 6).

3.3.2.2 Rigid Tetragon For a given word image, a hypothetical Rigid Tetragon is computed by estimating its four corner points $p1$, $p2$, $p3$ and $p4$ as the midpoints of upper, leftmost, lower and rightmost edges of the word bounding box, respectively (see Fig. 5b).

Feature computation using Rigid Tetragon Rigid Tetragon helps in utilizing the information variation in terms of pixel distribution in different local regions of the word images belonging to different word classes. In this feature computation, four feature values are extracted from each outer region such as (1) *number of foreground pixels* (normalized by total number of foreground pixels in the word image), (2) *ratio between the number of foreground pixels and background pixels*, (3) *ratio between the number of foreground pixels present in the considered region and the number of foreground pixel present inside the rigid tetragon*, (4) *ratio between the number of foreground pixels and the area of the considered region*. From the inner region of the rigid tetragon, two more feature values such as (1) *number of foreground pixels* (normalized by total number of foreground pixels in the word image) and (2) *ratio between the number of foreground pixels and background pixels* are computed. Finally, *number of foreground pixels along boundary of the tetragon* (normalized by total number of foreground pixels in the word image) is also estimated.

Therefore, from a single image initially, 19 (i.e., (4 × 4) + 2 + 1) feature values are extracted globally. After that, to get local information, the word image is further divided into four sub-images depending on the center of gravity of the image and from each sub-image same set of Rigid Tetragonal features are computed. Thus, in total 95 (i.e., 19 × (4 local + 1 global)) feature values are estimated. That means from a given word image all total 185 (90 feature values using Flexible Tetragon + 95 feature values using Rigid Tetragon) Tetragonal features are computed. Figure 7 shows the Tetragonal features computed from a single sub-image.

Tetragonal features estimate the true shape structure of a given word image not only by considering the geometrical properties of a hypothetically drawn tetragon over it but

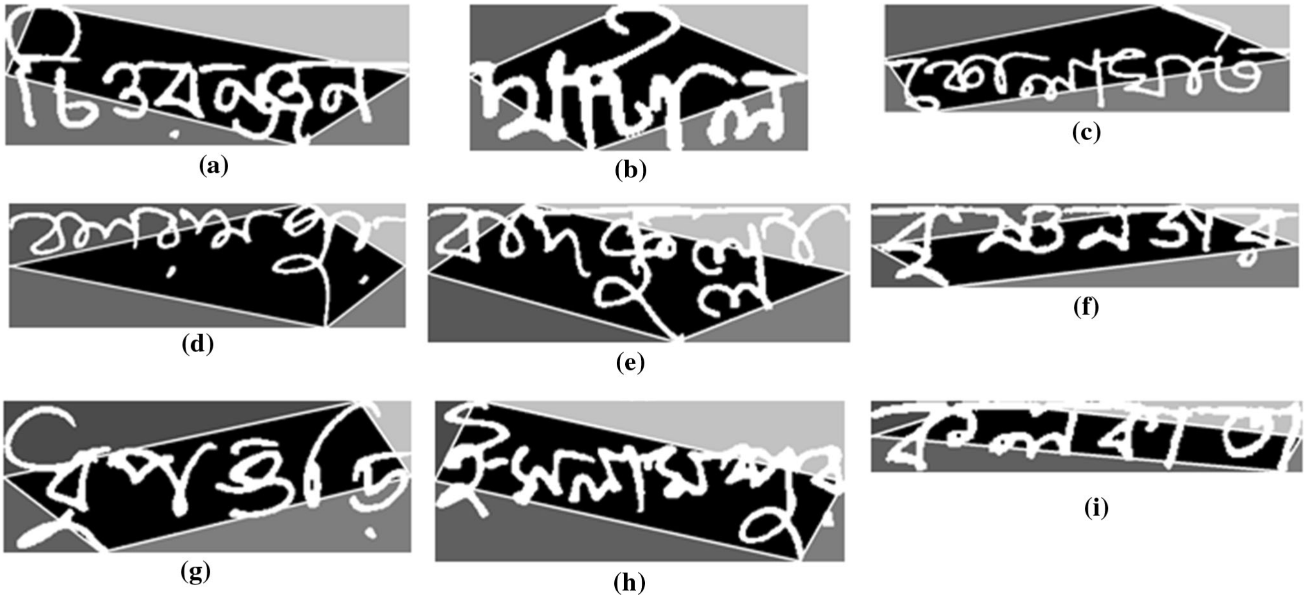
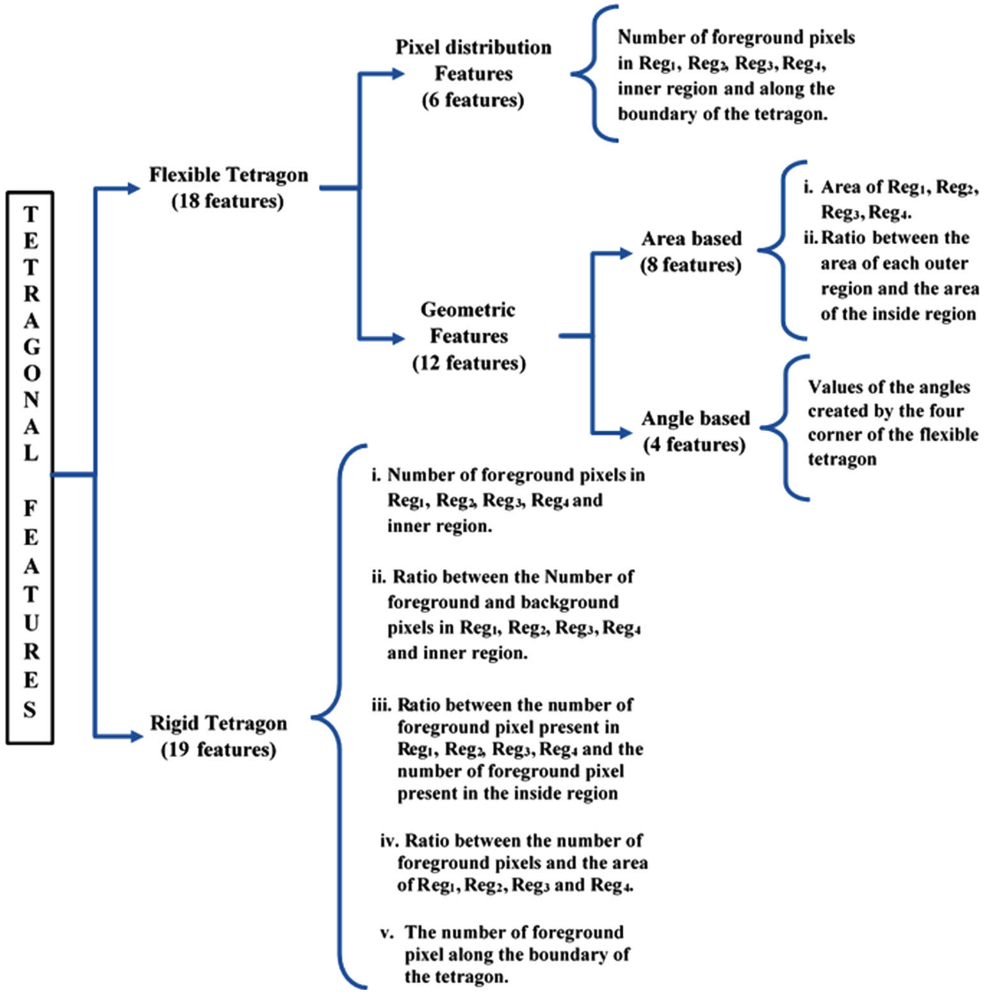


Fig. 6 Flexible Tetragons around the word images: **a**–**c** ascendant in left, middle and right respectively, **d**–**f** descendant in right, middle and left respectively, **g**, **h** both ascendant and descendant, **i** no ascendant and descendant

Fig. 7 Hierarchical description of different types of tetragonal features used in our current work for word classification


TETRAGONAL FEATURES
Flexible Tetragon (18 features)
Pixel distribution Features (6 features)
Number of foreground pixels in Reg₁, Reg₂, Reg₃, Reg₄, inner region and along the boundary of the tetragon.
Geometric Features (12 features)
Area based (8 features)
i. Area of Reg₁, Reg₂, Reg₃, Reg₄.
ii. Ratio between the area of each outer region and the area of the inside region
Angle based (4 features)
Values of the angles created by the four corner of the flexible tetragon
Rigid Tetragon (19 features)
i. Number of foreground pixels in Reg₁, Reg₂, Reg₃, Reg₄ and inner region.
ii. Ratio between the Number of foreground and background pixels in Reg₁, Reg₂, Reg₃, Reg₄ and inner region.
iii. Ratio between the number of foreground pixel present in Reg₁, Reg₂, Reg₃, Reg₄ and the number of foreground pixel present in the inside region
iv. Ratio between the number of foreground pixels and the area of Reg₁, Reg₂, Reg₃ and Reg₄.
v. The number of foreground pixel along the boundary of the tetragon.

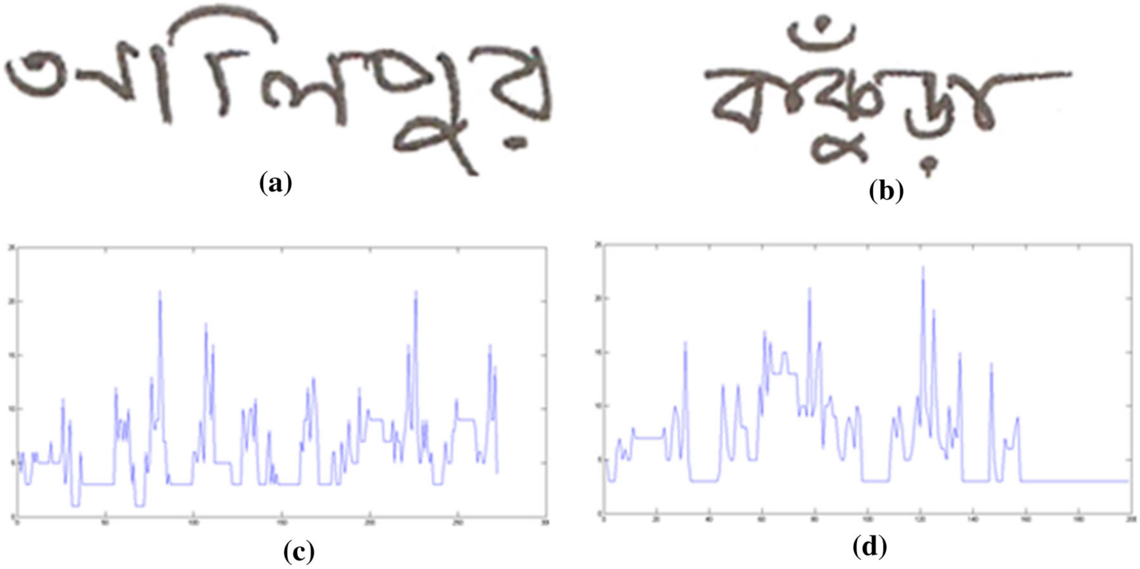


Fig. 8 Vertical pixel density histogram of (**a**, **b**) are shown in (**c**, **d**) respectively

Table 4 Threefold cross-validation result using SVM

Fold #	Number of training samples	Number of test samples	Accuracy in training data (in %)	Accuracy in test data (in %)
1	12,000	6000	98.26	76.18
2			98.05	**80.72**
3			**98.37**	73.18

Bold values indicate the best score

also by estimating the local pixel distribution at various regions created by these tetragons.

3.3.3 Vertical pixel density histogram-based feature

To compute these feature values, vertical pixel density histogram of a given word image is estimated by counting the number of data pixels present at each column of the word image. Figure 8 presents the vertical pixel density histogram of two handwritten Bangla words. After computing the vertical pixel density histogram, *the number of valleys* and *the number of peaks* are computed as two feature values. As in Bangla, words are generally written from left to right direction, vertical pixel density histogram of a word image will be more useful to capture the shape information in comparison with the horizontal pixel density histogram. This is why, in the present work, the vertical pixel density histogram of the word images is considered.

Table 5 Threefold cross-validation result using MLP

Fold #	Number of training samples	Number of test samples	Accuracy in training data (in %)	Accuracy in test data (in %)
1	12,000	6000	96.92	75.33
2			96.39	**79.87**
3			**96.92**	72.67

Bold values indicate the best score

Table 6 Fivefold cross-validation result using SVM

Fold #	Number of training samples	Number of test samples	Accuracy in training data (in %)	Accuracy in test data (in %)
1	14,400	3600	97.69	79.44
2			97.79	77.33
3			97.73	**83.64**
4			**98.08**	79.3
5			97.86	77.19

Bold values indicate the best score

Table 7 Fivefold cross-validation result using MLP

Fold #	Number of training samples	Number of test samples	Accuracy in training data (in %)	Accuracy in test data (in %)
1	14,400	3600	96.23	77.89
2			96.60	75.50
3			96.32	**81.72**
4			**96.65**	77.75
5			96.58	73.75

Bold values indicate the best score

4 Experimental evaluation

Proposed HWR method is implemented on a machine with Intel®Core(TM)i3-5010U@2.10 GHz as the CPU with 4 GB RAM. The current method is evaluated with a dataset comprising 120 handwritten city names of West Bengal (a state of India). Here the classification is carried out separately using two well-known classifiers MLP and SVM. For that purpose, a machine learning tool called WEKA is used [52]. All the steps of the current experiments along with error case analysis are described in Sect. 4.1 and finally in Sect. 4.2, comparison of the current work with *state-of-the-art* feature descriptors as well as methods is presented.

4.1 Experimental results and error analysis

The present experiment is carried out by following fivefold and threefold cross-validation schemes using both MLP and SVM classifiers. MLP classifier in this experiment has 185 neurons in its hidden layer (only one hidden layer is considered here) and trained with learning rate (η) = 0.3 and momentum term (α) = 0.2 for each fold using 1000 iterations. For SVM classifier, a polynomial kernel is used. All these parameter values are chosen experimentally. In fivefold cross-validation, 14,400 images are used for training the model and rest 3600 images are used for testing, whereas, 12,000 and 6000 word images are used for training and testing, respectively, for each fold in threefold cross-validation. Detailed results using MLP and SVM in threefold cross-validation are given in Tables 4 and 5, respectively. Outcomes of the fivefold cross-validation scheme using MLP and SVM are given in Tables 6 and 7, respectively.

4.1.1 System performance with increasing number of word classes

In the present experiment, performance of the proposed system is also observed when number of word classes increased gradually. For this purpose, entire experiment is carried out using threefold and fivefold cross-validation

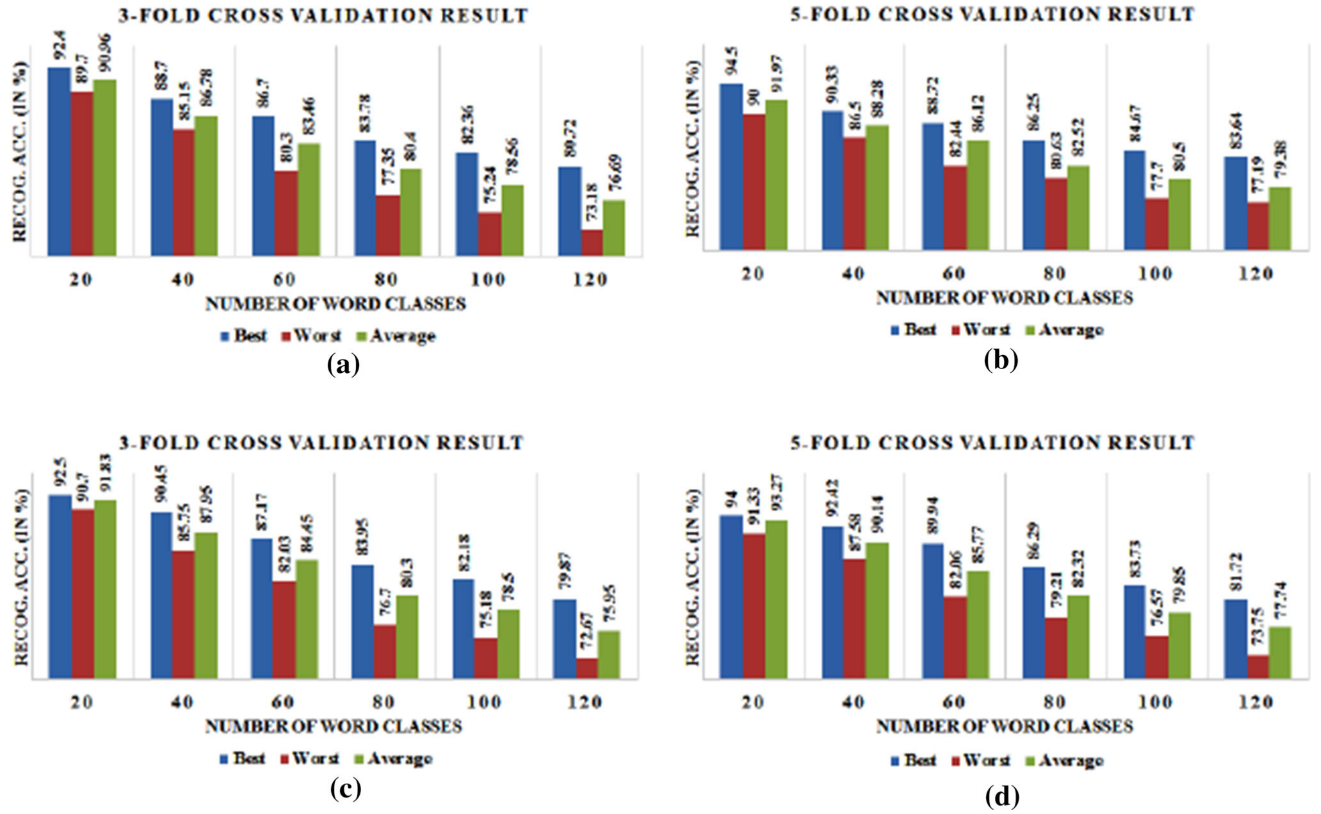


Fig. 9 Performance of the system for different number of word classes using SVM (**a**, **b**) and MLP (**c**, **d**)

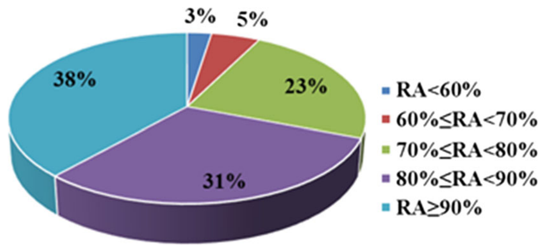


Fig. 10 Number of word classes (in percentage) having recognition accuracy (RA) within a given range

schemes with MLP and SVM classifiers following the previously specified setup for 20, 40, 60, 80, 100 and 120 word classes. Results of these experiments are shown in Fig. 9.

From the above experiments it is observed that in entire dataset (of 120 classes), SVM performs well in comparison with MLP. Thus rest of the discussions are presented based on the results obtained using SVM with fivefold cross-validation.

4.1.2 Error case analysis

The proposed system has achieved impressive results for most of the word classes except a few. A detailed analysis of the result is given in Figs. 10 and 11.

Our analysis revels that the probable reasons behind these misclassifications could be (1) *spelling disparity*, (2) *complex shape* and (3) *similar shaped words from different classes.*

1. Spelling disparity

The disparity in spellings within the same word class causes some alarming shape varieties. As the present work estimates some shape-based features, this variation of shapes due to wrong spellings surely misleads the classifier to identify the true class of the word samples. Figure 12 illustrates such spelling disparities found in our database.

2. Complex shape

Large skew of the word images and complex shape angle can be another two reasons for such misclassifications observed in the current work (see Fig. 13). Since during the feature extraction, the main focus is given on various local regions of any word image, the expected local feature values of the samples belonging to the same word class, as shown in Fig. 13, would differ significantly due to the above reasons.

3. Similar shaped words from different classes

Along with various geometrical features, some pixel count features are also estimated from the word images in

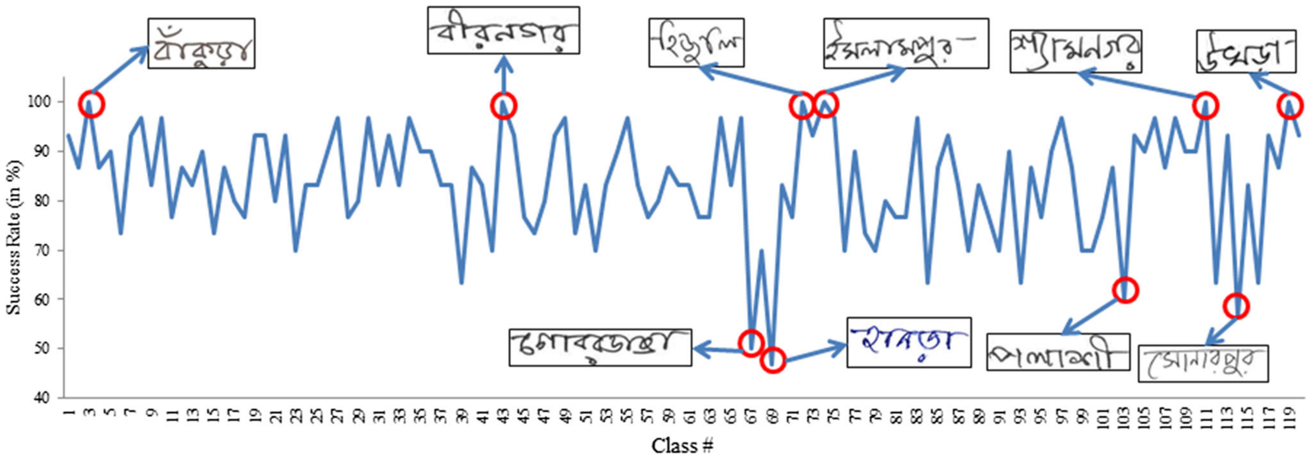


Fig. 11 Class-wise recognition performance is shown with sample word images from classes having top-5 and worst-4 recognition accuracy

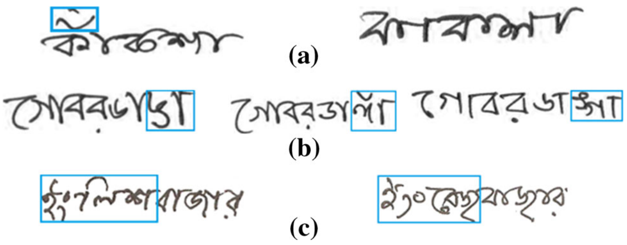


Fig. 12 Illustration of spelling disparities **a**, **b** show disparity caused by miss-spelling, **c** is basically alternative spelling

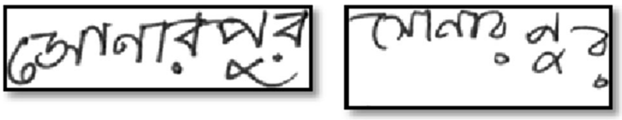


Fig. 13 Illustration of words with skew variation

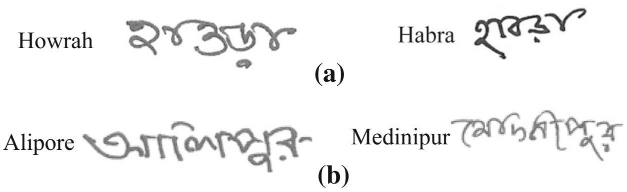


Fig. 14 Some of the word classes having high shape similarity

this work. Thus, in few cases, it is observed that word images with high shape resemblance end up with almost same feature values, which also results in the misclassification. Figure 14 shows some most confusing word classes.

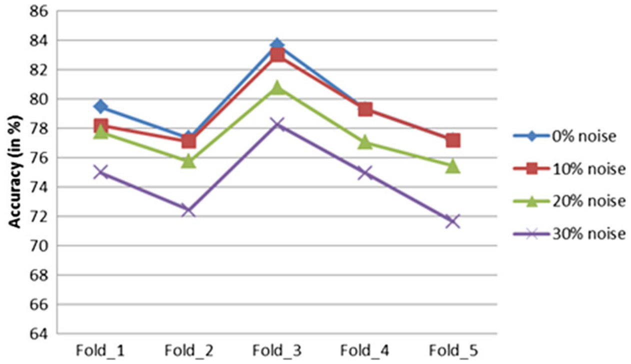


Fig. 16 Fivefold cross-validation results of the proposed system using SVM for data with different level of embedded Gaussian noise

4.1.3 Experiment with noisy data

Performance of the proposed system is also observed while dealing with noisy data. For that purpose, we have synthetically created noisy data by adding Gaussian noise with different levels to the word images (see Fig. 15). Due to the addition of 10%, 20% and 30% noise, a reduction of 0.43%, 2.04% and 4.93% in average recognition rate, respectively, is observed in each fold (see Fig. 16).

4.2 Performance comparison

Recognition performances of each category of elementary feature descriptors, described in Sect. 3.3, are evaluated and given in Table 8. Next, the performance of the

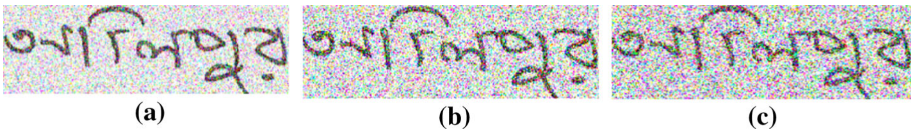


Fig. 15 Sample of handwritten Bangla word image with 10%, 20% and 30% embedded Gaussian noise, respectively

Table 8 Performance comparison of different feature descriptors used in this work on CMATERdb2.1.2

Feature descriptors	Feature dimension	Accuracy (in %)		
		Best	Worst	Avg
Elliptical	65	59.97	53.56	56.5
Tetragonal	185	76.42	70	72.74
Elliptical and tetragonal	250 (65 + 185)	82.61	74.61	77.96
Elliptical, tetragonal and vertical pixel density histogram-based features (Proposed)	**252 (250 + 2)**	**83.64**	**77.19**	**79.38**

Bold values indicate the best score

Table 9 Performance comparison of proposed feature descriptor with the *state-of-the-art* feature descriptors on CMATERdb2.1.2

Feature descriptors	Feature dimension	Avg. feature computation time (in s)	Accuracy (in %)		
			Best	Worst	Avg
LGH [34]	786	0.7199	83.4	76.10	78.55
PHOG [35]	672	0.1721	78.42	73.47	75.18
G-PHOG [53]	720	0.1759	72.53	67.14	78.93
Proposed descriptor	**252**	**0.1158**	**83.64**	**77.19**	**79.38**

Bold values indicate the best score

Table 10 Performance comparison of proposed method with some of the *state-of-the-art* holistic HWR methods on CMATERdb2.1.2

Method with year of publication	Maximum accuracy	Minimum accuracy	Average accuracy	Standard deviation
Bhowmik et al. [40], 2014	58.85	52.82	55.97	2.9
Dasgupta et al. [26], 2016	76.69	68.56	72.71	2.81
Malakar et al. [31], 2017	73.03	67.42	69.72	2.099
Barua et al. [42], 2017	83.06	75.75	79.06	2.72
Proposed method	**83.64**	**77.19**	**79.38**	**2.33**

Bold values indicate the best score

combined feature descriptor is compared with some *state-of-the-art* feature descriptors used for Bangla word recognition [53]. This comparison is performed in terms of recognition accuracy, feature dimension and feature computation time (see Table 9). Feature descriptors considered for comparison include, viz., Local Gradient of Histogram (LGH), Pyramid Histogram of Oriented Gradient (PHOG) and combination of GABOR and PHOG called G_PHOG [53]. Although the recognition accuracy achieved by LGH is very close to the proposed technique but if we consider the other parameters, it is observed that the proposed technique outperforms the LGH and others.

Finally, the proposed method is compared with some recently published holistic word recognition methods [26, 31, 36, 42]. Work reported in [36, 42] has dealt with Bangla HWR, whereas the remaining two methods have developed for the recognition of handwritten English [26] and Hindi [31] words. For the comparison, our fivefold cross-validations result is considered. The best, worst, average case recognition accuracies along with feature dimension and the classifier used by these techniques are summarized in Table 10. It also includes deviation from average recognition rate. From the table, it is clear that the present technique outperforms the said methods. Besides the results, we would also like to mention here that, in the literature no holistic Bangla word recognition work has been reported, where such a large number of word classes are considered.

5 Conclusion

In the present work, a holistic HWR scheme is developed for the recognition of handwritten Bangla words. For that purpose, a shape-based feature descriptor which is a combination of Elliptical, Tetragonal and Vertical pixel density histogram-based features is designed. Recognition process is carried out using two well-known classifiers, viz., MLP and SVM. But the proposed method performs comparably better with SVM than MLP. In holistic word recognition approach, as a given word image is considered as a single and indivisible unit, shape dissimilarity of the words belonging to different classes can be very effective during recognition. This very fact motivates us to design the proposed feature descriptor. For the evaluation purpose, a database of 18,000 Bangla handwritten word images belonging to 120 different word classes is also prepared and it is also made freely available to the research community.

Although the proposed method is currently used for recognition of handwritten Bangla words but as it emphasizes on shape-level information, it may equally be useful for recognition of handwritten words written in other scripts. In addition to that, no skew or slant correction has been undertaken at the preprocessing stage (which is very common in handwritten word recognition) but still it has achieved reasonable accuracy. Thus in future, inclusion of a suitable skew and slant correction module can make it more effective.

Compliance with ethical standards

Conflict of interest We declare that we do not have any conflict of interest.

References

1. Chacko BP, Krishnan VRV, Raju G, Anto PB (2012) Handwritten character recognition using wavelet energy and extreme learning machine. Int J Mach Learn Cybern 3(2):149–161
2. Prasad JR, Kulkarni U (2015) Gujrati character recognition using weighted k-NN and mean χ^2 distance measure. Int J Mach Learn Cybern 6(1):69–82
3. Pal U, Roy K, Kimura F (2009) A lexicon-driven handwritten city-name ecognition scheme for Indian postal automation. IEICE Trans Inf Syst 92(5):1146–1158
4. Pal U, Roy RK, Kimura F (2012) Multi-lingual city name recognition for Indian postal automation. In: 2012 international conference on frontiers in handwriting recognition (ICFHR), pp 169–173
5. Morita M, El Yacoubi A, Sabourin R, Bortolozzi F, Suen CY (2001) Handwritten month word recognition on Brazilian bank cheques. In: Sixth international conference on document analysis and recognition. Proceedings, pp 972–976
6. Bunke H, Bengio S, Vinciarelli A (2004) Offline recognition of unconstrained handwritten texts using HMMs and statistical language models. IEEE Trans Pattern Anal Mach Intell 26 (6):709–720
7. Madhvanath S, Govindaraju V, Ramanaprasad V, Lee D-S, Srihari SN (1995) Reading handwritten US census forms. In: Proceedings of the third international conference on document analysis and recognition, vol 1, pp 82–85
8. Srihari SN, Shin YC, Ramanaprasad V, Lee D-S (1995) Name and address block reader system for tax form processing. In: Proceedings of the third international conference on document analysis and recognition, vol 1, pp 5–10
9. Prasad JR, Kulkarni U (2015) Gujarati character recognition using adaptive neuro fuzzy classifier with fuzzy hedges. Int J Mach Learn Cybern 6(5):763–775
10. Broumandnia A, Shanbehzadeh J, Varnoosfaderani MR (2008) Persian/arabic handwritten word recognition using M-band packet wavelet transform. Image Vis Comput 26(6):829–842
11. El Qacimy B, Kerroum MA, Hammouch A (2015) Word-based Arabic handwritten recognition using SVM classifier with a reject option. In: 2015 15th international conference on intelligent systems design and applications (ISDA), pp 64–68
12. Dehghan M, Faez K, Ahmadi M, Shridhar M (2001) Handwritten Farsi (Arabic) word recognition: a holistic approach using discrete HMM. Pattern Recognit 34(5):1057–1065
13. Liu C-L, Koga M, Fujisawa H (2002) Lexicon-driven segmentation and recognition of handwritten character strings for Japanese address reading. IEEE Trans Pattern Anal Mach Intell 24 (11):1425–1437
14. Su T (2013) Chinese handwriting recognition: an algorithmic perspective. Springer, Berlin
15. Srihari SN, Yang X, Ball GR (2007) Offline Chinese handwriting recognition: an assessment of current technology. Front Comput Sci China 1(2):137–155
16. Koerich AL, Sabourin R, Suen CY (2005) Recognition and verification of unconstrained handwritten words. IEEE Trans Pattern Anal Mach Intell 27(10):1509–1522
17. Bunke H (2003) Recognition of cursive Roman handwriting: past, present and future. In: Seventh international conference on document analysis and recognition. Proceedings, pp 448–459
18. Bozinovic RM, Srihari SN (1989) Off-line cursive script word recognition. IEEE Trans Pattern Anal Mach Intell 11(1):68–83
19. “Bengali language”. https://en.wikipedia.org/wiki/Bengali_language. Accessed 27 Dec 2017
20. Das N, Sarkar R, Basu S, Saha PK, Kundu M, Nasipuri M (2015) Handwritten Bangla character recognition using a soft computing paradigm embedded in two pass approach. Pattern Recognit 48 (6):2054–2071
21. Rahman MM, Akhand MAH, Islam S, Shill PC, Rahman MMH (2015) Bangla handwritten character recognition using convolutional neural network. Int J Image Graph Signal Process 7(8):42
22. Das N, Basu S, Saha PK, Sarkar R, Kundu M, Nasipuri M (2015) A GA based approach for selection of local features for recognition of handwritten Bangla numerals. arXiv Prepr. arXiv:1501.05495
23. Plamondon R, Srihari SN (2000) Online and off-line handwriting recognition: a comprehensive survey. IEEE Trans Pattern Anal Mach Intell 22(1):63–84
24. Tappert CC, Suen CY, Wakahara T (1990) The state of the art in online handwriting recognition. IEEE Trans Pattern Anal Mach Intell 12(8):787–808
25. Ruiz-Pinales J, Jaime-Rivas R, Castro-Bleda MJ (2007) Holistic cursive word recognition based on perceptual features. Pattern Recognit Lett 28(13):1600–1609
26. Dasgupta J, Bhattacharya K, Chanda B (2016) A holistic approach for Off-line handwritten cursive word recognition using directional feature based on Arnold transform. Pattern Recognit Lett 79:73–79
27. Koerich AL, Sabourin R, Suen CY (2003) Large vocabulary off-line handwriting recognition: a survey. Pattern Anal Appl 6 (2):97–121
28. Plötz T, Fink GA (2009) Markov models for offline handwriting recognition: a survey. Int J Doc Anal Recognit 12(4):269–298
29. Madhvanath S, Govindaraju V (2001) The role of holistic paradigms in handwritten word recognition. IEEE Trans Pattern Anal Mach Intell 23(2):149–164

30. Madhvanath S, Kleinberg E, Govindaraju V (1999) Holistic verification of handwritten phrases. IEEE Trans Pattern Anal Mach Intell 21(12):1344–1356
31. Malakar S, Sharma P, Singh PK, Das M, Sarkar R, Nasipuri M (2017) A holistic approach for handwritten hindi word recognition. Int J Comput Vi. Image Process 7(1):59–78
32. Tamen Z, Drias H, Boughaci D (2017) An efficient multiple classifier system for Arabic handwritten words recognition. Pattern Recognit Lett 93:123–132
33. Pechwitz M, Maddouri SS, Märgner V, Ellouze N, Amiri H (2002) IFN/ENIT-database of handwritten Arabic words. Proc CIFED 2:127–136
34. Roy PP, Dey P, Roy S, Pal U, Kimura F (2014) A novel approach of Bangla handwritten text recognition using HMM. In: 2014 14th international conference on frontiers in handwriting recognition (ICFHR), pp 661–666
35. Roy PP, Bhunia AK, Das A, Dey P, Pal U (2016) HMM-based Indic handwritten word recognition using zone segmentation. Pattern Recognit 60:1057–1075
36. Vajda S, Roy K, Pal U, Chaudhuri BB, Belaid A (2009) Automation of Indian postal documents written in Bangla and English. Int J Pattern Recognit Artif Intell 23(8):1599–1632
37. Bhowmik TK, Roy U, Parui SK (2012) Lexicon reduction technique for Bangla handwritten word recognition. In: 2012 10th IAPR international workshop on document analysis systems (DAS), pp 195–199
38. Bhowmik TK, Parui SK, Roy U (2008) Discriminative HMM training with GA for handwritten word recognition. In: ICPR 2008. 19th international conference on pattern recognition, pp 1–4
39. Bhowmik S, Roushan MG, Sarkar R, Nasipuri M, Polley S, Malakar S (2014) Handwritten Bangla word recognition using HOG descriptor. In: Proceedings—4th international conference on emerging applications of information technology, EAIT
40. Bhowmik S, Malakar S, Sarkar R, Nasipuri M (2014) Handwritten Bangla word recognition using elliptical features. In: 2014 international conference on computational intelligence and communication networks (CICN), pp 257–261
41. Bhowmik S, Polley S, Roushan MG, Malakar S, Sarkar R, Nasipuri M (2015) A holistic word recognition technique for handwritten Bangla words. Int J Appl Pattern Recognit 2(2):142–159
42. Barua S, Malakar S, Bhowmik S, Sarkar R, Nasipuri M (2017) Bangla handwritten city name recognition using gradient-based feature, vol 515
43. Ghosh M, Malakar S, Bhowmik S, Sarkar R, Nasipuri M (2017) Memetic algorithm based feature selection for handwritten city name recognition, vol 776
44. Ban JC (2015) Neural network equations and symbolic dynamics. Int J Mach Learn Cybern 6(4):567–579
45. Li Z, Zhou M, Lin H, Pu H (2014) A two stages sparse SVM training. Int J Mach Learn Cybern 5(3):425–434
46. Liu CL, Koga M, Fujisawa H (2005) Gabor feature extraction for character recognition: comparison with gradient feature. In: Eighth international conference on document analysis and recognition (ICDAR'05), pp 121–125
47. Sarkar R, Das N, Basu S, Kundu M, Nasipuri M, Basu DK (2012) CMATERdb1: a database of unconstrained handwritten Bangla and Bangla-English mixed script document image. Int J Doc Anal Recognit 15(1):71–83
48. "CMATERdb2.1.2". https://drive.google.com/file/d/0B8rZngAQdufXemZmYlI2M2xwdXc/view?usp=sharing
49. Otsu N (1975) A threshold selection method from gray-level histograms. Automatica 11(285–296):23–27
50. Soille P (2005) Erosion and dilation. In: Morphological image analysis. Springer, pp 63–103
51. Yang M, Kpalma K, Ronsin J (2008) A survey of shape feature extraction techniques. IN-TECH
52. Smith TC, Frank E (2016) Introducing machine learning concepts with WEKA. Stat Genomics Methods Protoc 1418:353–378
53. Bhunia AK, Das A, Roy PP, Pal U (2015) A comparative study of features for handwritten Bangla text recognition. In: 2015 13th international conference on document analysis and recognition (ICDAR), pp 636–640