Signature and Date-Based Document Image Retrieval

Ranju Mandal

Master of Computer Application (MCA)

School of Information and Communication Technology
Griffith Sciences
Griffith University

Submitted in fulfilment of the requirements of the degree of Doctor of Philosophy

April, 2016

Abstract

It is a common organisational practice nowadays to store and maintain large digital databases in an effort to move towards a paperless office. Large quantities of administrative documents are often scanned and archived as images (e.g. the 'Tobacco' dataset [1]) without adequate indexing information. Consequently, such practices have created a tremendous demand for robust ways to access and manipulate the information that such images contain. Manual processing (i.e. indexing, sorting or retrieval) of documents from these huge collections need substantial human effort and time. So, automatic processing of documents is required for office automation.

In this context, Document Image Analysis (DIA) has enjoyed many decades of popularity as a research area to address these issues because of its huge application potential in many fields such as academics, banking and in industry. A document repository available for analysis in such a domain contains a large collection of heterogeneous documents. Automatic analysis of such large document database has been an interesting and challenging research field for many years, specifically due to the diverse layouts and contents. One way to efficiently search and retrieve documents from a large repository is to fully convert the documents to an editable representation (i.e. through Optical Character Recognition) and index them based on their content. There are many factors (e.g. high cost, low document quality, non-text components, etc.) which prohibit complete conversion of a document to an editable form. Hence, other components of a document, namely signatures, dates, logos, stamps/seals, etc. are worthy consideration for indexing, without the requirement for complete OCR.

Handwritten signatures are pure behavioural biometrics, which have been accepted as an official means to verify personal identity for legal purposes on documents such as cheques, credit cards, wills, for example. Dates (i.e. timestamps) usually coexist with signatures and are needed to properly validate each signature. In addition, dates themselves are a useful piece of information for date-based document searching and retrieval. Furthermore, combined information of signatures and dates could be useful to narrow down the search results. Obtaining relevant documents from large repositories using signatures and dates as a query is the main objective of the proposed content-based document retrieval approach. This research study mostly deals with handwritten signatures and date information which are two different entities with different properties

and are unconstrained in nature. Hence, two different independent multi-stage systems are proposed to address the signatures and dates separately.

Automatic detection, segmentation, and matching of signatures are the three major steps in a signature-based document image retrieval system. First, a component-level feature extraction and Support Vector Machine (SVM)-based classification technique detected the potential signature components from a document. Second, a density-based spatial clustering approach was used to segment these identified components as signature candidate. Finally, the segmented signature shape was characterised using a Bag-of-Visual-Words-based approach for matching with a query signature.

Similarly, two multi-stage approaches are proposed for date field extraction. The datasets considered for experiments are multi-script in nature. Hence, identification of scripts was performed in order to develop a robust system for multi-script documents. In the first stage, the document script was identified using a feature extraction and classification-based approach. A water reservoir-based technique was used to extract the foreground and background information from a handwritten line for feature extraction at the script identification stage.

First, a segmentation-based approach, where individual date components such as month-word (a month in word form i.e. January, Jan, etc.), numerals, punctuation and contraction categories were segmented and labelled from a text line. Profile-based features with a Dynamic Time Warping (DTW)-based similarity measure technique were applied to identify month-words. A recognition approach based on gradient-based features and an SVM classifier were applied to recognise other elements (i.e. numerals and punctuation) of the date. The second approach is a segmentation free-based method. Sliding window-wise Local Gradient Histogram (LGH)-based features and a character-level Hidden Markov Model (HMM)-based technique were applied for segmentation and recognition of date components. Next, Histogram of Gradient (HoG) features and SVM-based classifiers were used to improve the results obtained from the HMM-based recognition system. Subsequently, both numeric and semi-numeric (containing month fields as text strings) regular expressions of date patterns were searched for extraction in labelled components.

Statement of Originality

This work has not previously been submitted for a degree or diploma in any university. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made in the thesis itself.

Signed:

_____

Ranju Mandal
April 23, 2016

Contents

List of Figures

List of Tables

Acronyms

ANN	**A**rtificial **N**eural **N**etwork
DAR	**D**ocument **A**nalysis and **R**ecognition
DBSCAN	**D**ensity-**B**ased **S**patial **C**lustering of **A**pplication with **N**oise
DCT	**D**iscrete **C**osine **T**ransforms
DIA	**D**ocument **I**mage **A**nalysis
DIR	**D**ocument **I**mage **R**etrieval
DTW	**D**ynamic **T**ime **W**arping
GVF	**G**radient **V**ector **F**low
HOG	**H**istogram of **O**riented **G**radient
HMM	**H**idden **M**arkov **M**odel
IR	**I**nformation **R**etrieval
KNN	**K**-**N**earest **N**eighbours
LGH	**L**ocal **G**radient **H**istogram
LBP	**L**ocal **B**inary **P**attern
MLP	**M**ulti-**L**ayer **P**erceptron
MQDF	**M**odified **Q**uadratic **D**iscriminant **F**unction
MRF	**M**arkov **R**andom **F**ield
OCR	**O**ptical **C**haracter **R**ecognition
RBF	**R**adial **B**asis **F**unction
RLSA	**R**un **L**ength **S**moothing **A**lgorithm
SIFT	**S**cale-**I**nvariant **F**eature **T**ransform
SURF	**S**peeded **U**p **R**obust **F**eatures
SPM	**S**patial **P**yramid **M**atching
SVM	**S**upport **V**ector **M**achine

Acknowledgements

I am sincerely grateful to my principal supervisor Prof. Michael Blumenstein for his excellent guidance and support during my PhD candidature. His encouragement, interactive scientific discussions, regular meetings and all his contributions have made my experience during the tenure of PhD a very productive and fruitful one. I wish to express my sincere gratitude to my external supervisor Prof. Umapada Pal, Indian Statistical Institute, for his constant guidance, motivation and support during my PhD tenure. It was under Prof. Pal's guidance that I started my research career and have learned the philosophy and techniques of research. He has been a continuous source of inspiration. I would also like to thank my associate supervisor, Prof. Graham Leedham, Griffith University, for his support during my PhD candidature.

My special thanks to Dr. Partha Pratim Roy, Indian Institute of Technology, Roorkee for his advice on refining the research objectives, fruitful collaboration and support. I would like to thank my colleague Dr. Nabin Sharma and Dr. Srikanta Pal, Griffith University for many discussions and support during the PhD candidature.

I would like to thank Ms. Victoria Wheeler and Ms. Kate Schurmann, Secretary, School of ICT, Griffith University, and all other staff members for their help and support. I am thankful to my friends and colleagues, Rituraj Kunwar, Abhijit Das, Sukalpo Chanda, Adel Fazel, Arpita Chakraborty, Alireza Jolfaei and Rupam Deb for their help, support and inspiration. My special thanks to Ms. Claire Rodway for proofreading the chapters in this thesis.

My regards and respect to my parents and all my family members for their constant inspiration, love and support, without which it would not have been possible to complete my PhD journey.

List of Publications

List of Journal

1. **Ranju Mandal**, Partha Roy, Umapada Pal and Michael Blumenstein, "Signature-Based Multi-Script Document Retrieval using Bag-of-Visual-Words", Image and Vision Computing (Under revision).

2. **Ranju Mandal**, S. Pal, Partha Pratim Roy, Umapada Pal and Michael Blumenstein, "Spatial Pyramid Matching-based Multi-script Off-line Signature Identification", International Journal of American Society of Questioned Document Examiners, Vol. 18, No. 1, pp. 69-75, 2015.

3. **R. Mandal**, P. P. Roy, and U. Pal, and M. Blumenstein, "Multi-lingual date field extraction for automatic document retrieval by machine", Information Sciences, Vol. 314, pp. 277-292, 2015.

4. **R. Mandal**, P. P. Roy, and U. Pal, "Signature segmentation from machine printed documents using contextual information", International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI), vol. 26, no. 7, pp. 1-25, 2012.

List of Conferences

1. **Ranju Mandal**, Partha Roy, Umapada Pal and Michael Blumenstein, "Date Field Extraction from Handwritten Documents Using HMMs", In Proc. *International Conference on Document Analysis and Recognition (ICDAR)*, pp. 866-870, 2015.

2. **R. Mandal**, P. P. Roy, and U. Pal, and M. Blumenstein, Signature segmentation and recognition from scanned documents, In Proc. *Intelligent Systems Design and Applications (ISDA)*, pp. 80-85, 2013.

3. Nabin Sharma, **Ranju Mandal**, Rabi Sharma, Umapada Pal, and Michael Blumenstein, Bag-of-visual words for word-wise video script identification: A study, In

Proc. *2015 International Joint Conference on Neural Networks (IJCNN)*, pp.1-7, 2015.

4. Nabin Sharma, **Ranju Mandal**, Rabi Sharma, Partha Pratim Roy, Umapada Pal, and Michael Blumenstein, Multilingual text recognition from video frames, In Proc. *International Conference on Document Analysis and Recognition (ICDAR)*, pp. 951-955, 2015.

5. Nabin Sharma, **Ranju Mandal**, Rabi Sharma, Umapada Pal, and Michael Blumenstein, ICDAR2015 Competition on Video Script Identification (CVSI), In Proc. *International Conference on Document Analysis and Recognition, (ICDAR)*, pp. 1196-1200, 2015.

Dedicated to my grandparents, parents, and all family members ...

Chapter 1

Introduction

Document Image Analysis (DIA) is an area of research interest with a large variety of challenging problems. DIA has enjoyed many decades of popularity as a research area because of its huge application potential in the academic field and in industry. Researchers have been working for the last few decades on this topic as witnessed by the scientific literature. DIA is a subfield of image analysis and it inherits the more general techniques of image analysis. Furthermore, DIA has also acquired its own techniques, specially designed for their applications. Optical Character Recognition (OCR) is one of the first problems in the DIA domain, and has been widely explored. The objective of OCR is to produce a compatible electronic text format of a document, which makes a document easier and quicker to access. Commercial OCRs for multiple languages are also available with various applications and high user acceptance.

In Document Image Analysis, paper documents are initially scanned and stored in a document repository. The major tasks in DIA are interpretation, retrieval, indexing, and managing. The main objective of DIA is to convert a document from pixel information into a format that can be semantically read by a computer. In order to achieve this goal, a set of simple techniques and procedures are applied on document images. The idea of using computers to retrieve information relevant to a query became a very popular practice many decades ago. Obtaining such information resources relevant to the query is the main activity of Information Retrieval (IR). Many universities and public libraries use IR systems to provide access to books, journals and other documents. Web search engines are the best examples of real-time IR application and most of the developments have been done for editable electronic documents, and therefore it is not applicable on documents which are in image format.

Document Image Retrieval (DIR) is a technique used to retrieve relevant documents from a document image repository. Researchers have been working on this field for the last few decades and many techniques have been developed to retrieve document

images. A document may contain different types of components such as signature, logo, seal, stamp, etc. Images of word, signature, logo, stamp, seal, etc. are often used as a query for document image retrieval. Fig. 1.1 shows sample images of logo, stamp and signatures from the repository of "Tobacco" [1] administrative documents and some seals from the historical archive of border records from the Civil Government of Girona [9].

The remainder of this chapter is organized into different sections as follows: Firstly, a brief overview is presented in Section 1.1. Section 1.2 provides a concept of a DIR system. Then, the significance of signature and date-based DIR is described in Section 1.3. The motivation in this research is presented in Section 1.4 and in Section 1.5 objectives and research questions are described. The detailed discussion of the scope of the research is presented in 1.6. Section 1.7 presents the research contributions made in this thesis. Finally, the organization of the thesis is presented in Section 1.8.

1.1 Overview

In an attempt to move toward a paperless office, large quantities of printed documents are often scanned and archived as images, without adequate index information. It is a common organisational practice nowadays to store and maintain large databases of document images because of the economic feasibility. As a consequence, such practice has created a tremendous demand for robust ways to access and manipulate the information that these images contain.

One way to provide traditional database indexing and retrieval capabilities is to fully convert the document to an electronic representation which can be indexed automatically. However, there are many factors which prohibit complete conversion including high cost, low document quality, and the fact that many non-text components cannot be adequately represented in a converted form. Some samples of non-text components are shown in Fig. 1.1. In such cases, it can be advantageous to maintain a scanned copy of the document and use it in image form.

The performances of traditional OCR-based document indexing techniques on complex document data (i.e. documents with graphics and images) are not satisfactory. Additionally, it is also observed that the retrieval of partially converted documents has some useful application areas. A significant number of methods [10–12] have been developed by researchers to access and manipulate document images without complete and accurate conversion. The techniques developed have mainly been for the direct characterization, manipulation and retrieval of images of documents containing text, graphics, and scene images.

(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

(m) (n) (o) (p)

(q) (r) (s) (t)

FIGURE 1.1: Sample images of stamp and logo from the document repository of 'Tobacco-800'. (a-h) Samples of logo (i-l) Samples of stamp. (m-p) Samples of signature. (q-t) Samples of seal.

1.2 Concept of Document Image Retrieval

Document Image Retrieval (DIR) is a very attractive field of research with a continuous growth of interest and increasing security requirements for the development of a

modern society. Complex documents present a great challenge to the field of document recognition and retrieval with the combined presence of noise, handwriting, signature, logos, seal, and machine printed text with different fonts, and rule lines making it more challenging. Therefore, many algorithms that work relatively well on simple documents are not effective on such documents for the existing algorithmic constraints.

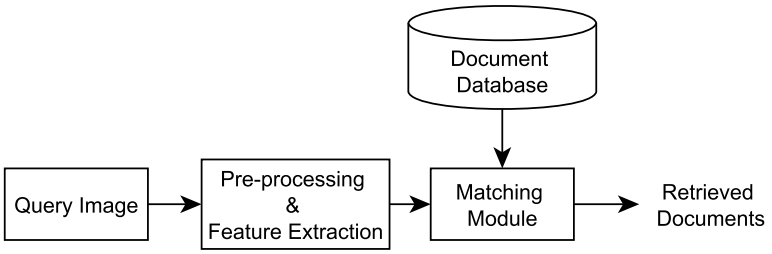


FIGURE 1.2: A block diagram of document image retrieval system

The primary task of processing these complex documents is to isolate different contents present in the documents. Once the contents are separated out, then they can be called indexed documents which are ready to be used for a content-based image retrieval system. A signature-based document image retrieval system is presented here as an example. A simple block diagram of document image retrieval system from indexed document database is presented in Fig. 1.2. However, before retrieval the repository needs to be indexed based on key information. A simple block diagram of indexing of documents based on signature information is shown in Fig. 1.3.

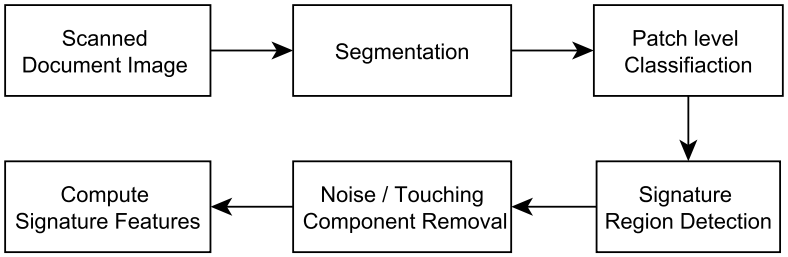


FIGURE 1.3: Block diagram of signature-based document indexing

The document image understanding, covering a variety of documents such as bank cheques [13], business letters [14, 15], forms, and technical articles, has been an interesting research area for a long time. In the context of document image retrieval, a signature provides an important form of indexing that enables effective explanation of data [11]. Given a large collection of documents, searching for a specific signature is a highly effective way of retrieving documents from the associated organization. Building an access to these document images requires designing a mechanism for search and retrieval of image data from the document image collection. In searching complex documents, such as a

repository of archival office documents [1], the task of relevance is relating the signature in a given document to the closest matches within a database of documents; this is known as the signature retrieval task. Given a database of signed documents, it would be of interest to relate a queried document to other documents in this database which have been signed by the same author.

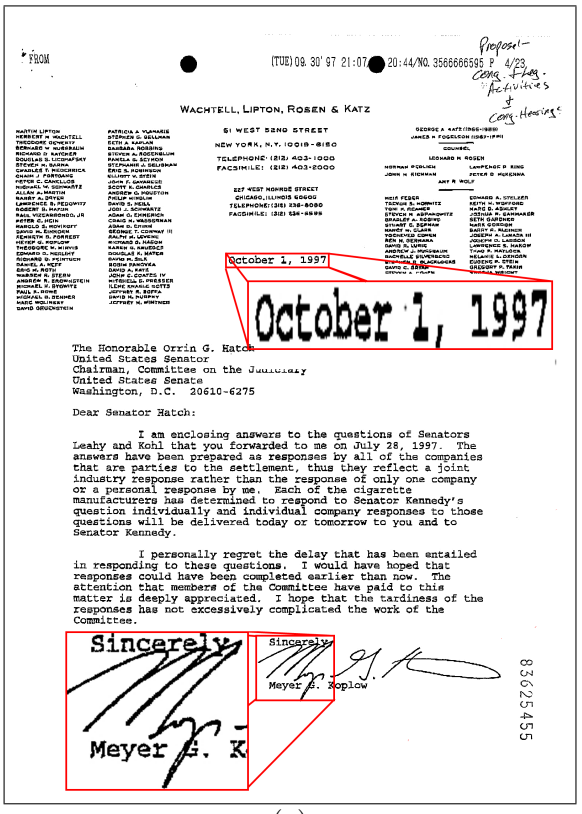
1.3 Importance of Signature and Date-Based Document Image Retrieval

Biometrics is one of the most widely used approaches for personal identification and verification. Among all the biometric authentication systems, handwritten signature is a pure behavioural biometric which has been accepted as an official means to verify personal identity for legal purposes on documents [13] such as cheques, credit cards, wills, etc.

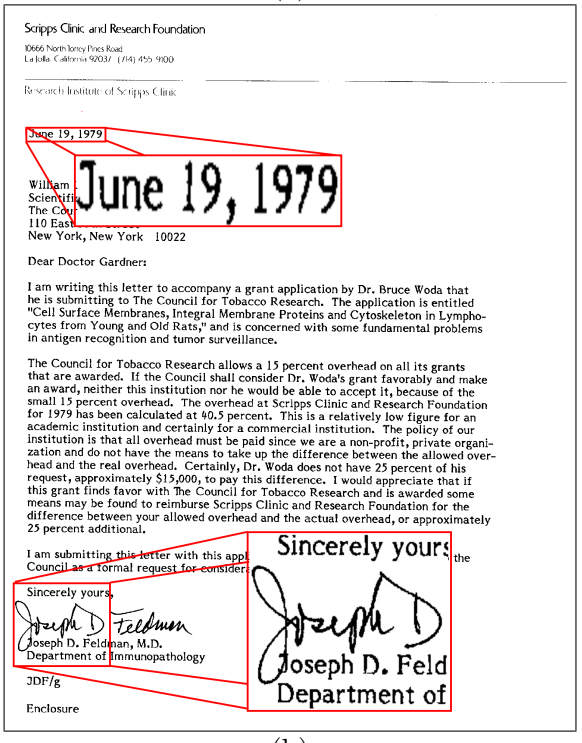
A signature provides useful information about a person as it consists of unique properties of human behaviour. It is therefore used for identification/verification purposes and thus signature verification and recognition can be widely used in many organizations such as banking, finance industry, etc. The signature in a document is examined by forensic document analysis experts for authenticating documents and to restrict fraudulent acts. Sample documents containing signature and date information are shown in Fig. 1.4.

Moreover, in the field of document image processing and retrieval, a signature can be used as key information for searching and retrieval of relevant documents (i.e. document contains the same signature) from large heterogeneous document image databases. The traditional OCR-based techniques used for indexing and searching from document image databases have some disadvantages because of their limitations in working on handwritten information, and also because the computational overhead is high due to the classification and conversion. Many documents may contain signatures which can provide rich and unique information about the document. Therefore, handwritten signatures will undoubtedly add advantage for document searching and retrieval.

Dates (i.e. timestamp) usually coincide with signatures and are needed to properly validate the signature. A signature is always associated with date information in administrative, financial, legal documents, etc. In such cases, the analysis of a timestamp (i.e. a timestamp is a sequence of characters or encoded information identifying when a certain event occurred) is also necessary to develop a more useful searching technique. Furthermore, the date itself is a useful piece of information, which could be used as a key in various applications, such as date-based document searching and retrieval



(a)



(b)

FIGURE 1.4: (a, b) Sample documents with signature and date information. Signature and date information is zoomed.

of document repositories (i.e. administrative documents, historical archives, and postal

mails). Therefore, signature and date both can be used individually or as compound key (combined) information to achieve the retrieval in the field of document image retrieval.

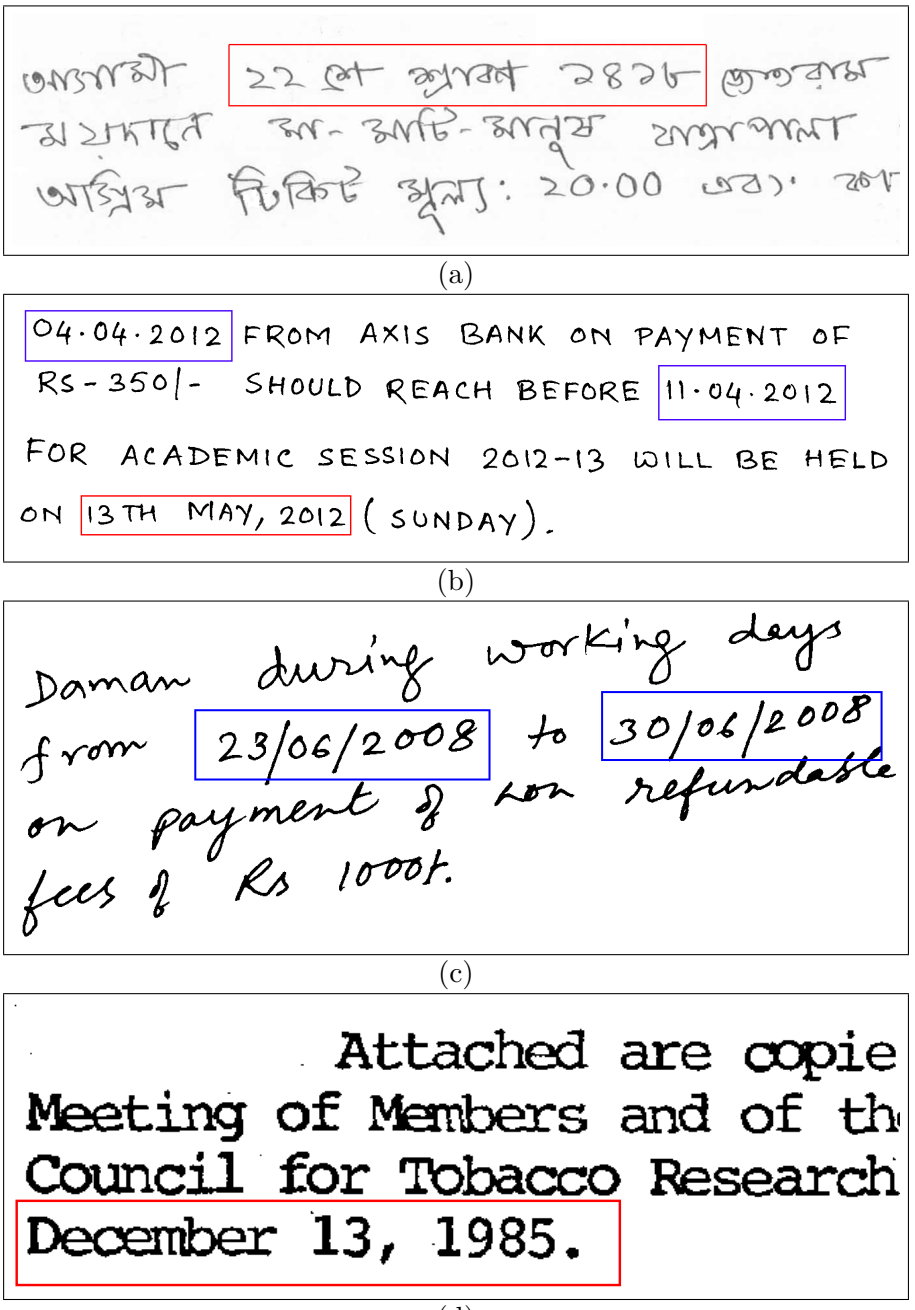
Dates can usually be found in all types of documents. Sometimes rubber date stamps are used in offices to stamp the current date on paper documents, to record when the document was received or processed. The date-field can also be found in document text portions. Four sample documents of different scripts with date information are shown in Fig. 1.5. As an example of the postal document, Fig. 1.6 shows an Indian handwritten Bangla postcard document containing an English date. English (Roman) script is widely used in India with popular handwritten Indic scripts like Bangla and Devnagari in a single document. Thus, an English date is very usual to be found in documents of Indic scripts.

1.4 Motivation

Nowadays, large institutions and corporations still receive a high volume of communication in paper form, for instance because of their legal significance, as a backlog of old documents to be archived, or as general-purpose correspondence. Google and Yahoo have recently announced their intention to make handwritten books accessible through their search engines [16]. In this context, field-based document image retrieval will be a valuable tool for users to browse the contents of these books. Moreover, recent surveys and interviews with the most active academic and industrial experts in the field indicate that over 55 billion cheques are processed annually in North America at a cost of 25 billion dollars [17].

The word image spotting technique is a classical and popular approach applied for document image retrieval because of the limitations and failures of available OCR engines [18] on handwritten documents. Such OCRs are mainly designed for the single script documents and mostly deal with Roman script. Significant work has been undertaken [19–21] in the area of word spotting to make the handwritten text available for searching and browsing. Likewise, signature and date information can also be used for searching and browsing of document images because of its huge application potentials.

Signatures have also been used for authentication purposes for centuries. Unlike other authentication protocols using passwords, access cards, or PIN codes, the ability of a human to sign is less likely to be lost, forgotten, or stolen. An individual should be able to produce their signature at anytime and anywhere upon request as a proof of identity. Signatures are unique amongst people and difficult to imitate. This confirms the convenience of signatures as a behavioural biometric. Moreover, an off-line signature does not require special sample capturing devices like in hand vein, fingerprint, or retina



(a) (b) (c) (d)

FIGURE 1.5: Example of handwritten and printed documents containing numeric and semi-numeric date-fields in Bangla (a) and English (b,c,d). Numeric and semi-numeric date-fields are marked with blue and red rectangle, respectively.

recognition. Automatic processing of date also has many potential applications such as automatic bank cheque processing [22] and date-based document searching and retrieval.

Considerable research has previously been undertaken in the area of signature verification [23–26]. However, less attention has been given to the usefulness of signature and date information in document searching and retrieval. It is to be noted that proper

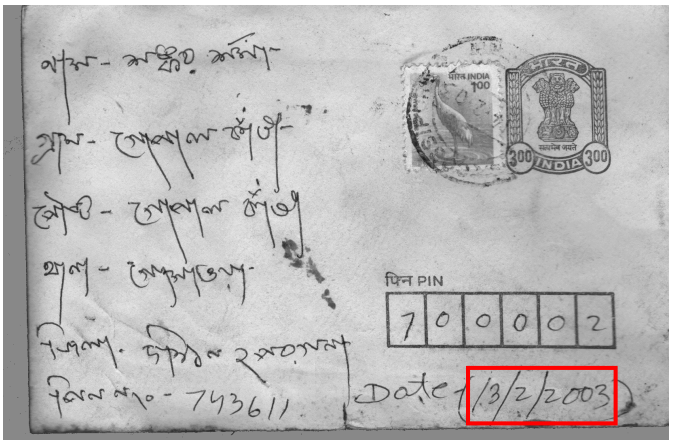


FIGURE 1.6: Indian postcard containing handwritten date-field is marked by red rectangle.

segmentation and recognition of signature is required before feeding those to a signature verification system. As not much research has been undertaken on signature and date-based document image retrieval, the research on this topic will be a challenging one.

1.5 Objectives

Development of a robust signature and date-based document image retrieval techniques are proposed in this thesis and the objectives of the system are:

(i) Identification and retrieval of multi-script document images.

(ii) Segmentation and recognition of signatures from scanned document images.

(iii) Automatic document image retrieval/indexing based on the signature information.

(iv) Extraction of numeric and semi-numeric date from multi-script document images.

(v) Automatic document image retrieval/indexing based on date information (i.e. timestamp).

(vi) Automatic document image retrieval (e.g. administrative documents, bank cheques, etc.) using combined information of signature and date.

Detection and recognition of signatures as well as different formats (numeric and semi-numeric) of date information for the document image retrieval/indexing are the main objective of the proposed research. Signature and date can both be very useful as a key in the field of document retrieval. Adequate research has not been undertaken to date on this topic. It would also be interesting to extend this proposed method to a

multi-script environment. Since no work has been undertaken on the classification of signature and other contents such as seal, logo, handwritten annotation, etc., research establishing whether it is possible to classify these components will also be challenging.

The above motivation and objectives have led to the following research questions:

(i) How is a signature differentiated from other graphics such as logo, seal, drop caps, etc.?

(ii) How is a signature differentiated from handwritten/printed text?

(iii) How does the system handle a signature touched/overlapped with printed text, graphics, etc.?

(iv) How can a date written in multiple scripts be extracted from a document?

(v) Can signature and date be used for content-based document retrieval? Can the approach be extended to multi-lingual documents?

1.6 Scope of Research

In this research, the primary focus will be on retrieval of documents based on the user provided query information (i.e. signature or date). Detection and extraction of signature and date information from a scanned document image are the pivotal tasks in this scenario. For example, in the case of a document image, document retrieval based on such information includes the following points:

- Identification of the scripts in a document.
- Determine the printed text and non-text regions (i.e. signatures).
- Classification of handwritten and printed text
- Detection/identification of signatures.
- Identification of date elements (i.e. numerals, alpha-numeric month and punctuation)
- Techniques for pattern matching.

To develop a robust content-based document image retrieval system a few shortcomings of the previously proposed approaches are identified. The following observations are made.

- The script identification problem needs to be addressed.

- Signature detection and segmentation from multi-lingual documents need to be addressed.
- Robust signature shape descriptors have to be developed which works for both English and non-English signatures.
- Development of a robust date pattern detection and searching algorithm for multi-script documents requires attention in every stage.
- Development of a robust system to avoid the problem of pre-segmentation of date-fields.

In multi-lingual and multi-script countries such as India, retrieval of multi-script documents can be very useful. A single document in multi-lingual countries may contain more than one script. For example, the West Bengal state of India uses three different scripts in official documents (i.e. English, Bangla, and Hindi). Therefore, three scripts namely, English, Hindi and Bangla for multi-lingual environments are considered for most of the experiments. The detection and extraction of this information from multi-script documents are the scope of the present study. However, some experiments are conducted on English scripts due to time constraints and lack of proper sets of data.

1.7 Contributions

The proposed techniques described and introduced in this thesis are based on a number of research areas within document image retrieval. The main research contributions made in this thesis are presented below.

- A novel approach for automatic detection and segmentation of signature from printed signed documents using contextual information is proposed.
- A novel approach for signature segmentation using Bag-of-Visual-Words (BoVW) features is proposed. The approach consists of densely sampled descriptors on a regular grid with Spatial Pyramid Matching (SPM) which presents a robust model for signature shape representation.
- An approach for date field extraction from multi-script documents. A novel idea of script identification is proposed based on combined information of foreground and background of a word. Next, a multi-step approach is proposed to identify all the components related to date and search for a valid pattern.
- A multi-stage approach for extraction of date field using two classifiers namely HMMs and SVMs in tandem. The proposed character level HMM-based system

avoids the problem of pre-segmentation of date fields. Next, an application of discriminative classifier trained with numerals to improve the accuracy is proposed.

1.8 Organization of the Thesis

Based on the proposed signature and date-based document image retrieval systems and the number of research areas involved in the different level, methodologies are divided across different chapters. Each section in a chapter presents different modules for solving a particular task for realizing the systems. The remainder of the chapters in this thesis is organized as follows.

- Chapter 2 presents a review of the recent development and approaches in the related area of automatic signature and date-based document image retrieval techniques.
- Chapter 3 contains the research methodology and the proposed approach for signature-based document retrieval. Different feature extraction procedures used in the proposed system are discussed.
- Chapter 4 describes the proposed approach of date-based document retrieval. Two different approaches for document retrieval based on date information are presented.
- Chapter 5 summarizes the research contributions and outlines the future research areas in content-based document retrieval.

Chapter 2

Literature Review

A brief review of the relevant research and state-of-the-art approaches related to the document image retrieval based on signature and date information are presented and discussed in this chapter. To identify the research scope the limitations of the approaches in extant literature are investigated. The proposed technique of signature-based document retrieval includes various stages of processing such as signature segmentation and signature recognition, whereas date-based technique includes script identification, month-word spotting, numerals, and punctuations extraction (e.g. slash '/', hyphen '-', and period '.'). Hence, the available techniques relevant to sub-modules of this study have also been reviewed to acquire concepts of the respective fields.

The review of related work presented in this chapter is organised into different sections in this chapter. As multi-script documents were considered for the experiment, published works on script identification are reviewed in Section 2.1. The signature detection and recognition techniques towards signature-based document retrieval are presented in Section 2.2 and 2.3. The date-based document retrieval system involves alpha-numeric month detection, numerals and punctuation recognitions. Section 2.4 and Section 2.5 present recently published works on word spotting and numeral extraction respectively. A few works on date extraction are presented in Section 2.6. Section 2.7 presents some popular datasets which have used for the experiments discussed here. Some evaluation methods are presented in Section 2.8 and finally a summary of the chapter is presented in Section 2.9.

2.1 Script Identification

There are many pieces of script identification research for printed and handwritten (offline/online) documents in the literature. The proposed script identification works could be categorized into three types: block/paragraph level, line level, and word level.

Script identification techniques were extensively studied and implemented [27, 28] and the literature in this area is very rich. A few recently developed techniques for off-line [29, 30] and online [31, 32] script identification are discussed here. Abirami and Manjula [28] reviewed recent works on script identification in their paper and Ghosh et al.'s [27] reviewed paper on script identification has covered methods for automatic script identification developed so far.

Spitz [2] described a two-stage approach for automatic identification of the script as well as language from printed document images. First, the script was classified into two initial classes (Han-based or Latin-based) using vertical position distribution of upward concavities information. Distribution of optical density information was used to determine the language of Han script class (Chinese, Japanese and Korean). Latin-based languages were determined using information based on character shape codes.

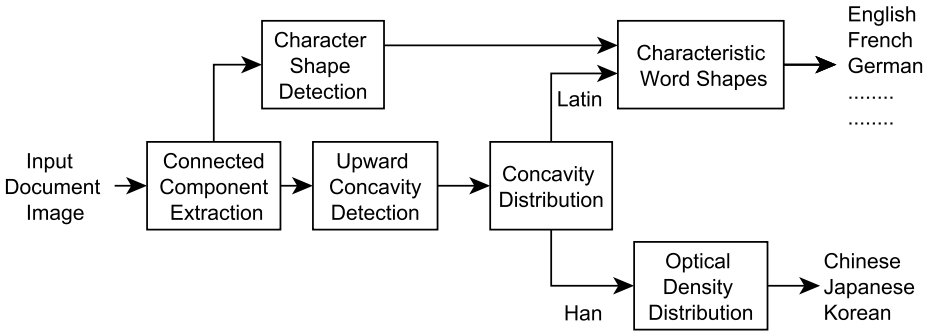


FIGURE 2.1: Spitz's method [2] of script identification.

Connected component-based script and language identification method for handwritten documents was proposed by Hochberg et al. [33]. A set of heuristic features were extracted from each component and linear discriminant analysis was used for classification of each possible pair of scripts in the dataset (Arabic vs. Chinese, Arabic vs. Cyrillic, etc.). Roy and Majumdar [30] illustrated a method for script separation of postal documents using features based on geometric pattern, busy-zone, and topology. The Neural Network classifier was used in this work for classification. Three scripts (Bangla, Devnagari, and English) were considered in their work.

Namboodiri and Jain [31] have used the SVM classifier to classify words and lines in an online handwritten document of different scripts; classification was based on spatial and temporal features of strokes. The features were extracted either from the individual strokes or from a collection of strokes. The extracted features were namely, average stroke length, shirorekha strength, shirorekha confidence, stroke density, aspect ratio,

reverse distance, average horizontal stroke direction and average vertical stroke direction. Schenk et al. [32] described a method for on-line handwritten whiteboard note recognition.

Pal and Chaudhuri proposed an automatic system for the identification of Latin, Chinese, Arabic, Devnagari, and Bangla textlines in printed documents [3]. The headline (shirorekha) information was used first to separate Devnagari and Bangla script lines from Latin, Chinese, and Arabic script lines as illustrated in Fig. 2.2. Next, Bangla script lines were distinguished from Devnagari by observing the presence of certain script specific principal strokes. Similarly, Chinese text lines were identified by checking the existence of characters with four or more vertical runs. Finally, Latin (English) text lines were separated from Arabic using statistical as well as water reservoir-based features. Statistical features include the distribution of lowermost points in the characters the lowermost points of characters in a printed English text line lie only along the baseline and the bottom line while those in Arabic are more randomly distributed. Water reservoir-based features give a measure of the cavity regions in a character.

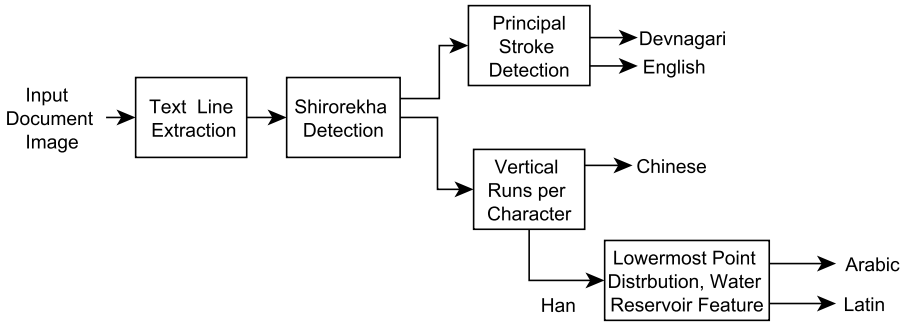


FIGURE 2.2: Pal and Chaudhuri's [3] method of script identification.

In [34] Roy et al. proposed a technique towards the identification of handwritten Roman and Persian script. A set of 12 features based on fractal dimension, the position of a small component, topology etc. were used. Four classifiers (Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Multi-Layer Perceptron (MLP) and Modified Quadratic Discriminant Function (MQDF)) were used for classification task and a comparison analysis was presented in the paper. In another work Roy et al. [35] described an algorithm for script identification at document level from Indian handwritten documents written in six popular scripts. Fractal-based features, component-based feature and topological features were used to extract a total set of 46 features used here. MLP and Neural Network were used for identification of handwritten Bangla, Devnagari, Malayalam, Urdu, Oriya and Roman scripts.

Zhou et al. [36] illustrated a method for identification of Bangla/English scripts. The method was based on connected component profile extracted from a block of handwritten addresses. Singhal et al. [37] proposed a technique for classification of four different scripts (viz. English, Devnagari, Bangla and Telugu). Rotation invariant textures feature using multi-channel Gabor filter was extracted and multi-prototype classifier was used for their classification task. Sarkar et al. [38] have illustrated a script separation technique for Bangla and Devnagari script mixed with Roman script. A set of holistic features was extracted at word level and MLP-based classifier was used in this work.

Rajput and Anitha [39] have described a novel method towards multi- script identification at a block level. The recognition was based on features extracted using Discrete Cosine Transforms (DCT) and wavelets of Daubechies (is a wavelet used to convolve image data) family (i.e., features were extracted by transforming the image in the time domain to the image in the frequency domain). KNN classifier was adopted for recognition purpose and the classifier computes the Euclidean distances between the test feature vectors with that of the stored features and identifies the k-nearest neighbor. Finally, the classifier assigns the test image to a class that has the minimum distance with the voting majority and the corresponding script was declared as recognized script.

2.2 Signature Detection and Segmentation

Overlapping and touching printed text separation from handwriting/signature is a challenging problem in document processing. Although, there are a few works on handwritten annotation separation from printed documents, not many studies are found for signature segmentation from such documents. Since signatures are also handwritten, research on handwriting text identification is also discussed here. An earlier work by Kuhnuke et al. [40] illustrated a method for machine written and handwritten character distinction and used straightness of vertical and horizontal lines and symmetry relative to the centre of gravity. A feed forward Neural Network was used as classifier in their work. Djeziri et al. [13] proposed an approach to extract signatures from check backgrounds. This approach was inspired by human visual perception and was based on filiformity criteria whose specialized task was to extract lines. Based on filiformity measure contour lines of objects were differentiated from handwritten lines. Pal and Chaudhuri [41] proposed an algorithm for a line-wise printed and handwritten text separation scheme for Bangla and Devnagari scripts. Script-based characteristics and regularities using structural and statistical feature set were used in their work. This method was script-dependent and thus may not work on handwritten annotation due to the line-based features used.

Guo and Ma [42] used Hidden Markov Models (HMM)-based classification for handwritten annotation separation from a printed document on word level. Vertical profile projection-based features were used for recognition and classification in this work. Zheng et al. [43] described an algorithm for segmentation and identification of handwriting in noisy document image using structural and texture features like Bi-level Co-occurrence, Bi-level 2x2-grams, Pseudo Run Lengths and Gabor Filters. Fisher classifier was used to classify handwritten and printed text. However, overlapping of handwritten text on printed words was not considered in this work. To segment signatures from bank cheques and other documents Madasu et al. [44] proposed an approach based on a sliding window approach to calculate the entropy and finally fit the window to signature block. A major deficiency of this technique, however, is that a priori information about the location of the signature was assumed. Jang et al. [45] proposed a method for separation of printed and handwritten addresses on Korean mail images using geometric features-based classification. A multi-layer perceptron network was used as a classifier in their work.

Farooq et al. [46] proposed directional Gabor filters for word level feature extraction and an Expectation Maximization (EM)-based probabilistic Neural Network was used for Arabic script classification containing printed and handwritten text. The performance of the system was evaluated using Neural Network and SVM. Peng et al. [47] used a modified K-Means clustering algorithm at an initial stage for classification of handwritten and printed text and then Markov Random Field (MRF) was used for relabelling. Structural and statistical features based on background and foreground information, connected component-based features and Gabor features were used. In another work [48], overlapped texts were segmented by shape context-based aggregation and MRF. Zhu et al. [12] described a multi-scale structural saliency approach to capture the dynamic curvature using a signature production model for signature detection and segmentation. Authors claim the saliency measure based on a signature production model effectively quantifies the dynamic curvature of 2-D contour fragments. The Tobacco-800 database and the University of Maryland Arabic database were used in their experiment. However, this method was not focused towards segmenting characters which were overlapping/touching with the signature.

Srinivasan and Srihari [49] proposed a method for signature-based retrieval of scanned documents. A model based on Conditional Random Fields (CRF) was used to label extracted segments of scanned documents as machine-printed, signature and noise. Next, a technique based on Support Vector Machine (SVM) was used to remove noise and printed text overlapping the signature images. Finally, a global shape-based feature was computed for each signature image. In a recent work Peng et al. [50] illustrate a modified tree-structured multi-class classifier to identify annotation and overlapping

text from machine printed documents. To overcome the over-fitting problem of a tree-structured classifier, a boosting algorithm was introduced which was able to take the advantages of the classifier and can handle the overfitting problem. Mazzei et al. [51] proposed a method for classification of handwriting annotation using pixel clustering techniques. DBSCAN (Density-Based Spatial Clustering of Application with Noise) clustering algorithm and decision tree classifier were used for classification of annotation into categories like underlined elements, symbol or short notes in between lines or over the text and notes on the blank portions. To the best of the researcher's knowledge, however, substantial work on signature segmentation has not been done yet to segment the signatures that were overlapped or touched with printed text, logo, seal, etc.

Kumar et al. [52] applied multiple instance learning-based method for retrieval and localization of signatures and retrieval of machine printed documents. The document images were segmented into different regions using a Voronoi-based segmentation. A descriptor for each zone was computed for statistics of the frequency of each Three-Adjacent-Segment (TAS) feature occurrence. Finally, a single feature vector was obtained for each zone. Ahmed et al. [14] proposed a Speeded Up Robust Features (SURF)-based approach for signature segmentation from document images. As SURF is a part-based approach that represents image as a set of key points; the proposed approach extracts key points from the images. Next, a 128 bit descriptor was extracted from each key point and used to find the similarity between different key points. All the points having Hessian threshold less than 400 were neglected to filter the unimportant features from the images. The Euclidean distance metric was used as a distance measure. Finally, a majority voting-based approach was applied for the classification of the connected component. The method was evaluated on the publicly available Tobbaco-800 [1] dataset.

The following section discusses studies with similar objectives to those of this study, but which have used other content such as logos, or text, for example, instead of signatures for retrieval of documents. A content-based retrieval algorithm based on an hierarchical matching tree was proposed by Dewan et al. [53]. Hough transform-based feature descriptors were extracted from paragraphs and line blocks and based on these descriptors, documents were indexed. The similarity of two images was defined by the Euclidean distance between document feature points in space. Wang [54] proposed an algorithm for logo detection and recognition using a Bayesian model. A multi-level step-by-step approach was used for recognition of logos and the logo matching process involved a logo database. Here, a region adjacency graph (RAG) was used for representing logos, which models the topological relations between the regions. Finally, Bayesian belief networks were employed as well in a logo detection and recognition framework. Recently, Alaei and Delalandre [55] proposed a system for detection and recognition

from document images. A Piece-wise Painting Algorithm (PPA) and some probability features along with a decision tree were used for logo detection and a template-based recognition approach was proposed to recognize the logo.

2.3 Signature Recognition

Ozgunduz et al. [56] described a technique for off-line signature verification and recognition using global, directional and grid features of signatures. Global features like signature area, signature height-to-width ratio, maximum horizontal histogram, maximum vertical histogram, local maxima numbers of signature, edge point numbers of signature etc. were used. 8 different 3×3 mask directional features were also used in their work. As the recognition of signature represents a multiclass problem SVM's one-against-all method was used for signature recognition task. Artificial Neural Network's (ANN) back propagation-based classification was also performed on the dataset for comparison purposes.

Chalechale and Martins [57] proposed a method for Persian signature recognition based on line segment distribution of sketches. Chain codes information was used from thinned images for line segment detection. The chain codes were then converted into several micro chains by employing the extreme points of the first derivative of shifted smoothed chain code function. Next, micro chains were approximated by straight line segments. Length and position distributions of line segments were combined to make a compact feature vector. The feature vector was used for retrieving Persian signatures in a hypothetical paperless office.

In paper [58] Roy et al. presented a signature-based document retrieval technique from documents with a cluttered background. Here, a signature object was characterized by spatial features computed from recognition result of background blobs. The background blobs were computed by analysing character holes and water reservoir zones in different directions. Zernike Moment features were extracted from each blob and a K-Mean clustering algorithm was used to create the codebook of blobs. During retrieval, Generalized Hough Transform (GHT) was used to detect the query signature and a voting was cast to find possible location of the query signature in a document. The spatial features computed from background blobs found in the target document were used for GHT. The peak of votes in GHT accumulator validates the hypothesis of the query signature. The proposed method was tested on a collection of mixed documents (handwritten/printed) of various scripts.

Oz [59] developed a method for signature verification and recognition was based on moment invariant features and Artificial Neural Network (ANN). It uses a four-step process: separates the signature from its background, normalizes and digitizes the signature, applies moment invariant vectors and finally implements signature recognition and verification. Two sequential neural networks were designed: one for signature recognition and another for verification (i.e. for detecting fraud). Verification network parameters change depending on recognition process results. There was a selection mechanism which determines verification network parameters for recognized signatures.

In another paper, Odeh and Khalil [60] described a signature verification and recognition method based on Neural Network. Four main features were extracted for signature recognition and verification, which are eccentricity, skewness, kurtosis, and orientation. Multi-layer Perceptron Neural Network was used for a signature recognition task in this work. Chalechale et al. [61] presented a method for document image decomposition and retrieval based on connected component analysis and geometric properties of the labelled regions. Documents having Arabian/Persian signature were considered for the experiment. First, the signature regions were detected and features were extracted from the detected regions. A novel angular-radial partitioning (ARP) method was used for verification. The ARP method is based on accumulating of the signature pixels in the sectors defined adjacently in the signature region. The magnitude of the Fourier transform in order to achieve rotation invariance was used.

Kudlacik and Porwik [62] presented a fuzzy-based approach for off-line handwritten signature recognition. First, the center of gravity was extracted from the signature image. After finding signature's center of gravity, a number of lines were drawn through it at different angles. Cross points of generated lines and the signature sample, which were further grouped and sorted, were treated as the set of features. On the basis of such structures, obtained from a chosen number of learning samples, a fuzzy model was created, called the fuzzy signature. During a verification phase, the level of conformity of an input sample and the fuzzy signature was calculated.

2.4 Word Spotting

Significant work has been undertaken [19–21] in the area of word spotting to make the handwritten text available for searching and browsing. Word spotting is becoming popular in such fields because of the low computational cost in comparison to transcription of the entire text. Rath and Manmatha [19] proposed a classical method for word spotting using profile-based features and Dynamic Time Warping (DTW). A Recurrent Neural Network-based approach was described in [21] to make handwritten documents available

for word-based searching and indexing. The Neural Networks and CTC Token Passing algorithms were used for this word spotting task.

Hidden Markov Model-based methods are extensively used for modelling handwritten text, word spotting etc. In [20], Fischer et al. proposed a learning-based word spotting system that uses HMM sub-word models to spot keywords. The proposed lexicon-free approach can spot arbitrary keywords from the handwritten text. In [63], an HMM-based method was applied for word spotting from handwritten documents. Local Gradient Histogram (LGH) features were used in this work.Tan et al. [64] discussed a method which uses 3 different codes to encode the extreme position of the vertical bar pattern of a word and depending on if they made a feature vector for matching purpose. Lu et al. [65] proposed a technique to retrieve document images by a word shape coding scheme based on topological shape features including character ascenders/descenders, character holes, character water reservoirs etc. Li et al. [66] described a method where used features were convex hull and centroid-based. Distance ratio was also computed between centroid position and intersection points.

A few other methods have recently been proposed towards word spotting in [67–69] using structural and statistical features. Lu and Tan [68] proposed a method for word spotting in Chinese documents using used stroke density of grid feature and matching algorithm based on Weighted Hausdorff Distance measure. In [70] a method towards word spotting using statistical features was proposed by Konidaris et al. Terasawa and Tanakav [69] used a modified HOG feature for a word spotting task. Slit style HOG feature was a gradient distribution-based feature with overlapping normalization and redundant expression. It was reported that the algorithm will work for all script regardless. The algorithm was tested on samples of Japanese documents. Tarafdar et al. [71] described a method for word spotting from printed documents used the position of extreme points, vertical bar position, crossing count, and loop and background information of words as features. Longest Common Subsequences (LCS) and edit-distance value-based string matching technique was used for the similarity matching of words.

2.5 Numerical Field Extraction

Few research works have been published for automatic form field (e.g. numerical sequence) extraction from handwritten documents [10, 72–74]. Recently, field-based information retrieval has gained more popularity than recognition of full handwriting document. Koch et al. [72] proposed a method for automatic extraction of numerical fields from handwritten documents. The approach exploits the known syntactic structure of the numerical field to extract, combined with a set of contextual morphological features to find the best label of each connected component. The HMM-based syntactic

analyser on the overall document was employed to localize/extract fields of interest. The localisation of numerical fields was performed using the Viterbi algorithm.

Chatelain et al. [10] proposed a method for automatic extraction of numerical fields from handwritten incoming mail documents. Different feature sets such as contextual/morphological, chain code and statistical/structural were extracted. A multi-layer perceptron was trained over each feature set and measurement level combination of classifiers was achieved. The main objective of this work was to localize the fields of interest by rejecting the major part of the document. The false alarm rate was slightly decreased by the use of combination of classifiers. Chatelain et al. [73] described a method for automatic extraction of numerical fields (ZIP codes, phone numbers, etc.) from incoming mail documents based on a segmentation-driven recognition that aims at locating isolated and touching digits among the textual information. A syntactical analysis was then performed on each line of text in order to filter the sequences that represent a particular syntax known by the system. A two-stage recognition method was proposed in this work. First, a segmentation-driven recognition was performed in order to identify the numeral components such as isolated or touching digit. The second stage of the component recognition system was dedicated to the reject and separator identification among the digit recognition trellis.

Thomas et al. [74] proposed an HMM-based classification model for alpha-numerical sequence extraction. The handwriting modelling is text line oriented and HMM-based technique was used as probabilistic tools in handwriting sequence modelling [75]. Here, the global line model is based on HMMs, which was employed for dual representation of the relevant and the irrelevant information. The shallow parsing of text lines allows the fast extraction of information. To extract the desired numerical sequence, a syntactical analysis was performed on each line of text. Most of the papers mentioned here deal with alpha-numeric string extraction. The reported results show that spotting-based approaches in handwritten documents outperformed OCR-based systems in terms of accuracy and time complexity for such information extraction task.

2.6 Date-Field Extraction

Handwritten date information processing from scanned documents still remains very challenging and it is hard to propose an algorithmic method for automatic segmentation and recognition. Date pattern detection and interpretation in handwritten documents is challenging due to the unconstrained nature of handwriting of different individuals, touching numerals, different patterns of a single date etc. Thus, there are very few published research works available on automatic segmentation and recognition of date-field.

Suen et al. [76] focus on one of the most challenging parts of a cheque recognition system, the segmentation and recognition of the date written on the bank cheques. First, the method segments a date image by using the separator and two separators were used to detect the 'Year', 'Day' and 'Month' zones based on shape and spatial features. Next, numeric and non-numeric month fields were recognised by a connected digit recogniser and a cursive word recognizer. The recognition results were finally sent to a parser, which was used to interpret acceptable results and reject invalid ones. Experiments in their study were conducted on a sample data set of cheques with diverse colour designs and languages.

Xu et al. [77] described a method knowledge-based segmentation system for handwritten dates on bank cheques. The knowledge derived from the writing style information, syntactic and semantic constraints was utilized, and different knowledge sources were adopted at different stages. In order to improve the performance and efficiency of the system, date image segmentation was divided into two stages. In the Segmentation and Multi-hypotheses Generation stage, structural features and some contextual knowledge about writing styles (encoded in a pattern based grammar) were used in a knowledge-based module to solve most segmentation cases. Ambiguous cases were handled by multi-hypotheses generation module at this stage, for a final decision to be made when more contextual information and syntactic and semantic knowledge are available (i.e., multi-hypotheses evaluation was made in the recognition stage).

Morita et al. [4] describes a system to recognize handwritten dates based on HMM-MLP hybrid approach. The block diagram of the proposed system is shown in Fig. 2.3. The system first segments implicitly a date image into sub-fields through the recognition process based on an HMM-based approach. Next, a recognition and verification strategy was proposed to recognize the three obligatory date sub-fields (day, month and year) using different classifiers. Markovian and neural approaches were adopted to recognize and verify words and strings of digits respectively. A concept based on meta-classes of digits was introduced to reduce the lexicon size of the day and year and improve the precision of their segmentation and recognition.

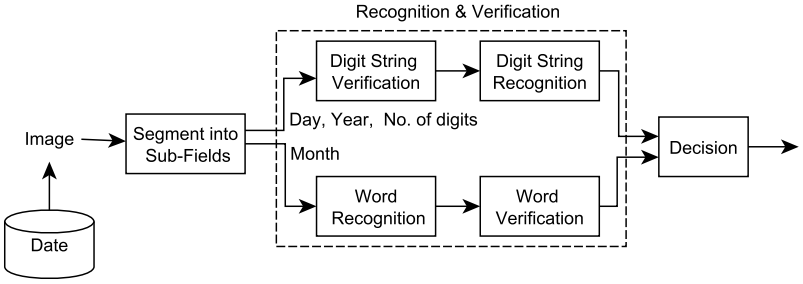


FIGURE 2.3: Date recognition system proposed by Morita et al. [4].

Xu et al. [78] describes a system for automatic segmentation and recognition system for handwritten dates on Canadian bank cheques. A segmentation based strategy was adopted in this system. In order to achieve high performances in terms of efficiency and reliability, a knowledge-based module was proposed for the date segmentation and a cursive month word recognition module was implemented based on a combination of classifiers.

The proposed work towards date interpretation in this thesis deals with the multi-script scanned documents. Alpha-numeric characters of three languages (Bangla, Devnagari, and English) were labelled to locate the date-fields in the Indian multi-lingual documents. Recently, a method for date-field extraction from handwritten English documents [**?**] was proposed. The extended version of date-field extraction method deals with date extraction from multi-script (English, Devnagari, and Bangla) handwritten Indian documents. To the best of this researcher's knowledge, there is no work available on date extraction in printed/handwritten multi-lingual documents.

2.7 Datasets

There are no standard public datasets for experimentation on document retrieval based on signature and date information. The Tobacco-800 [1] is a publicly available administrative document image collection which contains signatures and logo information. This dataset has been used for many signature-based document processing experiments. The SIG-DS-I and SIG-DS-II [5] are two publicly available offline signature datasets for signature matching/verification. The GDPS-960 [6] is a signature dataset used for signature recognition and verification for many experiments. Few samples of signatures are presented in Fig. 2.4 from the SIG-DS-I, SIG-DS-II, and GDPS-960 datasets.

An English digits dataset MNIST [7] (see Fig. 2.5) has been used in experiments related to digits recognition. The IAM-database [8] is an English sentence database which has been used for offline handwritten recognition. Fig. 2.6 shows some sample lines from IAM English handwritten dataset.

2.8 Performance Evaluation Techniques

The proper evaluation of any system is a crucial part to estimate the performance. The primary purpose of an evaluation is to show how effectively a system works in a particular environment. The most widely used performance evaluation measures are precision(P) and recall (R) [79] for any retrieval system. In document retrieval systems,

FIGURE 2.4: Samples of signature from (a,b) SIG-DS-I and (c,d) SIG-DS-II [5] datasets. (e-h) Samples of signature from GPDS [6] signature dataset.

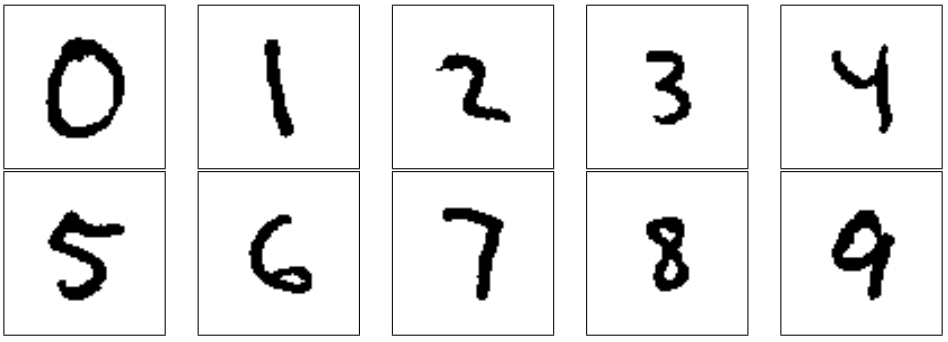
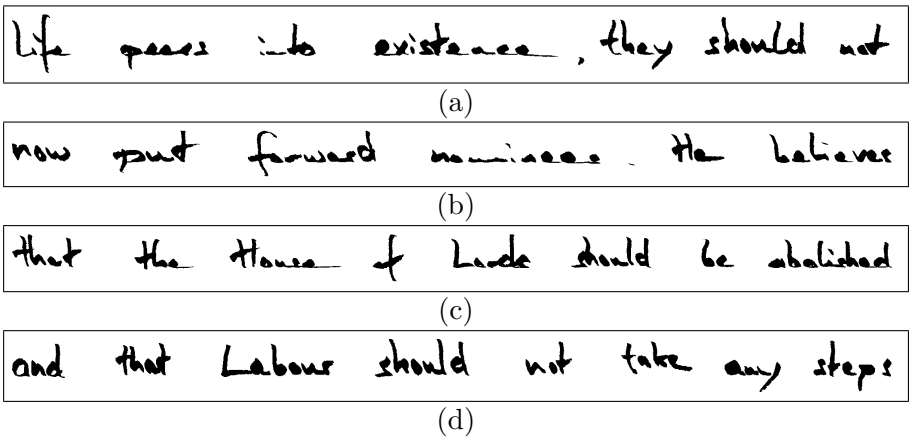


FIGURE 2.5: Samples of numerals from MNIST [7] digits dataset.

precision and recall are defined in terms of a set of retrieved documents and a set of relevant documents against a query.

Precision (P) is defined as the number of true positives (T_p) over the number of true positives plus the number of false positives (F_p), whereas Recall (R) is defined as the



(a) (b) (c) (d)

FIGURE 2.6: Samples of handwritten lines from IAM English handwriting dataset [8].

number of true positives (T_p) over the number of true positives plus the number of false negatives (F_n).

$$Precision\ (P) = \frac{T_p}{T_p + F_p} \tag{2.1}$$

$$Recall(R) = \frac{T_p}{T_p + F_n} \tag{2.2}$$

The harmonic mean of precision and recall known as F_1 score is defined as follows.

$$F_1 = \frac{P \times R}{P + R} \tag{2.3}$$

Receiver operating characteristic (ROC) curve is a graphical plot that is widely used [15] as well for reporting performance of the classification systems. The curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.

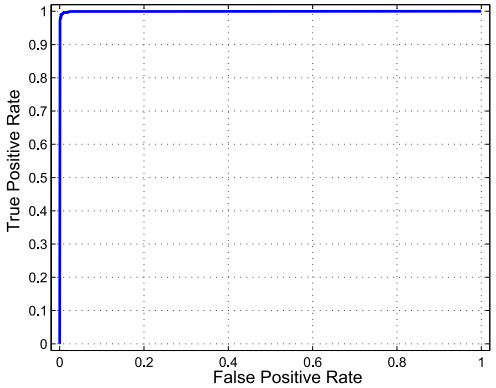


FIGURE 2.7: Sample ROC curve.

2.9 Summary

In this chapter a brief review of the techniques proposed by researchers towards signature and date information-based document processing are presented. The literature review reveals that an adequate volume of work has not been undertaken for document retrieval using signature and date information. However, there have been recent advances towards offline and online signature verification/authentication and a few established methods are available. Detection of signatures from a document with diverse layout and structures is a crucial task but is still an open research problem. It is also observed that robust techniques for repesenting signature shape used for signature recognition is still in its infancy. A significant number of works on script-identification, word-spotting and numeral recognition have also been found. However, not much work has been undertaken towards date processing and date-based document retrieval. Very few datasets specific to document retrieval are available. The literature review shows that researchers have used different datasets to evaluate their techniques, which has made it difficult to perform a comparative study of the results.

Chapter 3
Signature-Based Document Retrieval

The reviewed literature in the previous chapter reveals that research on document image retrieval using signature information has not yet been fully explored. It also shows that the research area consists of different challenges. In this context, some works were reported on content-based retrieval using a logo, seal, word image, etc., as a key element. But, document retrieval using signature information could also be a useful approach for searching/indexing of document images. This indicates that there is a huge amount of research scope available for developing such a document retrieval system. Moreover, processing of multi-script/multi-lingual documents has not been explored for document retrieval; hence, this research study will examine its significance and implementation. A multi-stage approach for document retrieval based on signature information is proposed in this study. The details of the experimental results obtained from the investigation of the proposed technique are presented as well.

This chapter is organized into different sections as follows. In Section 3.1 presents the introduction of signature-based document retrieval. Section 3.2 is devoted to the description of the proposed system. The proposed feature extraction and classification techniques are described in Section 3.3. Section 3.4 deals with the signature detection in a document. Section 3.5 describes the algorithm of signature segmentation. Section 3.6 describes the signature shape encoding and matching techniques and Section 3.7 presents the experimental results. Finally, 3.8 presents the summary of this chapter.

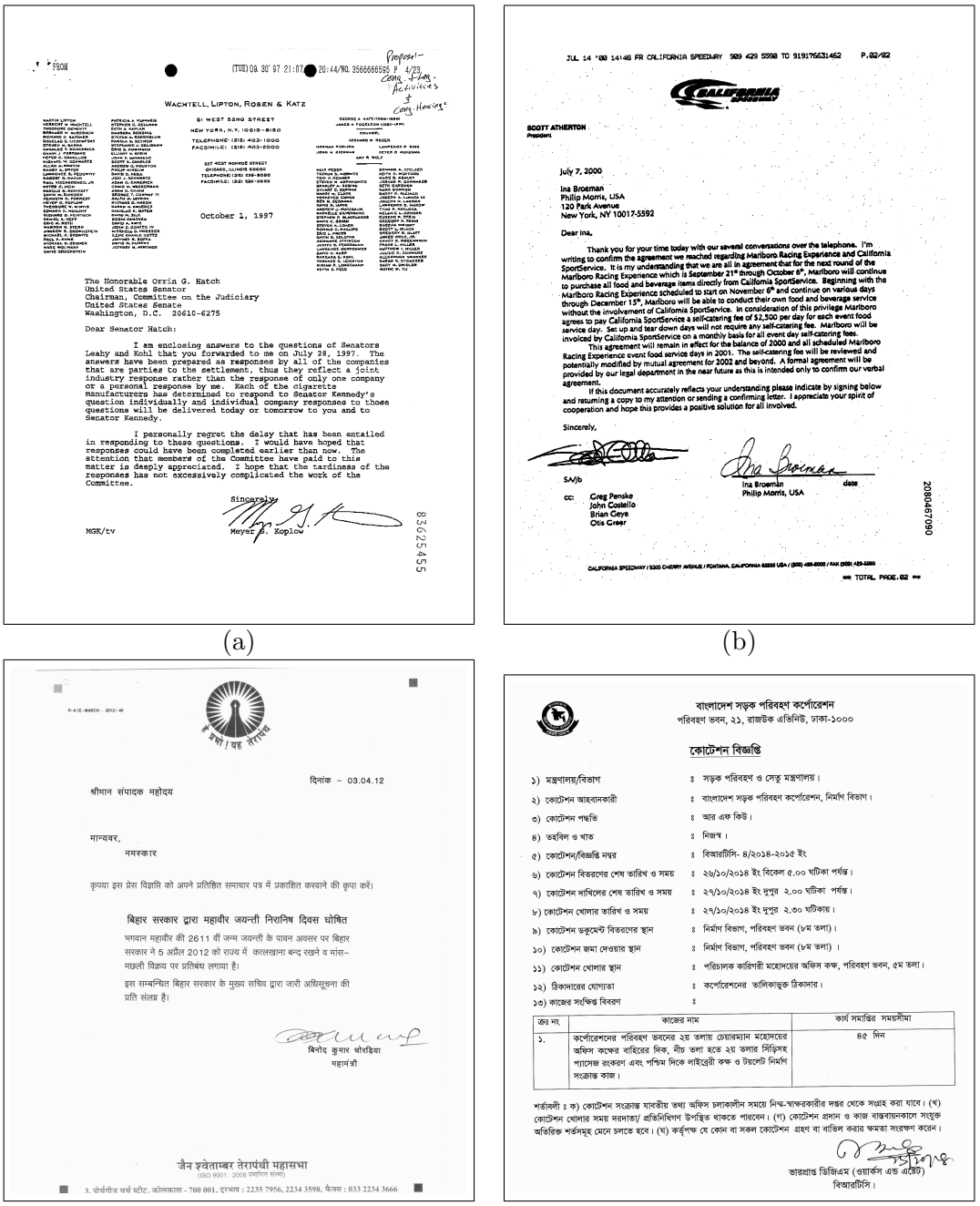
Major parts of this chapter are published in the paper titled '*Signature Segmentation from Machine Printed Documents Using Contextual Information*', Mandal et al. [80].

3.1 Introduction

Automatic retrieval of documents from a large heterogeneous document image repository [1] based on signature information is a crucial problem in document image retrieval. Multiple stages such as signature detection, signature segmentation, and signature matching are involved in developing a signature-based document retrieval system. Signature detection in a document is challenging in many aspects. The unconstraint nature of signature, interclass variance, and complex document background are just a few of these challenges. Automatic signature detection and segmentation is also a challenging task due to its touching/overlapping strokes with other text or graphical components of the document. Often, a signature contains text characters as well as some graphical entities for beautification purposes. Thus, the segmentation of signature from the background is difficult as well as interesting due to unconventional signing style and appearance of background text information over the signature. There are many research studies which deal with automatic online/offline signature verification and recognition [81]. However, these approaches use only isolated signatures and cannot be applied in the context of documents with complex background. The signature matching task is an equally important stage in the system and has challenges for the reason of interclass variance. Four sample documents containing signatures as shown in Fig. 3.1 were used in the experiments.

3.2 The Proposed System

Automatic signature detection and segmentation are the initial stages of a signature-based document image retrieval system. The proposed system has three main stages main stages: detection, segmentation, and matching of signatures. A flow diagram of the proposed system is given in Fig. 3.2 and shows the modules involved in the system. In the first stage, a component-based classification technique detects the potential signature components from other text and graphics components of a document. A signature may contain more than one component and in the second stage, the components are grouped to find the signature regions. A density-based clustering algorithm (DBSCAN [82]) is proposed for grouping the signature components in this stage. In the third stage, the segmented signature shape is characterized for matching with a query signature. The signature object is characterized here by spatial features from foreground strokes, background loops and background reservoirs. Finally, three distance measures are considered between the query signature and the signature in the target documents to retrieve the documents relevant to the query.



(a) (b)

(c) (d)

FIGURE 3.1: Signed printed documents of different scripts are shown here. (a,b) Samples of printed signed English documents from the 'Tobacco' [1] dataset. (c) A letter printed in Devnagari script and (d) an official notice printed in Bangla script.

3.3 Feature Extraction and Classification Techniques

In this section a brief description of the various features extraction and classification techniques used in the proposed method are detailed. The feature extraction techniques namely, 400-dimensional gradient features, Gabor-filter based features, Zernike moment-based features, and Bag-of-Visual-Words (BoVW) with SIFT descriptors are discussed.

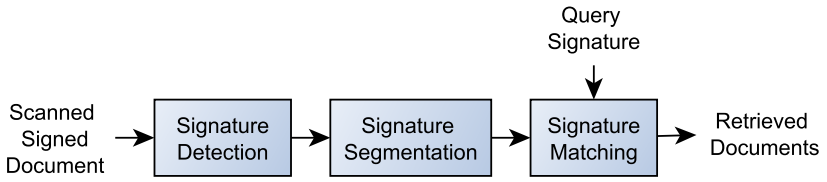


FIGURE 3.2: A block diagram of the proposed document retrieval system using signature information as a query.

Support Vector Machine (SVM) is considered for the classification task and is also discussed in this section.

3.3.1 400-Dimensional Gradient-Based Feature

The gray-scale local-orientation histogram of the component is used for 400 dimensional feature extractions. The feature is extensively used by the researcher [83] and hence this feature was used also in the experiment. To obtain 400 dimensional features the following steps are performed.

- First, size normalization of the input binary image is done. The image is normalized into 126×126 pixels based on the empirical analysis.
- The input binary image is then converted into a gray-scale image by applying a 2×2 mean filtering 5 times.
- The gray-scale image is normalized next so that the mean gray-scale becomes zero with maximum value 1.
- Next, the normalized image is segmented into 9×9 blocks.
- The Roberts filter is then applied on the image to obtain the gradient image. The arctangent of the gradient (strength of gradient) is quantized into 16 directions (an interval of 22.5°) and the strength of the gradient is accumulated with each of the quantized direction. By the strength of gradient, $f(x, y)$ means

$$f(x, y) = \sqrt{(\Delta u)^2 + (\Delta v)^2} \tag{3.1}$$

and by direction of gradient ($\theta(x, y)$) means

$$(\theta(x, y)) = \tan^{-1}(\Delta u / \Delta v) \tag{3.2}$$

where,

$$\Delta u = g(x+1, y+1) - g(x, y),$$

$$\Delta v = g(x+1, y) - g(x, y+1)$$

and $g(x, y)$ is a gray-scale value at an (x,y) point.

- Histograms of the values of 16 quantized directions (with an interval of 22.5°) are computed in each of 9×9 blocks.
- Finally, 9×9 blocks are down sampled into 5×5 by a Gaussian filter [83]. Thus, $5 \times 5 \times 16 = 400$ dimensional features are obtained.

3.3.2 Gabor Filter-Based Feature

Gabor filters [84] are capable of representing signals in both the frequency and the time domains. The Gabor filter is the product of a sinusoid and a Gaussian:

$$g(x, y; \lambda, \sigma, \theta, \phi, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \phi\right) \tag{3.3}$$

where,

$$x' = x\cos\theta + y\sin\theta,$$

$$y' = -x\sin\theta + y\cos\theta$$

.

In this equation, λ represents the wavelength of the cosine factor, θ represents the orientation of the normal to the parallel stripes of a Gabor function, ψ is the phase offset, σ is the sigma of the Gaussian envelope and γ is the spatial aspect ratio, and specifies the ellipticity of the support of the Gabor function.

To extract the Gabor filter-based feature, the image is normalized into 20×20 dimension. Next, on the normalized image Gabor filter is applied at four orientations (0°,45°,90° and 135°). Then the moment is computed by using an average of these four Gabor orientations matrixes.

3.3.3 Zernike Moments-Based Feature

Zernike Moment is a class of orthogonal moment and has been shown to be effective in terms of image representation. Here, the definition of Zernike moments is briefly summarised. Zernike moments are based on a set of complex polynomials that form a complete orthogonal set over the interior of the unit circle [85]. Zernike moments are

defined to be the projection of the image function on these orthogonal basis functions. The basis functions $V_{n,m}(x, y)$ are given by

$$V_{n,m}(x, y) = V_{n,m}(\rho, \theta) = R_{n,m}(\rho)e^{jm\theta} \tag{3.4}$$

where, n is a non-negative integer, m is non-zero integer subject to the constraints $n - |m|$ is even and $|m| < n$, ρ is the length of the vector from origin to (x, y), θ is the angle between vector ρ and the x-axis in a counter clockwise direction and $R_{n,m}(\rho)$ is the Zernike radial polynomial. The basis functions in equation 2 are orthogonal. The Zernike moment of order n with repetition m for a digital image function $f(x, y)$ is given by

$$Z_{n,m} = \frac{n+1}{\pi} \sum \sum_{x^2+y^2 \leq 1} f(x, y)V^*_{n,m}(x, y) \tag{3.5}$$

where, $V^*_{n,m}(x, y)$ is the complex conjugate of $V_{n,m}(x, y)$. To compute the Zernike moments of a given image, the image center of mass is taken to be the origin. A 72 dimensional Zernike Moments feature vector was computed for the experiment.

3.3.4 Bag-of-Visual-Words (BoVW) with SIFT Descriptors

The feature extraction module described here has three components. A flow diagram of feature extraction technique from an image is presented in Fig. 3.3. First, SIFT descriptors are extracted from the image. An image is divided into grids $(n \times n)$ to obtain $(n \times n)$ grids and from each grid one SIFT-descriptor is computed. Next, all the computed descriptors are used in the vector quantization step. The K-means clustering algorithm is used for vector quantization to create a visual codebook. Finally, the Spatial Pyramid Matching (SPM)-based scheme is applied for the final representation of an image. The general idea of the computation of SIFT-descriptors and the SPM employed in this work are described below in Section 3.3.4.1 and Section 3.3.4.2, respectively.

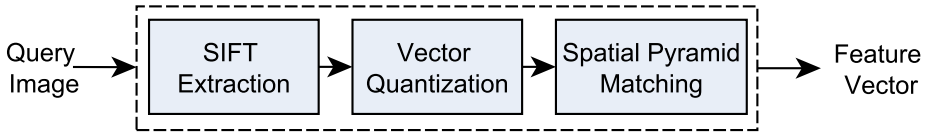
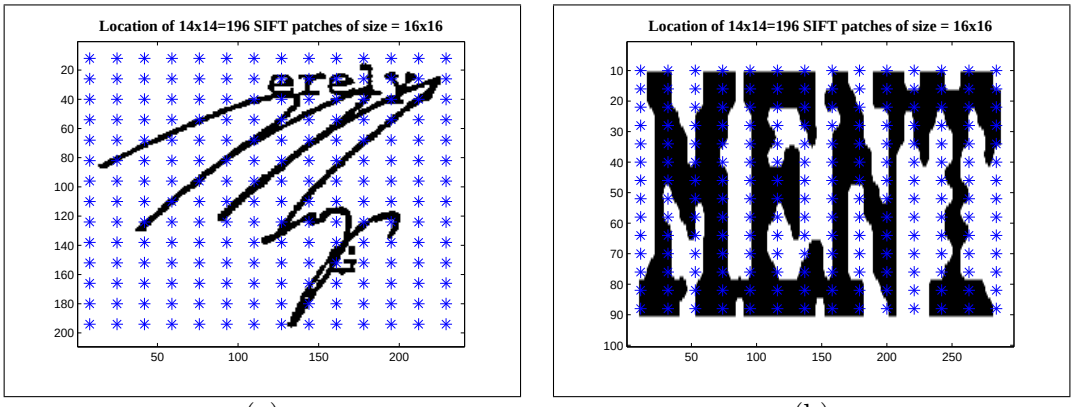


FIGURE 3.3: Flow diagram of the feature extraction module using Bag-of-Visual-Words with SIFT-descriptors.

3.3.4.1 SIFT Descriptor

The SIFT (Scale-Invariant Feature Transform) [86] is a local shape descriptor to characterize local gradient information. Here, a 128-dimensional vector for each key point is extracted which stores the gradients of 4×4 locations around a pixel in a histogram bin of 8 directions. The SIFT descriptor is scale and rotation invariant. The gradients are aligned to the main direction, which makes it a rotation invariant descriptor. Different Gaussian scale spaces are considered for the computation of a vector to make it scale invariant. The blue asterisk symbols in Fig. 3.4(a), 3.4(b) represent the 14×14 SIFT patches of a signature stroke and a printed word, respectively.



(a) (b)

FIGURE 3.4: Blue asterisk symbols represent 196 (of 14x14) SIFT patches of (a) Signature component (b) Printed component.

3.3.4.2 Spatial Pyramid Matching (SPM)

The SPM is an extended version of the Bag-of-Features (BoF) model, which is simple and computationally efficient. As the BoF model discards the spatial order of local descriptors, it restricts the descriptive power of the image representation. The limitation of BoF is overcome by the SPM [87] approach, which is successfully applied on image categorization tasks. An image is partitioned into $2^l \times 2^l$ segments where $l = 0, 1, 2, 3, ..., n$; represents different resolutions. Next, the BoF histograms are computed within each of the 2^l segments, and finally, all the histograms are concatenated to form a vector representation of the image. SPM is equivalent to BoF, when the value of the scale $l = 0$. Here, pyramid matching is performed in two-dimensional image space and uses a traditional clustering technique in feature space. The number of matches at level l is given by the histogram intersection function:

$$I(H_X, H_Y) = \sum_{i=1}^{D} min(H_X(i), H_Y(i)) \tag{3.6}$$

Finally, the representation of the image for classification is the total number of matches from all the histograms, which is given by the definition of a pyramid match kernel:

$$K_\Delta(\Psi(X), \Psi(Y)) = \sum_{i=0}^{L} \frac{1}{2^i} N_i \tag{3.7}$$

where, N_i is the number of newly matched pairs at level i and the value is determined by subtracting the number of matches at the previous level from the current level.

$$N_i = I(H_i(X), H_i(Y)) - I(H_{i-1}(X), H_{i-1}(Y)) \tag{3.8}$$

3.3.5 Classification Technique: Support Vector Machine (SVM)

SVM is a popular classification technique which can successfully be applied to a wide range of applications [88]. The Support Vector Machine (SVM) is used as one of the classification technique for block, word and character level classification in this experiment. The SVM is defined for two-class problems and it looks for the optimal hyper plane which maximizes the distance, the margin, between the nearest examples of both classes, named support vectors (SVs). Given a training database of M data: $x_m | m = 1, ..., M$, the linear SVM classifier is then defined as: $f(x) = \sum_j \alpha_j x_j + b$ where x_j are the set of support vectors and the parameters α_j and b are determined by solving a quadratic problem [88]. The linear SVM can be extended to various non-linear variants; details can be found in [88, 89]. The Gaussian kernel SVM outperformed other non-linear SVM kernels in this experiments, and hence the results reported are based on the Gaussian kernel only. The Gaussian kernel is of the form:

$$k(x, y) = e^{-\gamma \left\| x - y \right\|^2} \tag{3.9}$$

The SVM provides a class label along with its recognition confidence as weight for a fed feature. The value of the weight lies between 0 and 1. The combined information based on class level and recognition confidence are used in the experiments.

3.4 Signature Detection

Signature detection from a document is challenging as a signatory marks their signatures in different styles. A signature generally consists of some large strokes in comparison to the strokes of the printed text. This distinct feature of a signature was used to discriminate signature from printed text. Different feature extraction techniques based on structural, statistical features [40, 41, 43], Gabor [46, 50] and geometric features [45] and various classification model such as HMM [42], K-Means clustering [47], Neural Network [40] and Multi-layer Perceptron (MLP) Network [45] were reported in some of the published work on handwritten and printed text separation. In the proposed work, two different approaches were developed for the signature detection. Different features were computed and a SVM-based classification has been done. In the first approach, two different features namely, Texture (e.g. Zernike Moment-based and Gabor Filter) and Gradient (400 Dimensional feature) were analyzed. This analysis as well as literature review reveals that the gradient-based features performed better than texture features and were applied successfully in many works for signature detection. Hence, gradient-based features were used in the proposed approach for the classification of potential signature and printed text blocks using Support Vector Machine (SVM). In the second approach, Bag-Of-Visual-Words (BoVW)-based technique powered by SIFT descriptors with SVM-based classifier were considered for signature detection. Considering the various issues mentioned in Section 3.1 for signature detection, the proposed method attempts to address them in this section.

3.4.1 Signature Detection using Gradient-Based Features

The proposed work focuses on the detection of a signature from a printed document by eliminating the touching/overlapping characters that are often found on a signature portion of a document and hence, a two-step approach was proposed. A block diagram of the proposed approach is shown in Fig. 3.5. In the first stage, the binarized image was segmented into blocks using a morphological operation. Then, block level feature analysis (module T1 in Fig. 3.5) was performed to discriminate the potential signature blocks from the printed words/text blocks. The 400-dimensional gradient-based features 3.3.1 were used with Support Vector Machines (SVMs) as a classifier for this work (sub-modules C1 and C2 in Fig. 3.5).

A signature block may contain isolated as well as touching printed characters because of the overlapping of signature with printed text. Hence, in the second stage, printed characters, that may be present in isolated form or overlapped/touched with

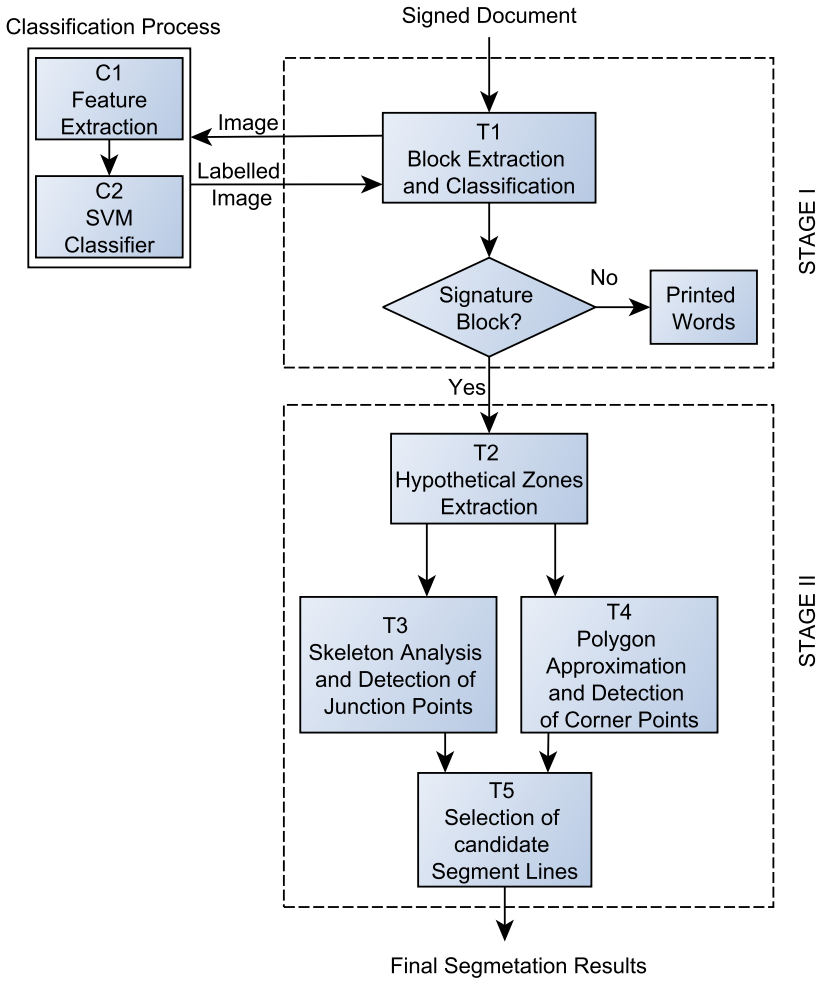


FIGURE 3.5: Block diagram of the proposed signature detection module. C1 and C2 are sub-processes of the classification module. T1 refers to the process of the word level classification module. T2-T5 are sub-processes for selecting candidate segmentation lines which are used for separation of printed text from signature strokes.

signature parts, were removed from signature blocks. To remove such characters, hypothetical zones of overlapping/touching printed text were computed in the signature block. Bounding box information of neighboring printed word blocks and local linearity of character strings near the signature blocks were used to detect these hypothetical zones (module T2 in Fig. 3.5). Using skeleton analysis, all the junction points that belong to hypothetical zones in a signature (module T3 in Fig. 3.5) were computed.

Next, a polygon approximation technique (module T4 in Fig. 3.5) was used to find the corner points on contours that belong to hypothetical zones. The information of junction points and corner points were used to find candidate segment lines inside those hypothetical zones. Finally, using these candidate segmentation lines, the touching

characters from the signature blocks were separated. Block wise word extraction and classification techniques are described in the following subsections.

3.4.1.1 Block-Wise Signature and Printed Text Classification

A classification-based approach using Support Vector Machine (SVM) was used to identify potential signatures components in a document. Two main stages of this approach are as follows :

3.4.1.1.1 Block/Word Extraction

Signed document images were in gray tone and the Otsu-based [90] threshold selection method was used to convert them into two-tone images (0 and 1). The digitized image may contain spurious noise pixels and irregularities on the boundary of the characters, leading to undesired effects on the system. To remove some of the above noises the image was smoothed [91] and the Hough Transform-based method was used to correct the skew of the documents.

The binarized document image was first segmented into words based on the inter-character spacing. For this purpose a morphological dilation operation [47] using a 5×5 structuring element was performed and a connected component labelling method was applied to find the bounding boxes of the word patches on the dilated document image. It was noticed that size of this structuring element performed well in the experiment. After finding the bounding box information of all the patches, any which overlap were merged. If the bounding boxes of two word blocks share a common area and if the shared area was more than 10% of any individual word's bounding box area, these two words overlapping were considered. The top-left and the bottom-right co-ordinates of the bounding box of the merged words were noted and based on the positional information of the bounding box of such word patches, the respective positions of the words were then extracted from the original document.

To discriminate the signature from a printed document, the features of all the components obtained from the component analysis were computed. A robust gradient-based feature extraction technique and the SVM as classifier were used in the experiment to classify those segmented words as a signature or printed words. The feature extraction technique and the classifier used in this method are described in Section 3.3.1 and Section 3.3.5 respectively.

3.4.1.1.2 Block/Word Level Classification

As described in Section 3.3.1, gradient features of the segmented words were computed and the feature vectors were fed into an SVM classifier [88] (Gaussian kernel with Radial Basis Function) for classification of blocks into a signature and printed word. The SVM (see Section 3.3.5 for a detailed description) was defined for a two-class problem: signature blocks and printed word blocks identification. The SVM provides a class label along with its recognition confidence as a weight for a fed feature. The value of the weight lies between 0 and 1. The combined information based on class level and recognition confidence was used for final classification decision. Two resultant images are shown in Fig. 3.6 after detection of signature blocks in document images.

3.4.1.2 Printed Text Portion Removal from Signature Block

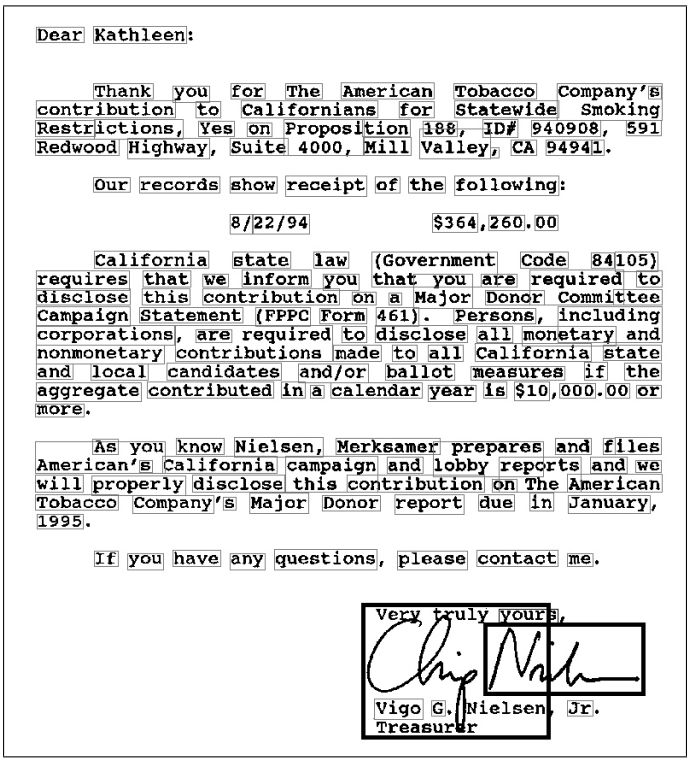
In a signature block, there may appear some printed characters along with some horizontal ruling lines. The printed characters appear mainly in two different forms: isolated form and touching/overlapping form with the signature. Sometimes, the ruling lines in document images may overlap with the signature strokes. Towards the removal of such ruling lines from the signature regions, first, the horizontal projection profile (See Fig. 3.7) of the signature block was computed to find the positions of ruling lines. Once the ruling lines were detected using the profile peak information, these lines were removed using the width information of ruling lines.

To remove the printed text portion from a signature block, first, some hypothetical printed text zones were computed in the signature block based on the neighboring printed text block information. Next, the printed text portions were removed from these hypothetical zones. Computation of the hypothetical printed text zones and the separation of text characters from these zones are discussed in the following sections.

3.4.1.2.1 Selection of Hypothetical Printed Text Zone in a Signature Block

A few isolated printed characters in the signature block may appear as mentioned earlier. Isolated characters mainly appear due to a morphological operation of the binary image, which was applied in the block extraction step. As connected component analysis was applied to the resultant image after the morphological operation, the isolated printed characters, which were very near to the signature, might have included in the signature block for their proximity.

Also, a signature may touch some printed characters during the writing of a signature. The printed characters were identified for removal from the signature blocks. To do so, the hypothetical zones of printed text were detected in a signature block. A hypothetical



Dear Kathleen:

Thank you for The American Tobacco Company's contribution to Californians for Statewide Smoking Restrictions, Yes on Proposition 188, ID# 940908, 591 Redwood Highway, Suite 4000, Mill Valley, CA 94941.

Our records show receipt of the following:

8/22/94 $364,260.00

California state law (Government Code 84105) requires that we inform you that you are required to disclose this contribution on a Major Donor Committee Campaign Statement (FPPC Form 461). Persons, including corporations, are required to disclose all monetary and nonmonetary contributions made to all California state and local candidates and/or ballot measures if the aggregate contributed in a calendar year is $10,000.00 or more.

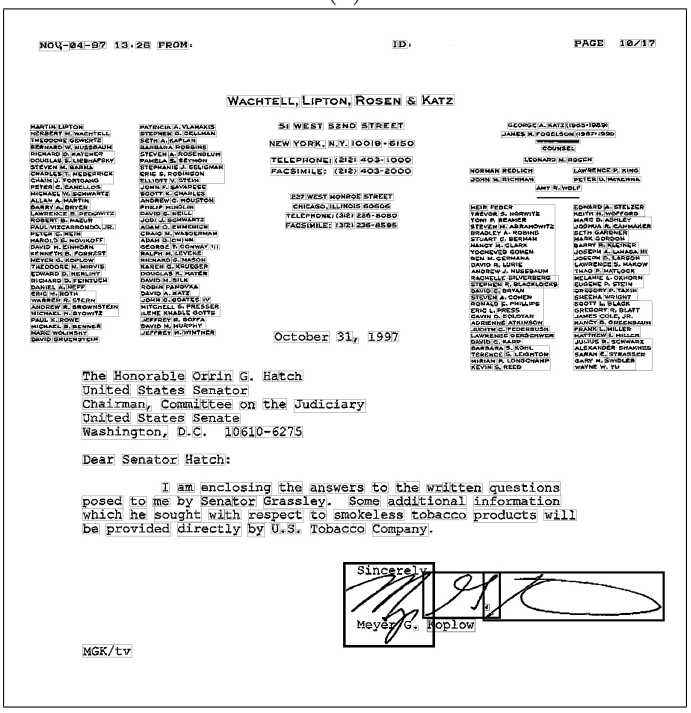
As you know Nielsen, Merksamer prepares and files American's California campaign and lobby reports and we will properly disclose this contribution on The American Tobacco Company's Major Donor report due in January, 1995.

If you have any questions, please contact me.

Very truly yours,

Vigo G. Nielsen, Jr.
Treasurer

(a)



NOV-04-97 13:28 FROM: ID: PAGE 10/17

WACHTELL, LIPTON, ROSEN & KATZ

51 WEST 52ND STREET
NEW YORK, N.Y. 10019-6150
TELEPHONE: (212) 403-1000
FACSIMILE: (212) 403-2000

227 WEST MONROE STREET
CHICAGO, ILLINOIS 60606
TELEPHONE: (312) 236-8080
FACSIMILE: (312) 236-8595

October 31, 1997

The Honorable Orrin G. Hatch
United States Senator
Chairman, Committee on the Judiciary
United States Senate
Washington, D.C. 10610-6275

Dear Senator Hatch:

I am enclosing the answers to the written questions posed to me by Senator Grassley. Some additional information which he sought with respect to smokeless tobacco products will be provided directly by U.S. Tobacco Company.

Sincerely,

Meyer G. Koplow

MGK/tv

(b)

FIGURE 3.6: Results of block-level signature detection. The detected signature blocks are marked by thick bounding boxes. Here, (a) and (b) show the signature regions are overlapped with printed text in the documents.

printed text zone refers to the area where a signature block contains the isolated or touching/overlapping printed characters. The other area in the signature block refers to

FIGURE 3.7: The figure (left) shows touching of the signature image with the ruling lines. The horizontal projection of the image is shown on the right side.

the signature zone (See Fig. 3.8).

The hypothetical zones of the printed text were estimated using the height and positional information of the neighboring printed word blocks/characters of a signature block. The neighboring word blocks of a signature block were selected first using boundary growing algorithm of the signatures block components in the left-right direction. From the boundary information of the neighboring printed word blocks, hypothetical printed text zone was calculated. If no neighboring word blocks exist on both sides of the signature block, the presence of isolated text characters was checked in the signature block itself. This is because sometimes there might not be any neighboring word blocks. The size of the printed text characters in that document was estimated using a character size histogram analysis.

Let, h_p is the mode of the character sizes in that document. Next, the isolated characters from the signature block of size less than $h_p \times 2$ were considered and the bounding box information of these characters were computed. The upper and the lower boundary lines of these printed characters were then calculated from the bounding box information. The isolated text characters in a single text line were finally verified by the consensus of bounding box information of these characters. The area between these two lines (upper and lower) was considered as hypothetical printed text zone. The uppermost and lowermost rows from the printed text zone were considered as Urow and Lrow (see Fig. 3.9) of the hypothetical printed text zone, respectively. In Fig. 3.9(a) and Fig. 3.9(b), the upper boundary and the lower boundary of these zones are marked by dotted lines. It is to be noted that there may present more than one printed text zones in a signature block and these are all detected using neighboring printed character analysis.

3.4.1.2.2 Segmentation of Printed Text Portion from Hypothetical Zone

Removal of Isolated Characters: Here first the isolated character components in the signature blocks were considered for removal. If the sizes of isolated components were less than the estimated printed character size $h_p \times 2$ and were inside the hypothetical printed text zones, these components were identified as printed characters and thus eliminated from the signature block. In Fig. 3.9(a) and Fig.3.9(b), the characters included in

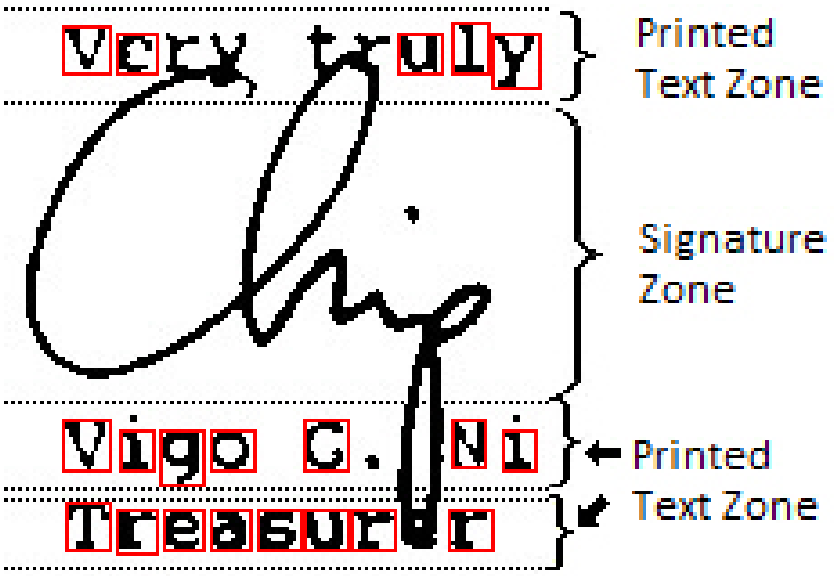
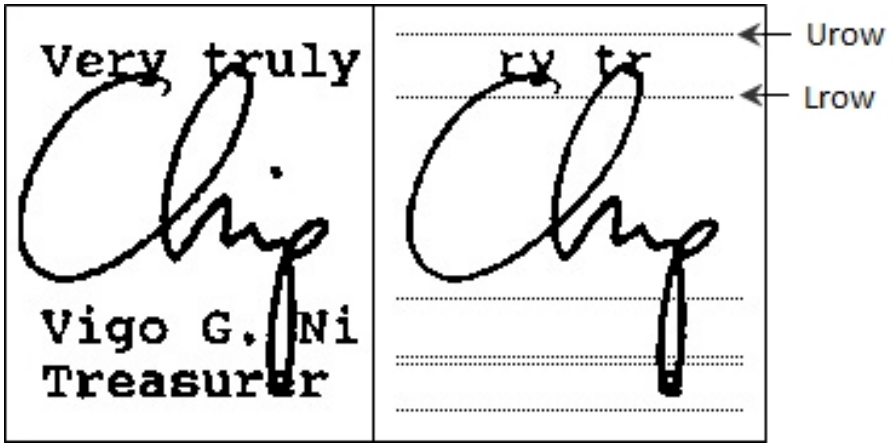


FIGURE 3.8: Computation of hypothetical printed text zones using bounding box information of characters (bounding boxes of the isolated characters are marked in red). Three printed text zones are shown here.

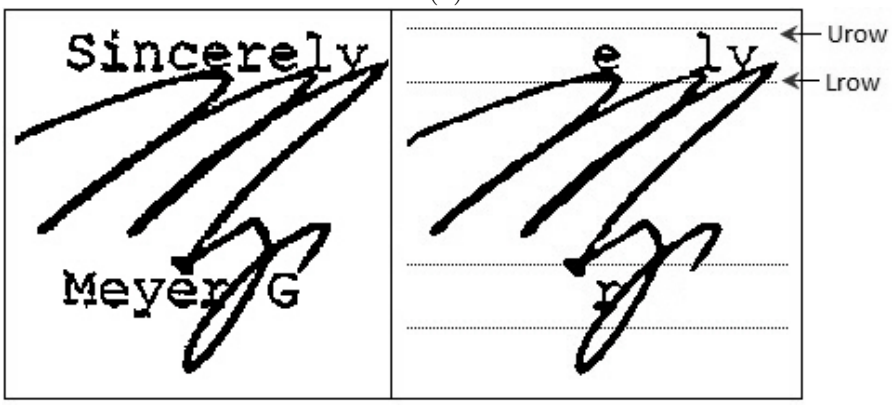
the signature block (on the left side) and signatures after removal of isolated printed characters (on the right side) are shown.

Segmentation of Touching/Overlapped Printed Text: As discussed earlier, the printed text characters which were included in the signature block and touching/overlapped with the signature cannot be separated by word block classification analysis. To remove such characters the hypothetical printed character zones information (as discussed in Section 3.4.1.2.1) of the signature block was used and it was assumed that the touching/overlapped printed characters exist inside these zones only. From the experiment it was noted that the accuracy of hypothetical printed text zone extraction method was 98.35%. For the segmentation of these touching/overlapped printed text characters, the skeleton and the contour information of the signature block was used. The skeleton/thinned image produces the junction points of strokes of the signature portion and the printed characters, whereas contour image helps to estimate the corner regions in the touching regions for effective segmentation process. Junction points, contour corner points detection and touching/overlapped text segmentation are discussed as follows.

- Computation of Junction and Contour Corner Points: The skeleton image was obtained by a rotation invariant rule-based thinning algorithm [92]. Due to the spurious effect of thinning, sometimes the strokes can be over-segmented. To avoid this, the image was smoothed before the thinning process. The junction points in this thinned image were then detected by searching the pixel locations where three



(a)



(b)

FIGURE 3.9: (a) and (b) show the signature blocks (left side) and the touching characters with the signatures in printed zones (right side). The uppermost and the lowermost rows from the printed text zone are marked by Urow and Lrow respectively.

or more neighbors exist. The junction points that lie in the hypothetical printed text zones were kept for possible segmentation.

The corner points in the contours were computed using a polygonal approximation method. The polygonisation provides corner points which were at the corner of edges. The Douglas and Peucker [93] polygonal approximation algorithm was used for this purpose. The value of the tolerance threshold in this algorithm was set with the average stroke width of the signature block. For a component, the stroke width S_w was calculated as follows. A component was scanned both row-wise and column-wise. Suppose n different runs of lengths $r_1, r_2, ...r_n$ with frequencies $f_1, f_2, ...f_n$, respectively, were obtained from the component by the scanning. Then S_w was considered as the run-length r_i having the maximum frequency. In other words, stroke width (S_w) will be r_i if $f_i = max(f_j), j = 1, 2, ...n$.

This polygonal approximation algorithm is well adapted to localize hard curvature points along a border. The corner points are necessary for segmentation because

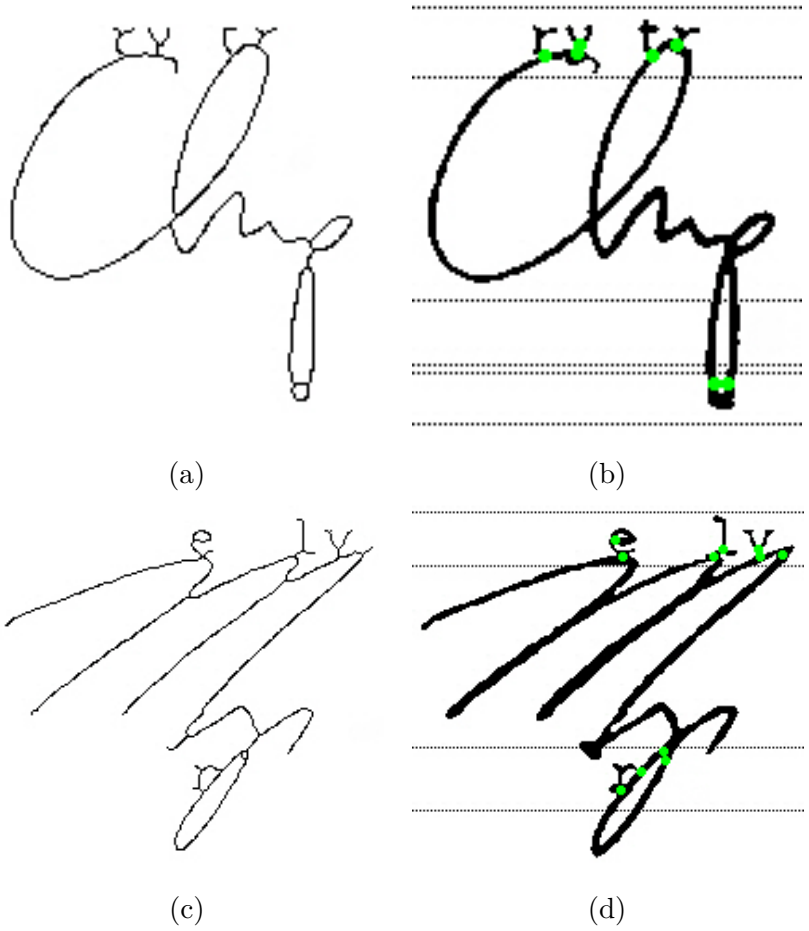


FIGURE 3.10: (a) and (c) show the skeleton images after performing thinning operation on signature of Fig.3.9 (a) and (b) respectively. (b) and (d) display the respective junction points inside the printed text zones of these signature blocks.

when the printed characters are touching with the signature strokes, they usually form corners at the touching region. Junction points and corner points obtained from hypothetical printed text zone are shown in Fig. 3.10 and Fig.3.11, respectively.

- Touching/Overlapped Text Segmentation: When a character touches a signature, smoothness of the signature contour will be affected at the touching portion. Here the idea is to find some possible touching points and verify them according to the smoothness of the signature contour. For this purpose, the process is as follows.

  After computation of the junction points (on the skeleton) and the corner points (on the contour) of the hypothetical printed text zone, the printed characters which were touching/overlapped in these regions were separated. For this purpose, first, the signature strokes that pass from the signature zones (outside of hypothetical printed text zone) to the hypothetical printed text zones were detected by scanning the hypothetical printed text zone from left to right. Next, some points where

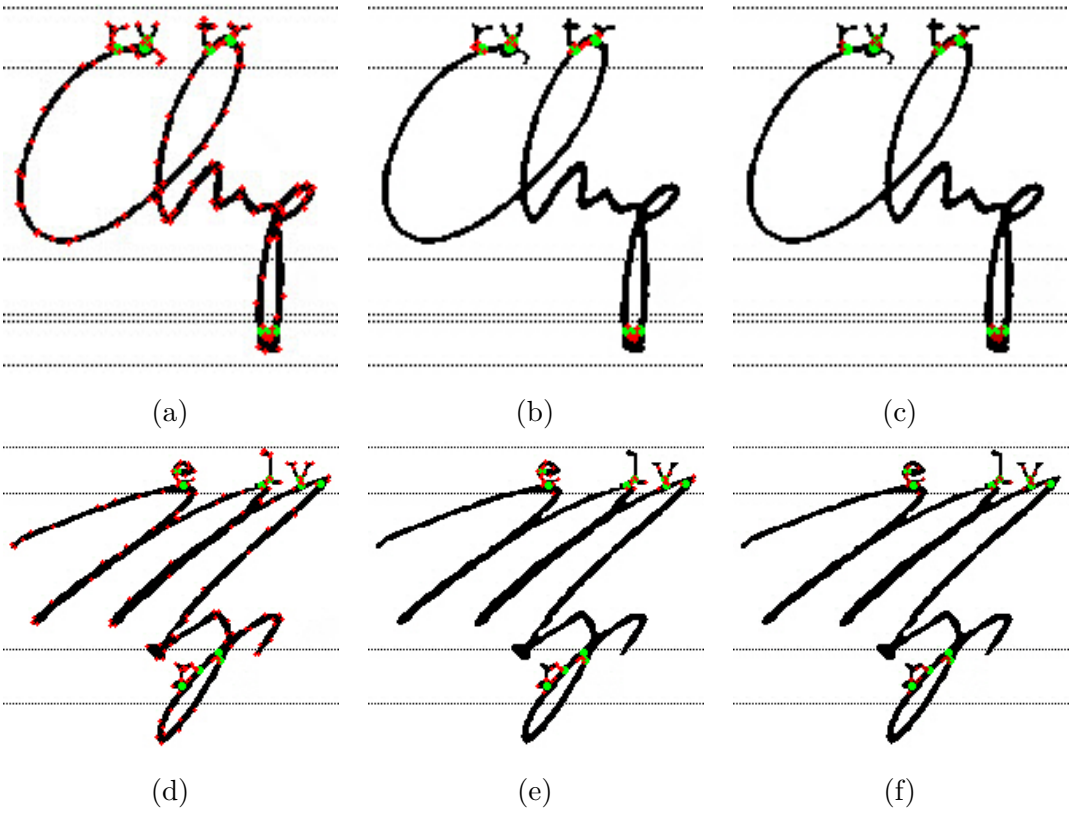


FIGURE 3.11: Figure shows the contour and junction points in printed text zones of the signature. (a) and (d) show the signature images with all the contours points marked by red colour. Junction points belong to the printed zones are marked by green circles. (b) and (e) show the contour points which are inside the printed zones and close to the junctions points. (c) and (f) show the contours candidate points for possible segmentation cut.

Urow or Lrow (Urow and Lrow are described in Section 3.4.1.2.1) touches the signature strokes were found. From these touching points of the signature strokes, the contours (both sides) of strokes were traced towards the hypothetical printed text zones.

During the tracing of signature strokes from the touching points, if a corner point (obtained by polygon approximation) was visited, that corner point was considered as a candidate corner point. The nearest junction point of this candidate corner point was then searched. If a junction point was obtained, the corner points surrounding (having a distance less than $2 \times S_w$) this junction point was also searched. The orientations of these corner points with the candidate corner point were computed. The distance between a junction point and corner point was considered as less than $2 \times S_w$ in the experiment. Here S_w is the stroke width of the segmentation block which discussed earlier. It should be noted that because of different writing medium of signatures, stroke width may differ. Since the method is based on stroke width, it can adapt and work well for different signatures.

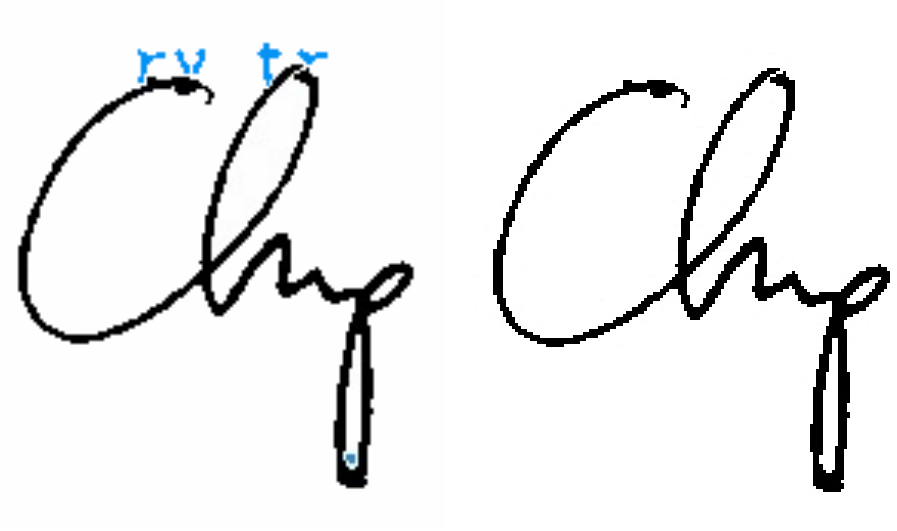
For each junction point, a segmentation cut was chosen using a pair of corner points and the orientation of skeleton lines. A segmentation cut is defined by the line joining a pair of candidate corner points. As more than one segmentation cut might exist, the best segmentation cut was chosen according to the orientation of strokes of the thinned image at the junction point. The orientation was computed locally at each junction point based on the neighboring pixels. Finally, the segmentation cut was verified by checking the bounding box of the segmented stroke. If the bounding box of the segmented stoke was within the hypothetical printed text zone, they were identified as text characters and removed. This process of segmentation was performed recursively until the tracing reaches stroke ends or crosses the hypothetical printed text zone. Algorithm 1 shows the major steps of the segmentation module of the proposed approach. Fig. 3.12(b) shows a sample segmented signature after removal of printed characters and the blue colour pixels of Fig. 3.12(a) shows the pixels that were removed from signature.

Algorithm 1 Signature strokes segmentation

Require: Printed text zones (Z_p) containing Corner points (C_p) and Junction Points (J_p)
Ensure: Signature strokes are separated from printed text characters
  Step 1: Find all text strokes (S_p^i) in Z_p by scanning left to right
  for Each S_p^i that touches Urow/Lrow of Z_p **do**
    Step 2: Select a corner point (C_p^i) by tracing the contour
    Step 3: Find the junction point (J_p^i) nearest to C_p^i
    Step 4: Find other corner points (C_p^k) near J_p^i having Euclidean distance $\leq 2 \times S_w$
    for Each corner point (C_p^k) **do**
      Step 5: Compute orientation θ^{ik} between C_p^i and C_p^k
      Step 6: Select best C_p^k according to similarity of θ^{ik} to local orientation at J_p^i
      if Residual part after segmentation cut obtained by (C_p^i, C_p^k) is inside Z_p **then**
        Step 7: Select (C_p^i, C_p^k) as final segmentation cut
      end if
    end for
    Step 8: goto Step 2
  end for

3.4.2 Signature Detection using Bag-of-Visual-Words (BoVW)

An efficient patch-based SIFT descriptor with a Spatial Pyramid Matching (SPM)-based pooling scheme was applied for feature extraction in the proposed signature detection task. Detail descriptions of computation of grid-wise sift descriptors and SPM are presented in Section 3.3.4. Here, detection of signatures was achieved by a classification-based approach. Components in a document were classified into two classes (i.e. signature components and printed components). The feature extraction module used here

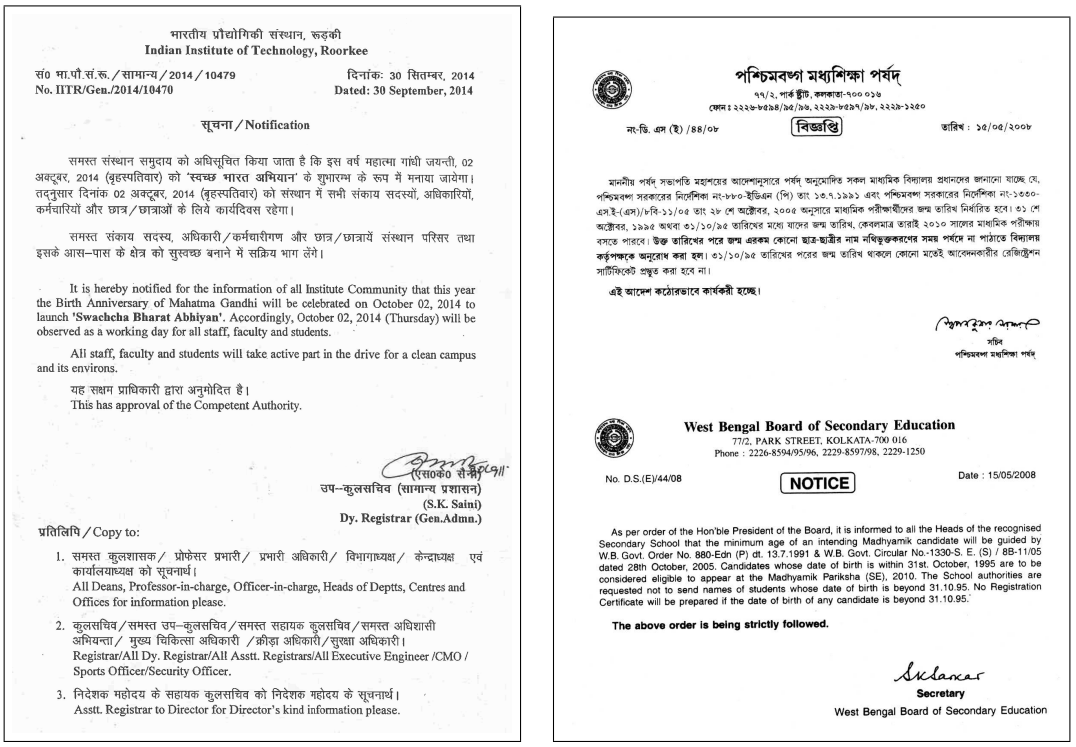


(a) (b)

FIGURE 3.12: Examples of touching character separation from signature. (a) Pixels marked by blue colour show segmented printed character (b) Segmented signature after removal of printed characters

has three components. A flow diagram of feature extraction and classification for signature detection is presented in Fig. 3.3. First, SIFT descriptors were extracted from the components of the signature and the K-means clustering algorithm was used to create the codebook. Next, the SPM-based scheme was applied for the final representation of an image. Finally, the SVM was employed for classification. The general idea of the SIFT-descriptors and the SPM were applied in the proposed approach are described in Section 3.3.4.1 and Section 3.3.4.2, respectively.

BoVW-Based Feature Extraction: This section briefly describes the feature extraction and classification method at the component level for signature detection. First, the component image was divided into 14×14 patches to obtain a dense regular grid instead of interest points based on the comparative evaluation of Fei-Fei and Perona [94]. The higher dimensional SIFT descriptors [86] of the 16×16 pixel patch were computed over each patch. Next, the K-means clustering technique was applied on the extracted SIFT descriptors from the training set for the generation of the codebook. The typical vocabulary size for the experiments was 256. The number of patches (14×14) and the size of the vocabulary (256) was selected empirically as no significant increase in performance beyond these numbers was noticed. Finally, an SPM scheme was employed to generate the feature vector, which was then fed to the SVM classifier [88]. In this experiment, the image was divided into $2^l \times 2^l$ segments in three different scales $l = 0, 1$ and 2. 21 (16+4+1) BoF histograms were computed (SPM configuration was adopted from Lazebnik et al. [87]) from these three levels, and all the histograms were concatenated to get



भारतीय प्रौद्योगिकी संस्थान, रूड़की
Indian Institute of Technology, Roorkee

सं0 भा.पौ.सं.रू. / सामान्य / 2014 / 10479 दिनांक: 30 सितम्बर, 2014
No. IITR/Gen./2014/10470 **Dated: 30 September, 2014**

सूचना / Notification

समस्त संस्थान समुदाय को अधिसूचित किया जाता है कि इस वर्ष महात्मा गांधी जयन्ती, 02 अक्टूबर, 2014 (वृहस्पतिवार) को **'स्वच्छ भारत अभियान'** के शुगारम्भ के रूप में मनाया जायेगा। तदनुसार दिनांक 02 अक्टूबर, 2014 (वृहस्पतिवार) को संस्थान में सभी संकाय सदस्यों, अधिकारियों, कर्मचारियों और छात्र / छात्राओं के लिये कार्यदिवस रहेगा।

समस्त संकाय सदस्य, अधिकारी / कर्मचारीगण और छात्र / छात्रायें संस्थान परिसर तथा इसके आस–पास के क्षेत्र को सुस्वच्छ बनाने में सक्रिय भाग लेंगे।

It is hereby notified for the information of all Institute Community that this year the Birth Anniversary of Mahatma Gandhi will be celebrated on October 02, 2014 to launch **'Swachcha Bharat Abhiyan'**. Accordingly, October 02, 2014 (Thursday) will be observed as a working day for all staff, faculty and students.

All staff, faculty and students will take active part in the drive for a clean campus and its environs.

यह सक्षम प्राधिकारी द्वारा अनुमोदित है।
This has approval of the Competent Authority.

(एस0के0 सैनी)
उप–कुलसचिव (सामान्य प्रशासन)
(S.K. Saini)
Dy. Registrar (Gen.Admn.)

प्रतिलिपि / Copy to:

1. समस्त कुलशासक / प्रोफेसर प्रभारी / प्रभारी अधिकारी / विभागाध्यक्ष / केन्द्राध्यक्ष एवं कार्यालयाध्यक्ष को सूचनार्थ।
   All Deans, Professor-in-charge, Officer-in-charge, Heads of Deptts, Centres and Offices for information please.
2. कुलसचिव / समस्त उप–कुलसचिव / समस्त सहायक कुलसचिव / समस्त अधिशासी अभियन्ता / मुख्य चिकित्सा अधिकारी / क्रीड़ा अधिकारी / सुरक्षा अधिकारी।
   Registrar/All Dy. Registrar/All Asstt. Registrars/All Executive Engineer /CMO / Sports Officer/Security Officer.
3. निदेशक महोदय के सहायक कुलसचिव को निदेशक महोदय के सूचनार्थ।
   Asstt. Registrar to Director for Director's kind information please.

(a)

পশ্চিমবঙ্গ মধ্যশিক্ষা পর্ষদ্
৭৭/২, পার্ক স্ট্রীট, কলকাতা-৭০০ ০১৬
ফোন ঃ ২২২৬-৮৫৯৪/৯৫/৯৬, ২২২৯-৮৫৯৭/৯৮, ২২২৯-১২৫০

নং-ডি. এস (ই) /৪৪/০৮ **বিজ্ঞপ্তি** তারিখ : ১৫/০৫/২০০৮

মাননীয় পর্ষদ্ সভাপতি মহাশয়ের আদেশানুসারে পর্ষদ অনুমোদিত সকল মাধ্যমিক বিদ্যালয় প্রধানদের জানানো যাচ্ছে যে, পশ্চিমবঙ্গ সরকারের নির্দেশিকা নং-৮৮০-ইডিএন (পি) তাং ৩.৭.১৯৯১ এবং পশ্চিমবঙ্গ সরকারের নির্দেশিকা নং-১৩৩০-এস.ই-(এস)/৮বি-১১/০৫ তাং ২৮ শে অক্টোবর, ২০০৫ অনুসারে মাধ্যমিক পরীক্ষার্থীদের জন্ম তারিখ নির্ধারিত হবে। ৩১ শে অক্টোবর, ১৯৯৫ অথবা ৩১/১০/৯৫ তারিখের মধ্যে যাদের জন্ম তারিখ, কেবলমাত্র তারাই ২০১০ সালের মাধ্যমিক পরীক্ষায় বসতে পারবে। **উক্ত তারিখের পরে জন্ম এরকম কোনো ছাত্র-ছাত্রীর নাম নথিভুক্তকরণের সময় পর্ষদে না পাঠাতে বিদ্যালয় কর্তৃপক্ষকে অনুরোধ করা হল।** ৩১/১০/৯৫ তারিখের পরের জন্ম তারিখ থাকলে কোনো মতেই আবেদনকারীর রেজিস্ট্রেশন সার্টিফিকেট প্রস্তুত করা হবে না।

এই আদেশ কঠোরভাবে কার্যকরী হচ্ছে।

সচিব
পশ্চিমবঙ্গ মধ্যশিক্ষা পর্ষদ

West Bengal Board of Secondary Education
77/2, PARK STREET, KOLKATA-700 016
Phone : 2226-8594/95/96, 2229-8597/98, 2229-1250

No. D.S.(E)/44/08 **NOTICE** Date : 15/05/2008

As per order of the Hon'ble President of the Board, it is informed to all the Heads of the recognised Secondary School that the minimum age of an intending Madhyamik candidate will be guided by W.B. Govt. Order No. 880-Edn (P) dt. 13.7.1991 & W.B. Govt. Circular No.-1330-S. E. (S) / 8B-11/05 dated 28th October, 2005. Candidates whose date of birth is within 31st. October, 1995 are to be considered eligible to appear at the Madhyamik Pariksha (SE), 2010. The School authorities are requested not to send names of students whose date of birth is beyond 31.10.95. No Registration Certificate will be prepared if the date of birth of any candidate is beyond 31.10.95.

The above order is being strictly followed.

Secretary
West Bengal Board of Secondary Education

(b)

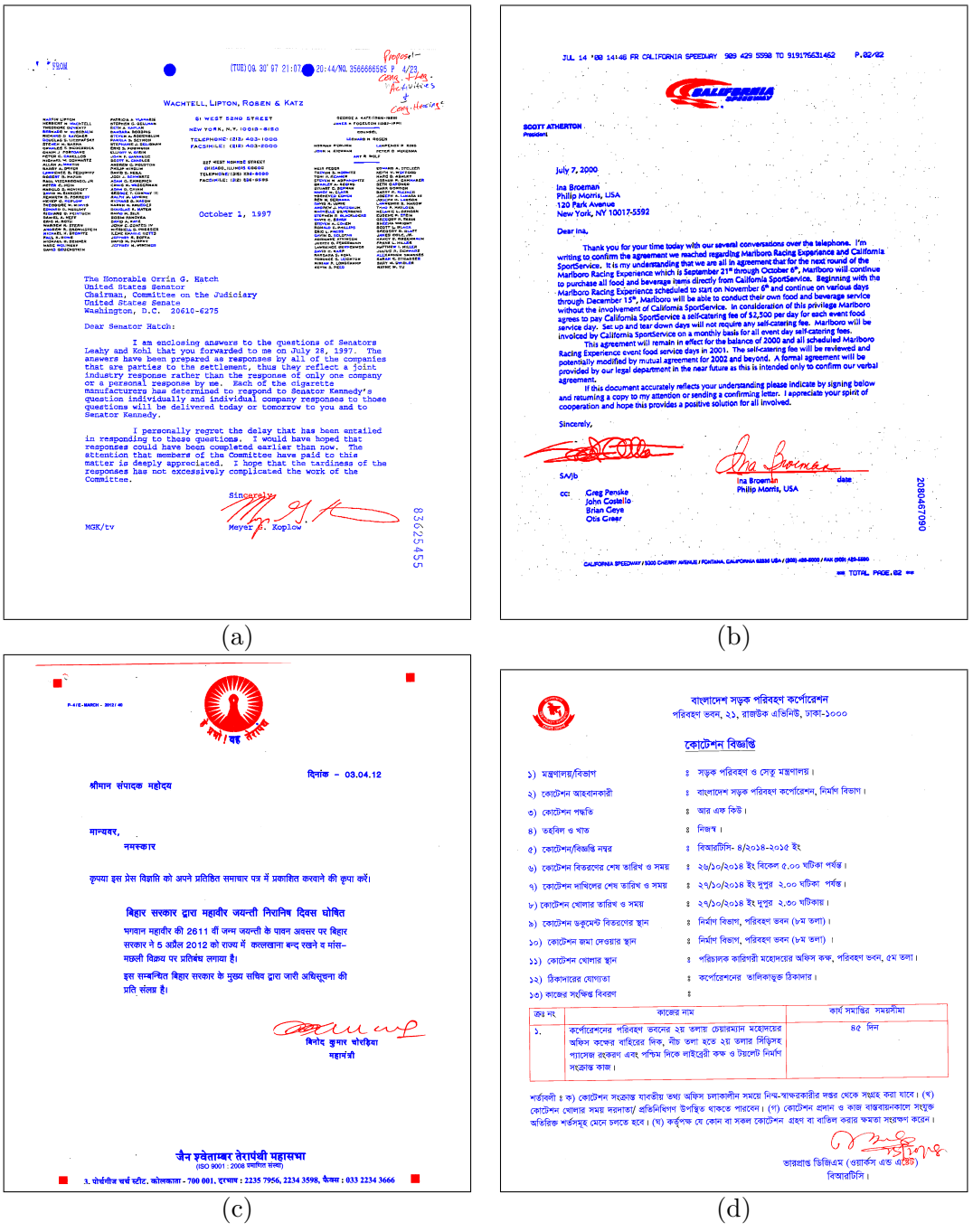
FIGURE 3.13: Samples of Indian multi-script official documents. (a) Document containing English (Roman) and Devnagari scripts (b) Document containing English and Bangla scripts.

the final vector representation of an image. The equation below represents the pyramid match kernel for three scales:

$$K_{\Delta} = I_2 + \frac{1}{2}(I_1 - I_2) + \frac{1}{4}(I_0 - I_1) \tag{3.10}$$

Classifier: Support Vector Machine (SVM) [88] was used as a classifier in the experiments; details are presented in Section 3.3.5. The reported results are based on the Gaussian kernel SVM as the Gaussian kernel outperformed other non-linear SVM kernels. The hyperparameters of the SVM were set as follows; kernel type = RBF, $\gamma = 1$ and $C = 1$. The best results were achieved by setting the above values of these parameters, which were determined by a validation process.

The qualitative signature detection results on single script documents are shown in Fig. 3.14. Fig. 3.15 shows the signature detection results on two sample multi-script documents.

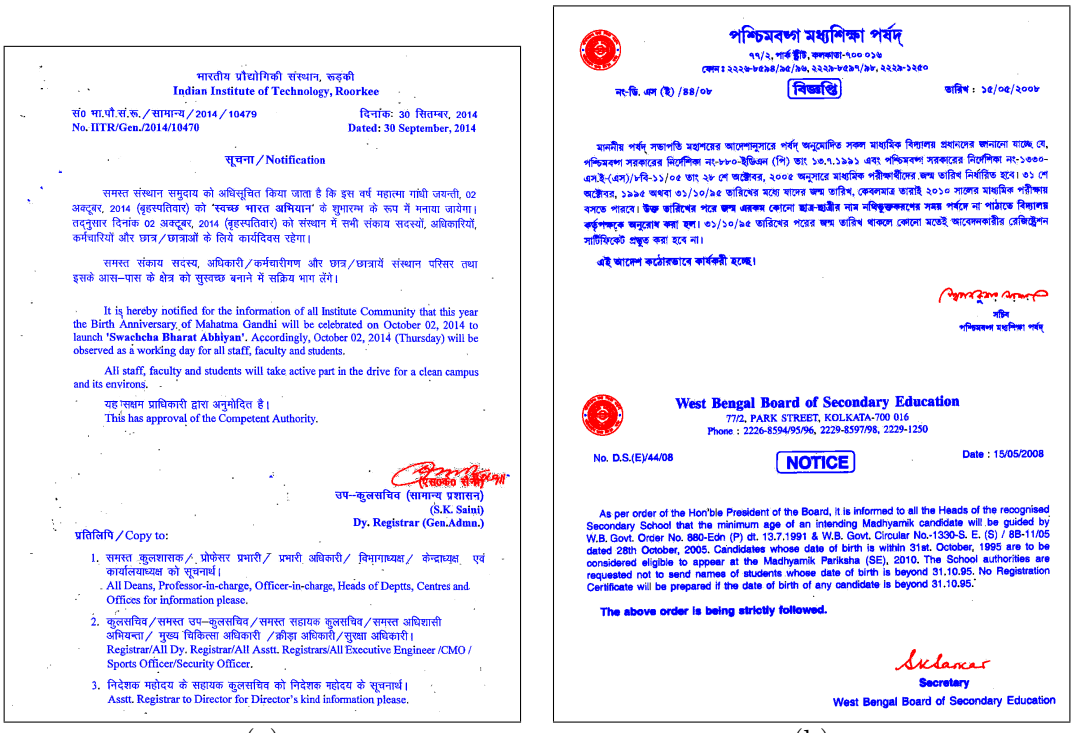


(a) (b) (c) (d)

FIGURE 3.14: Classification result of printed and signature/handwritten components on the documents shown in Fig. 3.1. (a,b) English ('Tobacco'), (c) Hindi and (d) Bangla. Printed text and signature/handwritten components are marked in blue and red, respectively.

3.5 Signature Segmentation

A signature can consist of one or more components and a document can contain more than one signature. Hence, multiple components of the signature can be present and can be detected as well in a document. Moreover, some misclassified non-signature



(a) (b)

FIGURE 3.15: Signature detection results from Indian bi-script official documents. (a) Results on bi-script document containing English and Devnagari scripts (b) Results on bi-script document containing English and Bangla scripts. Printed text and signature components are marked in blue and red, respectively.

components can also be present in the document. Thus, proper segmentation algorithm of signatures is an important task.

Therefore, a grouping of all the components belonging to a signature is required for signature matching with the query signature. To group signature components, first, corner points from the document image were computed and then a density-based clustering algorithm (DBSCAN [82]) was applied for discovering clusters of signature components. The algorithm computes the number of clusters starting from the estimated density distribution.

3.5.1 Corner Points Computation

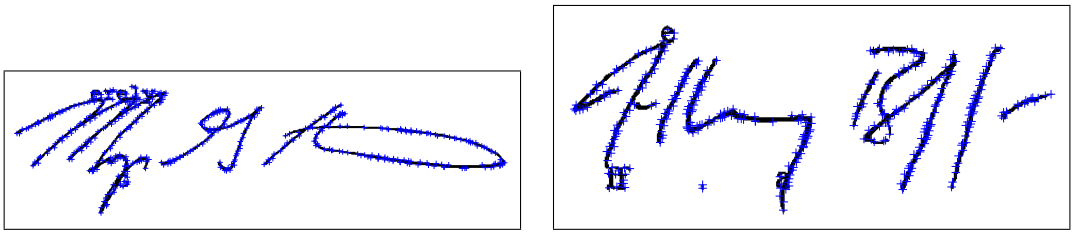
Corner points from the components of the document were first computed using Harris-Stephens combined corner/thin edge detector algorithm [95] which is invariant to rotation, shift or even affine change of intensity. The variance of light was computed using the local auto-correlation energy function:

$$E(x,y) = \sum_{u,v} W_{u,v}(I(x+u, y+v) - I(x,y))^2$$

where (u, v) denote a neighborhood of (x, y). A smooth Gaussian circular window with

$$W_{u,v} = exp(-\frac{u^2+v^2}{2\sigma^2})$$

is the window function, and normally its value is 1, whereas $I(x+u, y+v)$ is the shifted intensity. Fig. 3.16 shows two sample signatures where corners points are plotted using blue markers. Next, the co-ordinates of corner points were fed for processing by density-based spatial clustering.



(a) (b)

FIGURE 3.16: (a,b) Signature images after computation of corner points. Blue markers represent corner points.

3.5.2 Density-Based Clustering

DBSCAN is a clustering algorithm proposed by Ester et al. [82], which finds the number of clusters starting from the estimated density distribution of corresponding nodes. It shows its efficiency on a large spatial database of synthetic data as well as real data by discovering the clusters of arbitrary shape. In comparison to other clustering algorithms, it requires minimal domain knowledge. The algorithm prerequisite only one parameter (i.e. distance threshold which was used to determine the maximum distance among points in a cluster) and the algorithm also supports the user in determining the appropriate value for it.

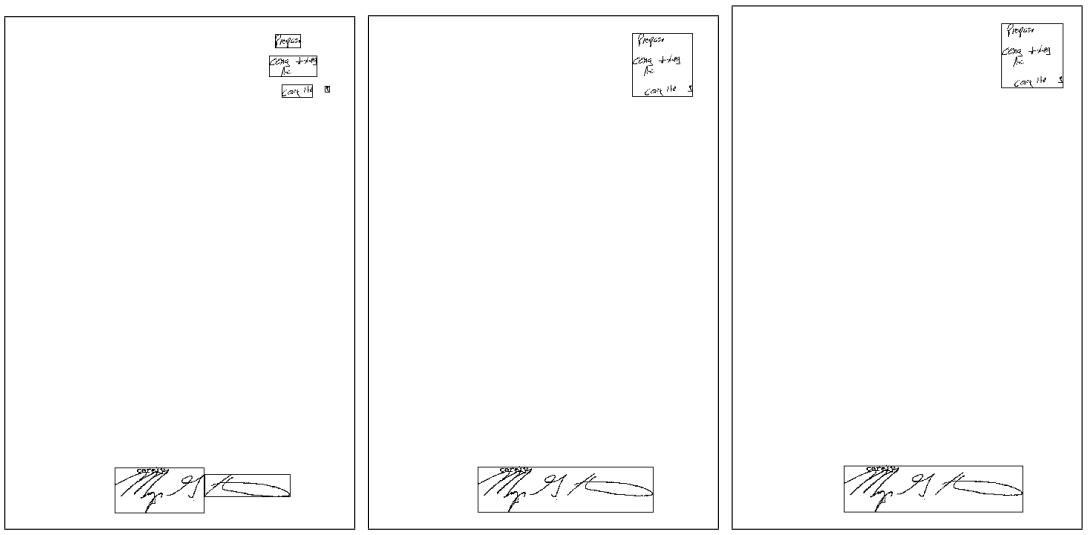
In the proposed component grouping work, an iterative method was used to set the threshold value, and for the iteration, some clusters from the corner points obtained from the segmented documents were computed. First, a maximum threshold for a density-based clustering algorithm was computed based on the size of the query signature. Ten percent of the maximum threshold was used as an initial threshold for the clustering in the first iteration. Next, the bounding box of all the clusters and then the features from each of the clusters' bounding boxes were computed.

The details of the cluster level feature extraction and matching techniques are described in Section 3.6. In the next step, features with the query signature's feature were matched and the minimum matching distance obtained from this iteration was stored. The distance threshold was increased by ten percent for the next iteration. If the minimum matching distance from any iteration was larger than the previous one the process was stopped and the minimum distance from the previous iteration was considered as

the final minimum distance. The step-by-step algorithm is presented in Algorithm 2. Although, the component grouping algorithm has scope for ten iterations, it was noticed from the experiments that signature components were properly grouped within the first three iterations. Fig. 3.17 shows some sample results from the signature component grouping experiment. In Fig. 3.17(a1) components were grouped into 6 clusters and the components of the actual signature were grouped into two clusters after the first iteration. Fig. 3.17(a2) and Fig. 3.17(a3) show the result after the second and third iterations respectively where the actual signature components were grouped properly into one cluster. In the second iteration, actual signature components were grouped properly.

Algorithm 2 Grouping of signature components and matching with the query signature

Require: A query signature with the document to be matched
Ensure: Return a matching score with the query signature
Computation of Maximum Threshold (MaxTh) for DBSCAN clustering. Height and Width refer to the query signature's height and width\
Step 1: $MaxTh \leftarrow max(Height, Width)$
Step 2: $InitTh \leftarrow MaxTh \times 0.1$
Step 3: $Dist_{Match} \leftarrow -1$
Step 4: $MinDist_{PreviousStep} \leftarrow -1$
for $k \leftarrow InitTh$ to $MaxTh$ step $InitTh$ **do**
  Step 5: $C \leftarrow DBSCAN(CornerPoints, k, MinPoints)$
  C refers to clustered corner points, CornerPoints refer to Harris-Stephens corner points computed from the documents and MinPoints refers to the minimum points threshold. The cluster bounding box refers to a rectangle computed using the boundary points of the cluster\
  for Each Cluster in C **do**
    Step 6: Extract feature from cluster bounding box image
    Step 7: $Dist \leftarrow FuncMatchDist(Query_{Signature}, Target_{Signature})$
    if $Dist_{Match} < 0$ **then**
      Step 8: $Dist_{Match} \leftarrow Dist$
    else
      if $Dist_{Match} \geq Dist$ **then**
        Step 9: $Dist_{Match} \leftarrow Dist$
      end if
    end if
  end for
  if $MinDist_{PreviousStep} < 0$ **then**
    Step 10: $MinDist_{PreviousStep} \leftarrow Dist_{Match}$
  else
    if $MinDist_{PreviousStep} > Dist_{Match}$ **then**
      Step 11: $MinDist_{PreviousStep} \leftarrow Dist_{Match}$
    else
      Step 12: Return $Dist_{Match}$
    end if
  end if
end for



(a1) (a2) (a3)

FIGURE 3.17: Example of clustering results. Component clusters are shown after (a1) first (a2) second (a3) third iterations on the document shown in Fig. 3.1(a).

3.6 Matching with the Query Signature

In this section, the signature shape encoding technique and matching procedure for the retrieval of documents are described. The encoding of signature images is almost the same as the proposed feature extraction technique described for signature components detection in Section 3.4. However, the signature background information along with the foreground information for encoding the signature are incorporated here.

3.6.1 Foreground-Based Feature

The shape coding technique of signatures also involves three steps as discussed in Section 3.2. First, to code the shape of the signature, the signature image was divided into densely sampled local patches and a descriptor has been computed from each of the patches. Here, signature images were divided into 900 (30×30) patches and one SIFT descriptor was computed from each patch. The number of patches determined in this stage was based on experimentation. Next, 900 SIFT-descriptors were used in the next process of computation of features based on codebook learning and a 3 level Spatial Pyramid Matching-based technique. Fig. 3.19(a1), Fig. 3.19(b1) and Fig. 3.19(c1) show 900 descriptor patches from three samples of foreground signatures namely English, Hindi and Bangla respectively.

3.6.2 Background-Based Feature

The cavity regions and loops in a signature are referred to as background information in this approach. The cavity regions were obtained using the Water Reservoir concept [96]. The water reservoir in all four directions (top, bottom, left, right) and loops present in an image were used. Fig. 3.18 shows reservoirs from all four directions extracted from a signature. Here, the background signature image was also divided into 900 (30×30) patches and one SIFT descriptor was computed from each patch. Next, 900 SIFT-descriptors were used in the next step for computation of features using code-book learning and a Spatial Pyramid Matching-based technique. Fig. 3.19(a2), Fig. 3.19(b2) and Fig. 3.19(c2) show three sample signatures from English, Devnagari, and Bangla, respectively, where the images were divided into 30×30 grid patches and the patch centres are marked. Finally, the foreground and background features were concatenated to get the final features.

FIGURE 3.18: Loops and water reservoirs in three signature images are shown and red is used to mark the reservoirs. The original signature, loops and the water reservoir from top, left, right and bottom sides are shown respectively in (a1- a6) for English, (b1- b6) Hindi and (c1- c6) Bangla signatures.

3.6.3 Distance between Signature Images

Three matching distances such as Euclidean distance, rank correlation and DTW-based methods for computation between the query signature and signatures from the document images were considered. Given the two feature vectors $X_m | m = 1, 2, ..., n$ and

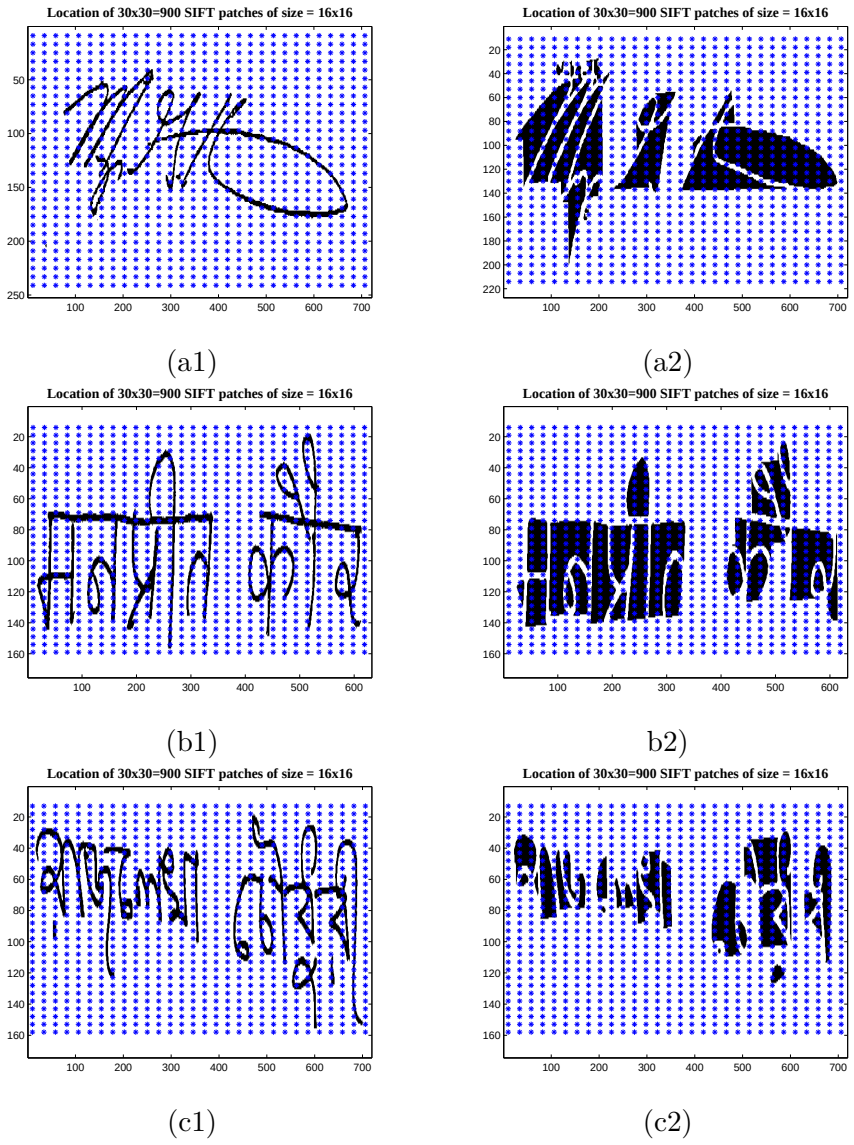


FIGURE 3.19: (a1,b1,c1) Samples of English, Hindi and Bangla foreground signatures after grid-based 900 (30×30) SIFT patches are marked. (a2,b2,c2) Samples of background signatures after grid-based 900 (30×30) SIFT patches are marked on background information.

$Y_m | m = 1, 2, ..., n$, similarity distance between X and Y using the Euclidean distance was calculated using Equation 3.11. Equation 3.12 shows the formula for the linear correlation coefficient, which measures the strength and direction of a linear relationship between the vectors of a query signature and signatures from the documents.

$$Distance_{Euclidean}(X, Y) = \sqrt{\sum_{i=1}^{n} (X_i - Y_i)^2} \tag{3.11}$$

$$Corr(X,Y) = \frac{n\sum XY - \Big(\sum X\Big)\Big(\sum Y\Big)}{\sqrt{n\Big(\sum X^2\Big) - \Big(\sum X\Big)^2}\sqrt{n\Big(\sum Y^2\Big) - \Big(\sum Y\Big)^2}} \tag{3.12}$$

Here DTW was used on two sequences of feature vectors. The DTW distance between two vectors X and Y were calculated using a matrix D. Where

$$D(i,j) = min\begin{pmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{pmatrix} + d(x_i, y_i) \tag{3.13}$$

$$d(x_i, y_i) = \sum (X_i - Y_j)^2 \tag{3.14}$$

Finally, this matching cost was normalized by the length of the warping path. Here, it was observed that slant and skew angle of a signature class were usually constant but the larger variation normally lies in character spacing. DTW performed better in the experiment because of the flexibility to compensate such variations.

3.7 Results and Discussion

This section evaluates the performance of different levels of the proposed signature-based document retrieval approach by considering various measures. The different datasets used in the different level of experiments are described in Section 3.7.1. Qualitative and quantitative results are detailed which shows the efficiency of the proposed approach.

3.7.1 Datasets

To detect the signature blocks in printed text documents using 400-dimensional gradient-based feature (Section 3.3.1 presents the feature details), The SVM-based classifier was trained for two classes (i.e. Signature and Printed word). The GPDS signature dataset [81] and English printed words block dataset were used to train the classifier. Here, the SVM-based classifier was trained with 3080 signatures blocks considered from GPDS dataset and 7684 English word blocks extracted from different types of printed document collected mainly from books, daily newspapers, official documents, magazines, journals etc.

For testing the proposed system of signature segmentation, the dataset of 'Tobacco-800' industrial archives [1] was used. The documents were written in English and the signatures on these documents also contain handwritten English characters. In the

'Tobacco-800' dataset, no ground truth of signatures is available based on pixel count information. Thus, the pixel-level ground truth was prepared for the evaluation of the system. A total number of pixels on a signature component with and without touching/overlapped printed characters was counted for the quantitative performance of the method. By using a template matching technique, a measure of difference on pixel count was taken for a quantitative measure.

There exists no standard dataset consisting of signatures and printed components of English, Devnagari and Bangla scripts to train the SVM classifier at the signature detection stage for multiscript documents. Hence, a dataset was created using components of English, Devnagari, and Bangla scripts. Printed components were extracted from different types of documents such as newspaper, books, magazines etc. English signatures used in the experiment were extracted from the 'Tobacco' dataset. The Hindi and Bangla signatures used for training the SVM classifier were taken from the dataset created by Pal et al. [97]. The signatures were collected from 300 and 200 writers of Hindi and Bangla, respectively.

Table. 3.1 and Table. 3.2 shows the details of the training and test data used in the multi-script document retrieval experiments respectively. It should be noted that the training and test datasets were different in the experiments. In total, 7390 and 5854 components of printed and signature/handwriting respectively were used from the English script to train the SVM classifier for signature detection experiment on English documents. Likewise, 7670 and 5618 components of printed and signature/handwriting from Devnagari script and 5575 and 6950 components of printed and signature/handwriting from Bangla script were used. These components were also used to train the classifier for bi-script document classification (i.e. documents shown in Fig. 3.13). The document retrieval system was tested on three sets of document data for the three scripts considered in this experiment. The 'Tobacco' dataset was used for testing the system on English script. A database of 560 official notices and letters written in Devnagari, Bangla, and bi-lingual scripts was also created. In total, 300 documents of Devnagari and 260 documents of Bangla script are present in the collected dataset. The dataset of logos from the Laboratory for Language and Media Processing, University of Maryland [98] along with 400 downloaded logos has been used for document retrieval experiments based on logo information. A few samples of logos are presented in Fig. 3.20.

FIGURE 3.20: Some samples from logo dataset.

TABLE 3.1: The dataset used for training the SVM classifier for signature detection

Types of Data	English	Hindi	Bangla
Printed components	7390	7670	5575
Signature/Handwritten components	5854	5618	6950
Logos	106+400	-	-

TABLE 3.2: The dataset used for testing the signature detection system

Types of Data	English	Hindi	Bangla
Full page documents	'Tobacco' [1]	300	260

3.7.2 Empirical Analysis for Selection of Gradient Feature

Different off-the-shelf features like Gabor (see Section 3.3.2) and Zernike Moment[85] (see Section 3.3.3) were compared against the gradient-based feature (see Section 3.3.1) for the first work on signature detection. In this experiment, the objective was to find the best features among these three features for block level classification. A 5-fold cross-validation method of the training dataset was used (3080 GPDS signatures and 7684 English word blocks) for this purpose and it was noted that the gradient-based features outperformed using the SVM classifier. Table 3.3 shows the comparison of the performance of the cross validation technique using these three (Gabor, Zernike Moment, gradient) features with 99.9% accuracy obtained on the classification of signature and printed word blocks using the 400-dimensional gradient features. Using Gabor and Zernike-based features 61.60% and 91.78% accuracy were obtained respectively on signature and printed text block separation. Since gradient-based features provide the best accuracy, the gradient feature was used for the classification in the first work of signature detection from the documents, although later on SIFT descriptors-based features (see Section 3.3.4) outperformed the gradient-based feature in the next experiment of signature detection.

TABLE 3.3: Signature and printed text block separation results using SVM on three different features. A 5-fold cross validation scheme was used here for result computation

Feature	Feature Dimension	Accuracy(%)
Gabor	400	61.60
Zernike Moment	72	91.78
Gradient	400	99.90

3.7.3 Performance Evaluation on Signature Detection

As discussed earlier, two different approaches were considered in the signature segmentation task. In the first experiment, the gradient-based features were considered and touching characters were separated from the signature strokes using contextual information. In the second method, Bag-of-Visual-Words (BoVW) with SIFT Descriptors-based features was used. The performances from both the systems are given below.

3.7.3.1 Results based on gradient features

Signature block detection: The patches obtained after morphological dilation of the 'Tobacco-800' dataset were used for signature block detection and an overall accuracy of 95.58% was achieved. The errors were mainly due to segmentation problems at the block level. Some broken parts of signatures were identified as non-signature and some patches which contain printed words of two consecutive rows were misclassified as signature blocks. It was noticed that patches of two consecutive rows were sometimes formed due to touching. By way of comparison, the performance of the proposed method and the performance of an earlier similar work on the same dataset are given in Table 3.7.

Evaluation on touching/overlapping segmentation: A common ratio of precision (P) and recall (R) measure was used to evaluate the separation of touching/overlapping printed characters from the signature block. For the experiment of touching characters segmentation in the signature region, 1120 touching/overlapping strokes were considered and an overall precision of 90.30% and recall of 86.2% was achieved. In Fig.3.21, some segmented signatures obtained by the proposed work are shown.

Evaluation on pixel level segmentation: A template matching technique was used to measure the quantitative accuracy on the pixel level and the segmented signature with the ground truth of signature image was compared. The results on some signature images are shown in Table 3.4. In this table the acronyms PGT (pixels on ground truth signature image), PBS (pixels on signature before touching characters segmentation), ETP (extra pixels on signature due to touching characters), EPD (extra pixels deleted by the proposed approach), PND (extra pixels could not be deleted), and SPD (signature pixels deleted by the proposed approach) were used. Overall, 7.28% extra pixels were removed out of 8.47%, and 0.44% of pixels on signature were deleted by the proposed touching/overlapping printed text separation method.

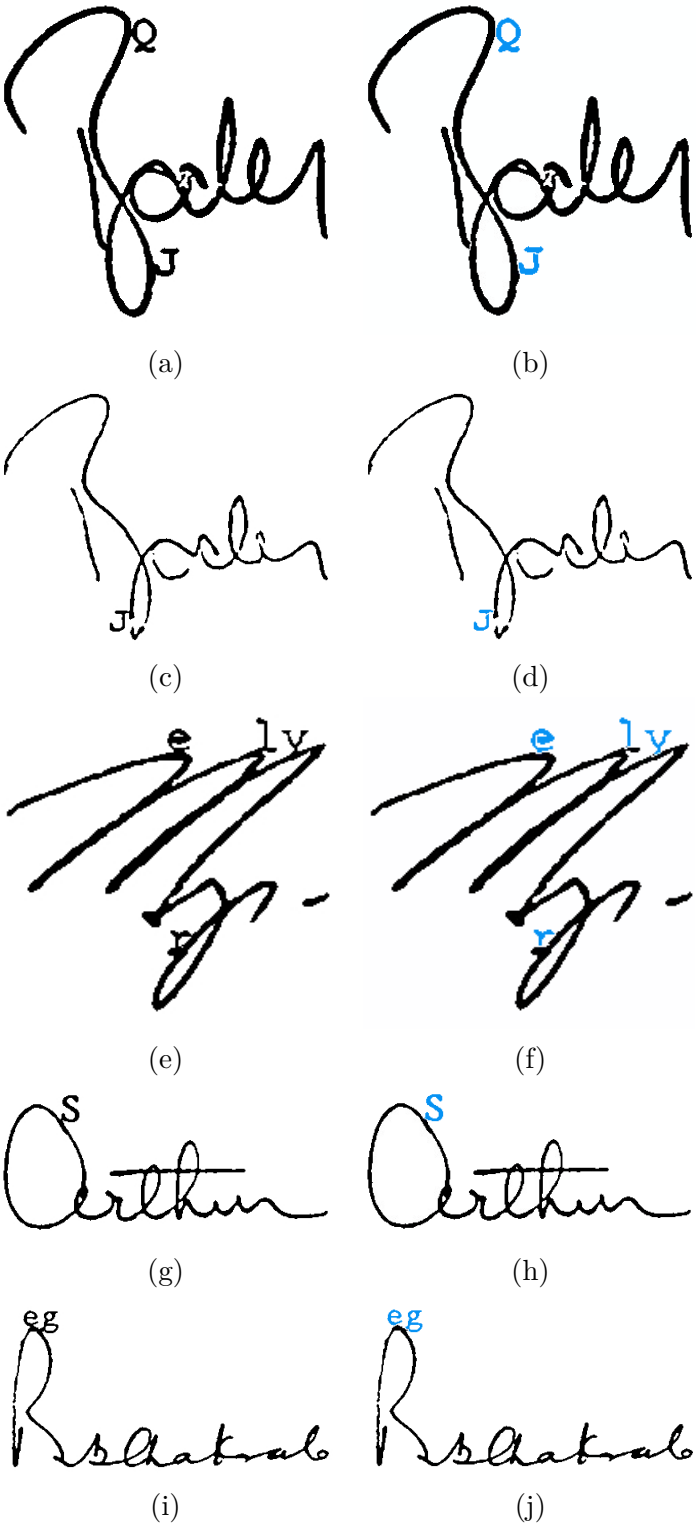


FIGURE 3.21: Images (a,c,e,g,i) showing signatures before segmentation. Images (b,d,f,h,j) show signatures after segmentation. Removed pixels are marked in blue.

3.7.3.2 Results based on Bag-of-Visual-Words

The signature detection experiments on the 'Tobacco' dataset demonstrate the excellent performance of the proposed approach. The accuracy obtained from the signature

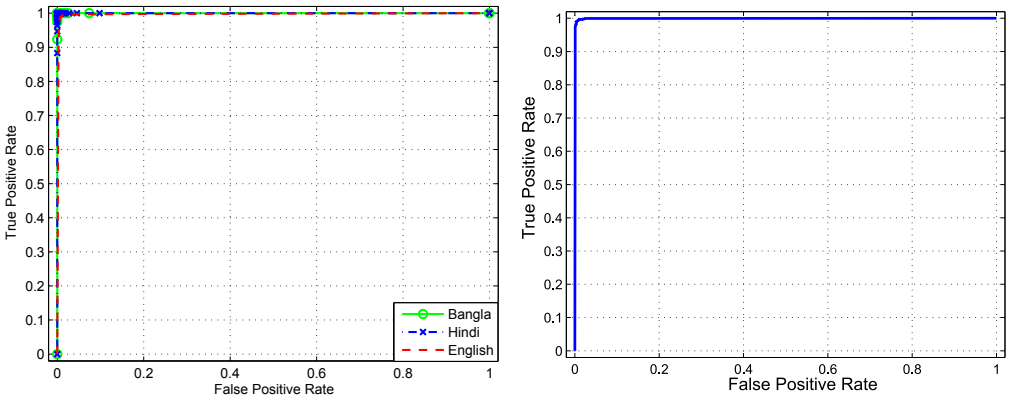
TABLE 3.4: Quantitative results of touching/overlapping separation on some signature images

Signature Sample	Number of			Percentage of		
	PGT	**PBS**	**ETP**	**EPD**	**PND**	**SPD**
S01	8128	8462	334	4.11	0.00	0.00
S02	4171	4542	371	9.40	0.00	0.55
S03	10472	11059	587	7.09	0.00	1.79
S04	1105	1167	62	6.33	0.00	1.00
S05	17434	19568	2134	9.99	2.25	0.00
S06	4253	4323	70	1.36	0.28	0.00
S07	3089	3171	82	0.00	2.65	0.00
S08	3297	3520	223	4.40	2.37	0.27
S09	2895	3191	296	7.70	2.52	0.00
S10	1159	1304	145	11.13	1.38	0.52
S11	4152	5382	1230	28.81	0.82	1.57
S12	8581	10022	1441	14.74	2.05	0.26
S13	26019	28011	1992	4.12	3.54	0.08
S14	5409	5761	352	5.14	1.37	0.09
S15	16256	17316	1060	6.16	0.36	1.54
S16	2509	2689	180	5.18	1.99	0.20
S17	16550	18507	1957	7.80	4.02	0.95
S18	4839	5132	293	6.05	0.00	0.00
S19	10974	11193	219	2.00	0.00	0.00
S20	4888	5092	204	4.17	0.00	0.00

detection experiments from English, Devnagari, Bangla and multi-script (English, Devnagari, and Bangla) combined dataset are presented in Table. 3.5. The ratio between True Positive Rate (TPR) and False Positive Rate (FPR) (i.e. Receiver Operating Characteristic (ROC) curve) obtained from the signature detection experiment is presented in Fig. 3.22. Fig. 3.22(a) shows the ROC curves obtained from the experiment on the 'Tobacco', Devnagari, and Bangla datasets. Fig. 3.22(b) shows the performance of signature/handwriting detection on the combined dataset of English, Devnagari, and Bangla.

TABLE 3.5: The signature detection performance using Bag-of-Visual-Words-based features on different scripts

Document script	**Accuracy (%)**
English	99.68
Hindi	99.94
Bangla	99.97
Multi-script	99.21



(a) (b)

FIGURE 3.22: (a) ROC curves obtained from signature/handwritten detection experiment on English (Roman), Devnagari (Hindi) and Bangla single script documents. Here, ROC curves on English, Devnagari and Bangla are almost overlapped because of the similar accuracy. (b) ROC curves obtained from signature/handwritten detection on multi-script documents of the combined dataset.

3.7.4 Performance Evaluation on Signature-Based Retrieval

The document datasets presented in Table 3.1 and Table 3.2 were used for the evaluation of the proposed system for signature retrieval. Four separate experiments were carried out on English, Devnagari, Bangla and the combined dataset of all the three scripts. Three different features based on foreground, the background, and combined information of foreground and background were used in this work. Moreover, the signature retrieval performances based on three different distances were measured for each case. Fig. 3.23, Fig. 3.24 and Fig. 3.25 show the precision-recall curves on English, Devnagari, and Bangla documents respectively using Correlation, Euclidean, and DTW-based distance measures. Fig. 3.26 shows the precision-recall curve on multi-script documents using all the three distance measures employed for the scripts individually. The experiment

shows that employed features containing combined information of foreground and background outperformed the performance of features that either contained only foreground or background information.

TABLE 3.6: The signature retrieval performance based on foreground, background and combined (foreground + background) information of signature (English scripts)

Signature information	**Precision(%)**	**Precision(%)**	Threshold
Foreground	91.84	82.57	0.63
Background	92.07	85.32	0.59
Combined	92.23	87.15	0.60

As an example, Table 3.6 shows 91.84% precision and a recall of 82.57% were obtained from the foreground information when a linear correlation threshold was 0.63. The precision of 92.07% and 85.32% recall were obtained on the same dataset using background information when the threshold for linear correlation was fixed to 0.59. Finally, 92.23% precision and 87.15% recall were obtained from the combined information of foreground and background when the linear correlation threshold was fixed to 0.60. Fig. 3.23 presents the precision-recall curves of the English script for all the thresholds.

3.7.5 Comparison with Similar Existing Systems

The previously proposed approaches on signature segmentation (or detection) and recognition were tested on different publicly available datasets such as 'Tobacco' and a few experiments were conducted on the dataset of Hindi and Bangla scripts. Table 3.7 shows the performance of the previously proposed approaches on signature detection from documents. In [12], the result was reported in two stages: signature detection and signature matching. The accuracy of 92.8% was reported on the 'Tobacco' dataset for signature detection using a multi-scale structural saliency-based [12] approach. After signature detection, signature matching was performed with a dissimilarity measure. With a combination of dissimilarity measures, the best matching accuracy MAP (Mean Average Precision) obtained was 90.5%. Though there was no report of the full signature retrieval result, theoretically, the combination of detection and matching results would provide approximately 84% (92.8% $\times$ 90.5%) MAP as 92.8% accuracy was obtained for detection and 90.5% for matching. A recall of 78.4% and a precision of 84.2% were reported by Srinivasan and Srihari [49] for the signature-based document retrieval task. A 96.13% accuracy (298 signatures were correctly identified out of 310 documents) was reported by [61] on signature detection from Arabic/Persian documents. In the previous work [80], 95.58% accuracy was achieved on signature components detection using

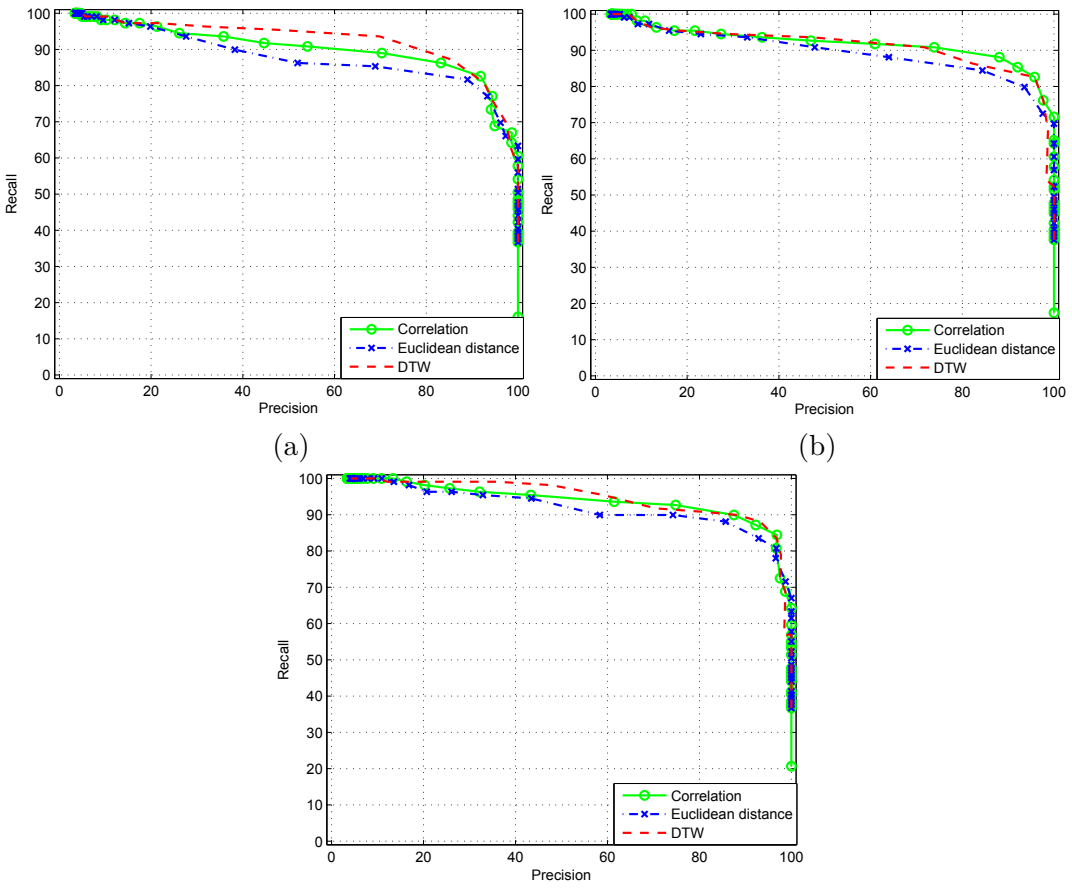


FIGURE 3.23: Precision-Recall curves of signature retrieval on English (Roman) script using (a) Foreground information (b) Background information and (c) Combined information of foreground and background. Three measures such as Correlation, Euclidean and DTW distances were computed in all the cases.

gradient-based features and an SVM classifier on the patch-wise classification of signatures and printed text from signed documents.

TABLE 3.7: Comparison of signature detection performance on 'Tobacco' document repository

Approach	**Dataset**	**Accuracy (%)**
Multi-scale structural saliency [12]	Tobacco-800	92.80
Conditional Random Field [49]	101 documents	91.20
Gradient-based feature with SVM [80]	Tobacco-800	95.58
BoVW-based feature with SVM	Tobacco-800	99.68

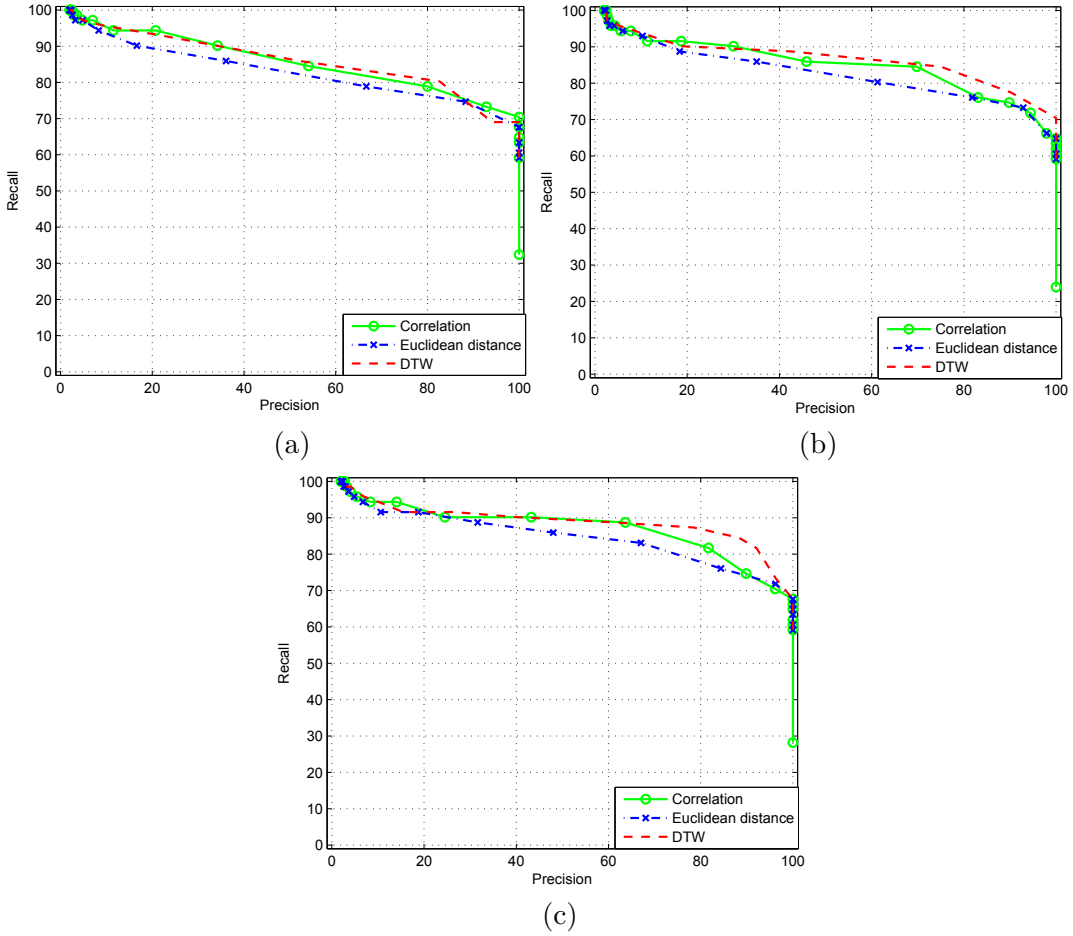


FIGURE 3.24: Precision-Recall curves of signature retrieval on Devnagari script using (a) Foreground information (b) Background information and (c) Combined information of foreground and background. Three measures such as Correlation, Euclidean and DTW distances were computed in all the cases.

Though the Bag-of-Visual-Words-based system achieves better accuracy than previously proposed approaches, the primary advantage is that the feature extraction technique is simpler and more robust than previous methods and works in a multi-script environment. The Bag-of-Visual-Words-based system does not need pre-processing or noise correction of signature portions for matching in an earlier stage. The empirical results of the experiments are encouraging and compare well with other state-of-the-art approaches in the literature.

3.7.6 Additional Experiments

Some additional experiments were conducted on synthetic noisy documents and the document retrieval based on logo information. The experimental outcomes obtained from the experiments are presented here.

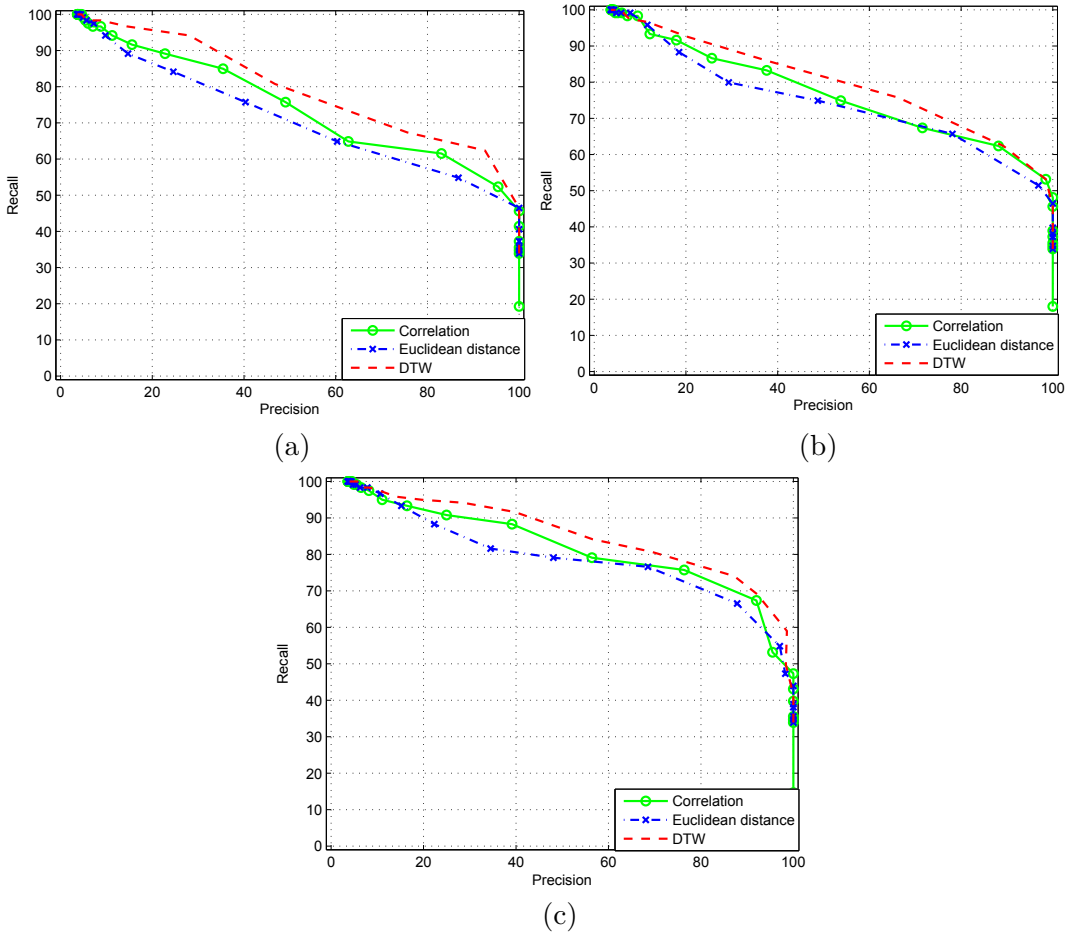


FIGURE 3.25: Precision-Recall curves of signature retrieval on Bangla script using (a) Foreground information (b) Background information and (c) Combined information of foreground and background. Three measures such as Correlation, Euclidean and DTW distances were computed in all the cases.

3.7.6.1 Experiment on Noisy Documents

To evaluate the robustness of the proposed system on noisy documents, a synthetic noisy document dataset was created. Gaussian noises of two different variances (i.e. 0.005 and 0.01) were applied on the 'Tobacco' database for this work. Fig. 3.27(a) and Fig. 3.27(b) show the same document with Gaussian noise of 0.005 and 0.01 variances respectively. The qualitative performance of signature detection results on these two sample noisy documents is shown in Fig. 3.27(c) and Fig. 3.27(d) respectively.

Fig. 3.28(a) shows the ROC curve obtained from the experiment of signature detection from noisy document images. The area under the curve was 99.91%. The accuracy was dropped by 0.74% (98.94% accuracy was obtained in contrast to 99.68% in the experiment on normal, less noisy documents) in the experiment on synthetic noisy documents. Fig. 3.28(a) and Fig. 3.28(b) shows precision-recall curves obtained from signature-based document retrieval experiments on noisy documents with different

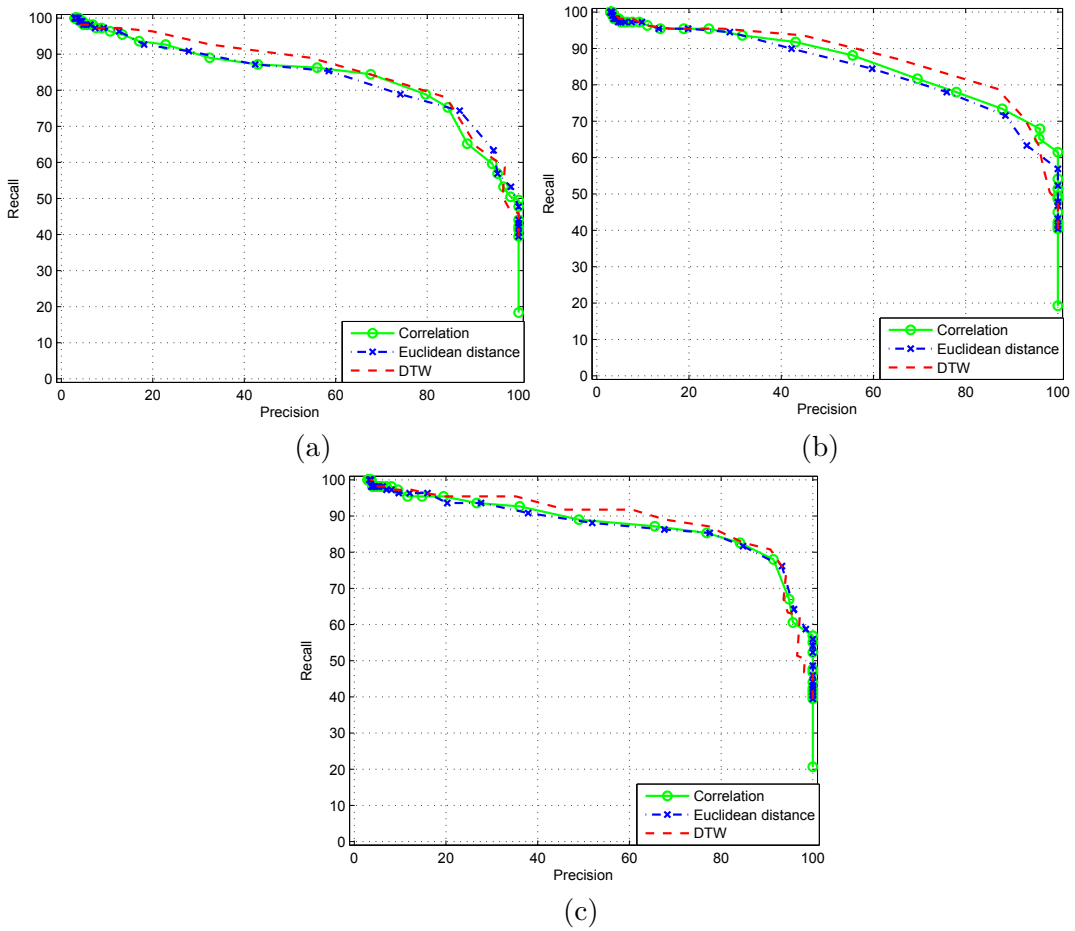
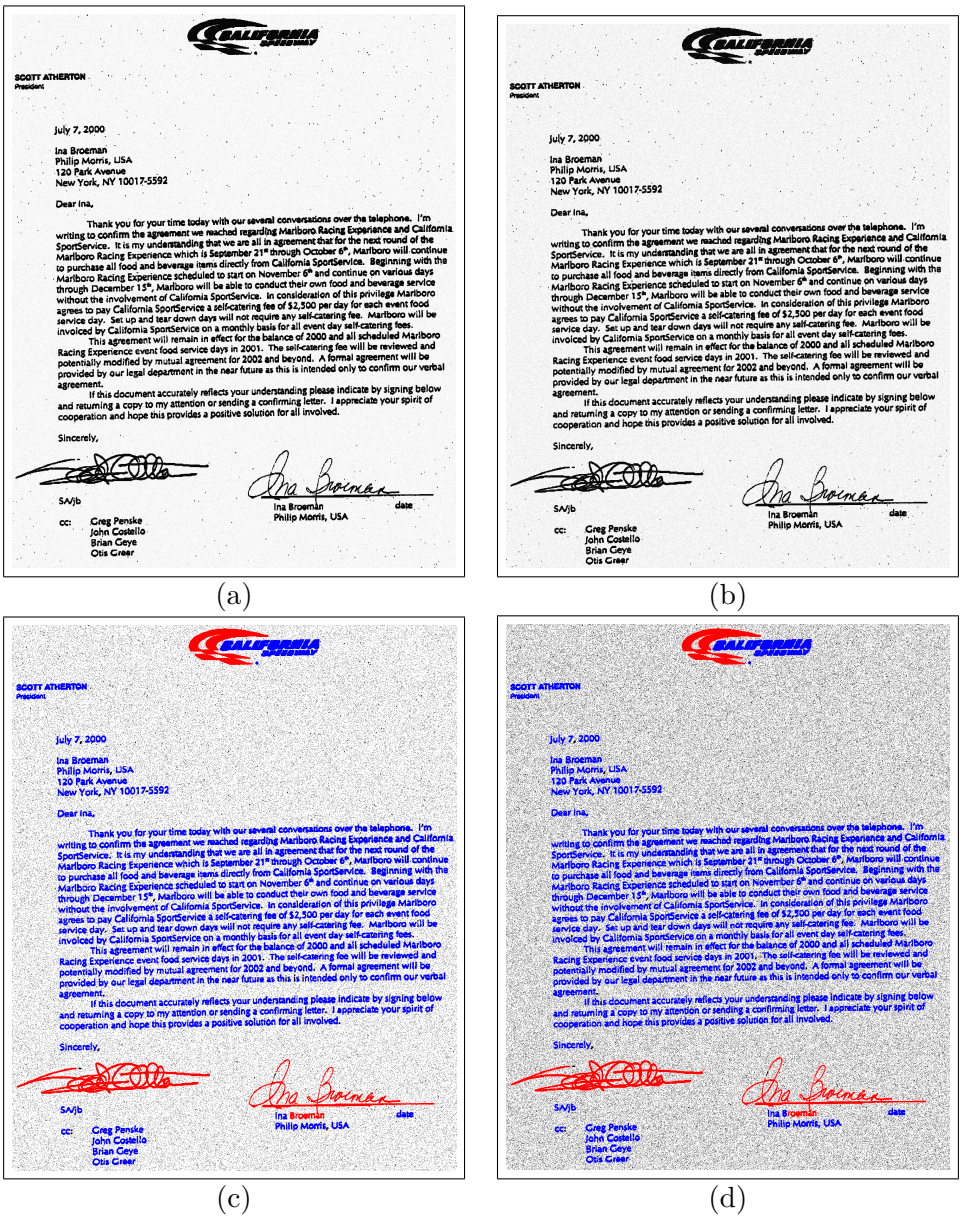


FIGURE 3.26: Precision-Recall curves of signature retrieval on a multi-script dataset (English (Roman), Devnagari and Bangla) using (a) Foreground information (b) Background information and (c) Combined information of foreground and background. Three measures such as Correlation, Euclidean and DTW distances were computed in all the cases.

Gaussian noise. In this experiment, two different variances such as 0.005 and 0.01 were used to create the synthetic Gaussian noisy document images. The performance of the system is decreased by approximately 7-8% in comparison to the less noisy documents during the retrieval stage.

3.7.6.2 Document Retrieval Based on Logo Information

As described earlier, an experiment on logo-based retrieval was performed and the outcomes of the experiments are presented using the ROC curves. Fig. 3.29(a) shows three ROC curves obtained from the experiments of logo detection from documents. The area under the ROC curves quantifies the overall performance obtained from the experiments. In the logo detection experiment, three different cases were considered. The first experiment was a two-class problem where classes contain logos and printed text and no classification errors were obtained. The second experiment also contained two



(a) (b) (c) (d)

FIGURE 3.27: Samples of English official documents after addition of Gaussian noise (a) with variance 0.005 (b) with variance 0.01 (c,d) signature detection results on the binary version of (a) and (b), respectively. Printed text and signature components are marked in blue and red, respectively.

classes. The printed and handwritten components were kept in one class and the other class contained logos. Accuracy of 99.61% was obtained from this experiment for logo detection. Finally, in the third experiment logos, printed text and signature/handwritten texts as different classes were considered and a 98.46% accuracy was achieved. It is observed that 5.5% and 1.38% of logos were confused as signature/handwritten text and printed text, respectively.

The background information of logos is not always present. Thus, the foreground

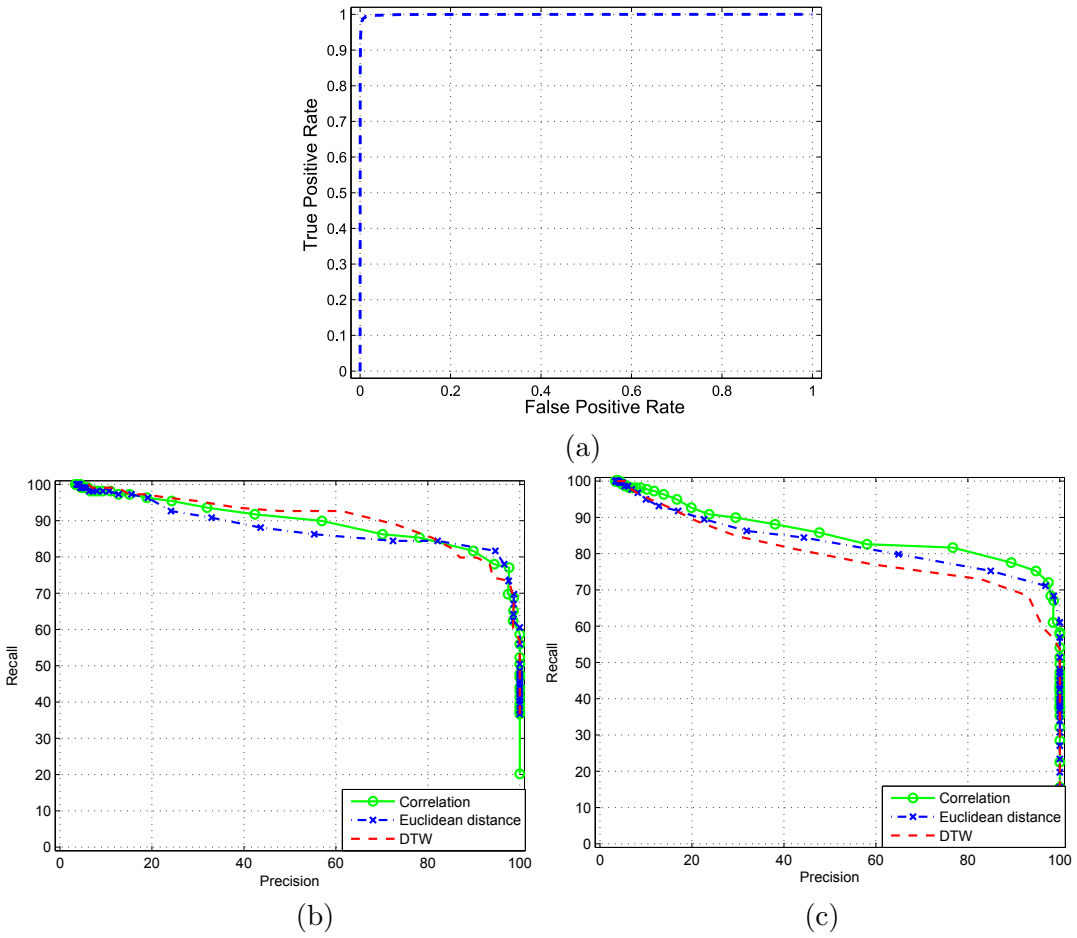


FIGURE 3.28: (a)ROC curve obtained from the experiment of signature detection from Gaussian 'tobacco' documents with noise. Precision-Recall curves of signature retrieval on Gaussian noisy dataset (b) with variance 0.005 (c) with variance 0.01. Three measures such as Correlation, Euclidean and DTW distances were taken for all the cases.

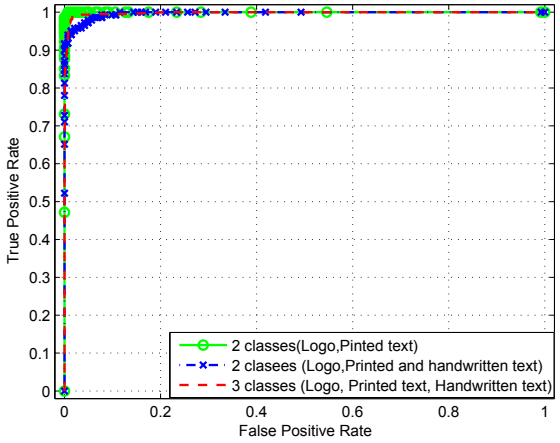


FIGURE 3.29: ROC curves obtained from the experiments of detection of logos from documents.

information is only used in the experiments of document retrieval based on logo information. The precision is always 100% for all recall values. Different recall values based on different thresholds is presented in Table 3.8.

TABLE 3.8: Threshold vs. Recall from logo-based document retrieval using three similarity measures (Correlation, Euclidean Distance, and DTW)

Similarity Measure: Correlation				
Threshold	0.30	0.25	0.20	0.15
Recall(%)	88.92	95.30	98.13	99.41
Similarity Measure: Euclidean Distance				
Threshold	1.71	1.61	1.51	1.41
Recall(%)	99.76	98.14	91.95	74.55
Similarity Measure: DTW				
Threshold	71.71	61.61	55.55	53.54
Recall(%)	99.89	99.55	90.07	76.89

Table 3.9 shows the comparative study of logo detection and recognition performance on the 'Tobacco' document dataset. The proposed approach outperformed the recently proposed approaches on logo detection and recognition. Overall accuracy of 99.50% (99.61% $\times$ 99.89%) was achieved on logo detection from 'Tobacco' dataset. Here, the best accuracy obtained from the experiments is considered for comparison with the recently proposed approaches.

TABLE 3.9: Comparison of logo detection and recognition performance on the 'Tobacco' document repository

Approach	**Detection Accuracy (%)**	**Recognition Accuracy (%)**	**Overall Performance(%)**
Alaei and Delalandre [55]	99.31	97.90	97.22
Wang [54]	94.70	92.90	87.98
Proposed BoVW-based Method	99.61	99.89	99.50

3.7.7 Error Analysis

The limitations of the system at all stages are detailed in this section. The errors obtained from detection and retrieval of signatures are presented in the following sub sections.

3.7.7.1 Signature Detection/Segmentation

Here, some errors that resulted from signature detection experiments based on gradient features are described. Most of the errors occured because of heavy touching (see Fig.3.30(d)) of signature strokes with text characters (marked by red boxes in the figures). When the signature strokes touched many characters in the printed text zone, the segmentation of characters was not always correct. Sometimes, due to the over segmentation on signature strokes a few errors occured. It is also noted that when multiple segments appeared in the printed text zone, small segmented signature strokes may be lost during removal of small isolated segmented components from text zones. In Fig.3.30(b), a portion of a signature stroke removed from a printed text zone is shown marked with a red box. Also, if some isolated strokes of signature appeared in the printed text zone, they were misclassified as text characters and thus removed from the signature.

It is observed that the proposed method wrongly identified some handwritten annotations as signatures because some signatures were very similar to handwritten annotations. It is also noted from the experiment that when a large portion of graph schematic or large non-signature strokes was present in a document, they were classified as signatures. The components of a seal in some documents were sometimes recognized as signatures as the system was not trained with such components at the signature detection stage.

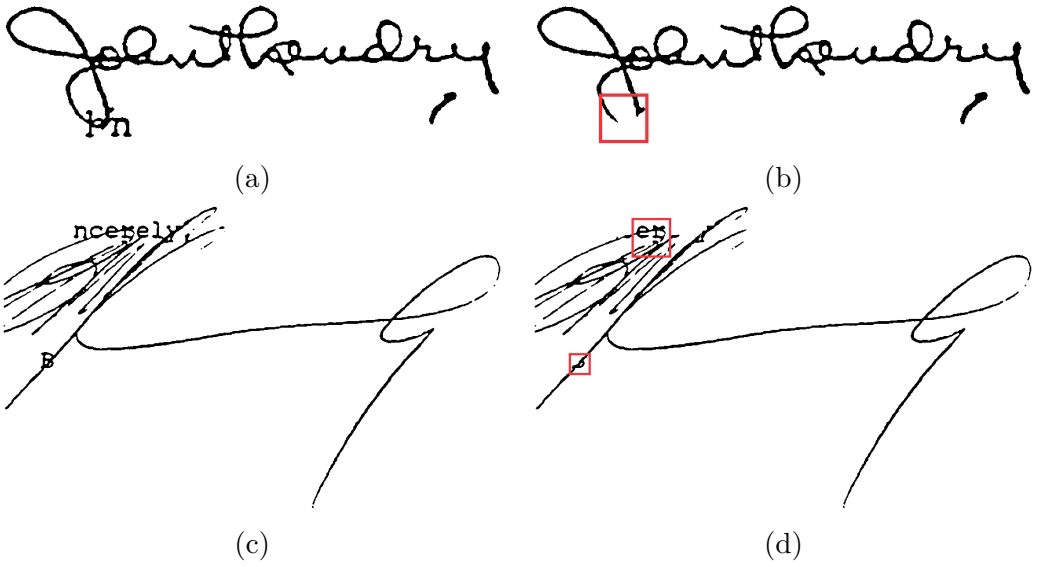


FIGURE 3.30: Examples of some erroneous results. Images (a,c) are signatures overlapped with printed characters, (b,d) are their respective segmentation results.

Some errors that resulted from signature detection experiments based on Bag-of-Visual-Words features were also found. In the signature detection stage, some printed

components such as logo, seal and figures were incorrectly classified as signature components. It is to be noted that small components such as small dots were ignored in this classification stage and the average stroke-width of components-based threshold values were used. Since samples of non-text printed components were not included in the training phase, some non-text printed components were misclassified as handwritten/signature components in the experiments. A few examples of such components are shown in Fig. 3.31.

(a) (b) (c) (d) (e)

FIGURE 3.31: Samples of logos and printed components recognised as signatures or handwritten components.

3.7.7.2 Errors in Signature Retrieval

Table 3.10 and Table 3.11 show Type I and Type II errors respectively obtained from the signature retrieval step in the experiments. The first column of Table 3.10 shows two sample query signatures and the second row of Table 3.10 shows the retrieved signatures present in the target documents. Although, query signatures and retrieved signatures belong to different classes, the correlation between query signatures and retrieved signatures were high. Likewise, Euclidean and DTW distances were low among these samples.

TABLE 3.10: Three different distances among different signature samples show Type I error (false positive) cases in the signature retrieval experiments

Signature Samples	**Correlation**		**Euclidean Distance**		**DTW**	
						
	0.606	0.604	1.141	1.140	38.665	38.705
	0.608	0.612	1.132	1.123	38.818	38.745

Similarly, Table 3.11 shows similarity measures of two different sets of signatures. Samples of query signatures are written in a slanted style, whereas signatures present in the target documents are written in a standard style. As a result, the correlation measure is low between the query signature and the retrieved signature in the target

document belong to the same class. Likewise, the Euclidean distance and the DTW distance were high. Therefore, a Type II error occurs in this case.

TABLE 3.11: Three different distances among different signature samples show Type II error (false negative) cases in signature retrieval experiment

Signature Samples	**Correlation**		**Euclidean Distance**		**DTW**	
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	0.508	0.545	1.765	1.692	44.386	42.387
[illegible]	0.528	0.557	1.725	1.666	43.673	41.669

3.8 Summary

A novel approach of a signature-based document image retrieval technique involving multi-script (i.e. English, Hindi and Bangla) offline signatures has been detailed in this chapter. The proposed method is organised into three stages namely signature detection, segmentation and signature matching. The proposed method detects signature candidates by a classification based method. Two different features namely, 400-dimensional gradient and Bag-of-Visual-Words (BoVW)-based methods with the SVM-based classifier were used for the signature detection task. Next, an approach based on BoVW powered with SIFT descriptors was applied for signature shape representation for the signature matching. The experimental results demonstrate that the proposed method outperformed previously proposed methods on document retrieval. Another important advantage of the method is that it works with multi-script documents and signatures. Considering all the erroneous cases, the scope of further improvement is possible in future work. A multi class classification approach with rejection capability would improve the signature detection performance. Type I and Type II errors obtained in the signature retrieval stage would need to be addressed in future work.

Chapter 4
Date-Based Document Retrieval

Searching and retrieval of document images from a large document repository have been a problem of interest over the last decade. Manual processing and management of such document datasets need significant effort and cost. For example, date-wise sorting of administrative documents of an organization requires significant manual effort and is time-consuming. An automatic document indexing and retrieval system based on a key information (i.e. date) can be very useful in this context. According to the reviewed literature in Chapter 2, existing quality work is not present on such an application to document analysis such as document/form reading, document indexing, etc. This indicates that there is significant scope for research for a date-based document retrieval system. The date is a key piece of information, which can be used in various applications such as date-wise document indexing/retrieval. Thus, it is necessary to extract the key information (i.e. date) from the document based on the query to retrieve the relevant documents. A date field extraction approach from document images towards date-based document retrieval system is presented in this chapter.

The chapter is organized into different sections as follows. Section 4.1 presents the various issues and problems associated with the system. A brief description of the proposed system is provided in Section 4.2. The feature extraction and classification techniques used in the proposed approach are described in Section 4.3. Section 4.4 describes the date extraction system from multi-script documents. Section 4.5 presents the algorithm of date field extraction approach using HMMs. The experimental results obtained are presented and analysed in Section 4.6. Finally, Section 4.7 presents the summary of this chapter.

Major parts of this chapter are published in the papers titled '*Multi-lingual date field extraction for automatic document retrieval by machine*', Mandal et al. [99] and '*Date field extraction from handwritten documents using HMMs*', Mandal et al. [100].

4.1 Introduction

Given a dataset of multi-script handwritten documents, developing a document retrieval system based on date field as key information would be a useful application. However, the problem would be a challenging task due to the unconstrained nature of handwritten dates. The traditional Optical Character Recognition (OCR) approach is not applicable in such circumstances because of the high degree of variability in handwritten dates. Moreover, in multi-lingual and multi-script countries such as India, retrieval of multi-script documents using the date pattern can be more challenging. An Indian state generally uses three official languages. For example, the West Bengal state of India uses Bangla, Devnagari and English (Roman) as the official languages. Thus, Devnagari, Bangla and English (Roman) scripts are often found in official documents. In this context, a typical date-based multi-script Document Image Retrieval (DIR) system has three main stages : script identification, date extraction, and date matching. A flow diagram of the date-based retrieval system is presented in Fig. 4.1.

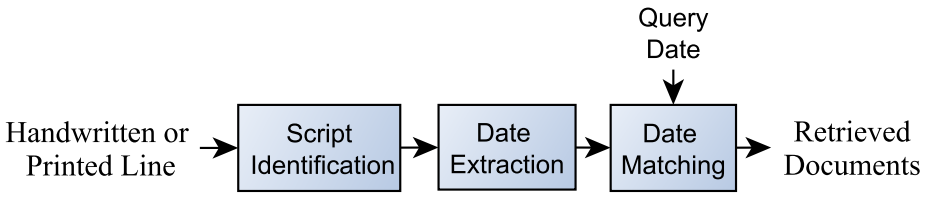


FIGURE 4.1: The block diagram of a multi-script date-based document retrieval system.

The date can be written in various formats such as numeric and semi-numeric with a variable number of components. Some of these formats of a single English date are 12/03/2012, 12th March 2012, March 12, 2012, 12-03-2012, 12.03.2012, 12.03.12, etc. Examples of Bangla dates are ১৫/০৭/১৯৯০ in dd/mm/yyyy format, ১লা বৈশাখ ১৪১১, ২রা আশ্বিন ১৪১২, ৪ঠা বৈশাখ ১৪১৪, ৫ই আষাঢ় ১৪১০, ২১ শে বৈশাখ ১৪১৪, ১লা জানুয়ারী ১৯৯০, ২রা ফেব্রুয়ারী ১৯৯০ in dd month yyyy format and ১৬-০৭-১০ in dd-mm-yy format, etc. Some sample formats of Devnagari dates are १२/०८/२००९ in dd/mm/yyyy format, २३ शे जनवरी, २०१२ in dd month yyyy format, etc.

Thus, a tri-lingual date extraction method is required which can handle the following five cases:

- A document contains a date field in Devnagari script.
- A document contains a date field in Bangla script.
- A document contains a date field in English script.
- A bi-lingual document contains a date field in Bangla or English scripts.
- A tri-lingual document contains a date field in Bangla or Devnagari or English scripts.

In addition, two types of date field patterns (numeric and semi-numeric) need to be considered for all the above cases. A date pattern is called semi-numeric date if the month information is in text form (i.e. January, Jan). Thus, a date extraction system will be very useful in searching and interpreting multi-script documents based on date field.

4.2 The Proposed Approach

To use the date as key information for date-based document retrieval, two different systems are proposed. Firstly, a segmentation-based approach is proposed for date-based document retrieval from multi-lingual (English, Devnagari, and Bangla scripts) handwritten documents. In order to retrieve the documents, the script of the document was identified, and based on the identified script, word components of each text line were classified into the month and non-month classes using word-level feature extraction and classification. Next, non-month words were segmented into individual components and labelled into one of text, digit, punctuation or contraction categories. Subsequently, the date patterns were searched using the labelled components. Both numeric and semi-numeric regular expressions were used for the date extraction. Dynamic Time Warping (DTW) and profile feature-based approaches were used for classification of month/non-month words. Other date components such as numerals and punctuation marks were recognised using a gradient-based feature and Support Vector Machine (SVM) classifier. The experiments were performed in English, Devnagari, and Bangla document datasets.

In the second method, a segmentation-free date extraction approach based on Hidden Markov Model (HMM) is proposed for date-retrieval and then an SVM-based classification approach is applied to refine the results obtained from HMM-based recognition system. The system handles the segmentation problem which is encountered in the first system in a better way. However, the system has not been adapted for multi-script documents because of time constraints. The experiments were performed on the English handwritten document dataset and the encouraging results obtained from the

approach indicate the effectiveness of the proposed system. Three different classification techniques and four different feature extraction approaches were applied to design the systems and the following section details all the methods.

4.3 Feature Extraction and Classification Techniques

In this section a brief description of the various feature extraction and classification techniques used in the proposed methods are detailed. The feature extraction techniques namely, word profile features, 400-dimensional gradient features, Local Gradient Histogram and Histogram of Oriented Gradients are discussed. The different classifiers namely, Nearest Neighbours (NN) Classifier, Support Vector Machines (SVM) and Hidden Markov Model (HMM) which were considered for the classification task are also discussed in this section.

4.3.1 Word Profile-Based Feature

Word profiles capture part of the outlining shape of a word. Word profile-based information was used here as the word-level feature. Four components such as vertical projection profile, upper profile, lower profile and vertical crossing (vertical ink transition) were extracted as discussed in [19] for word-level features. A thresholding technique was applied on the image for identification of foreground pixels which are found to be sufficient for this study's purposes. The vertical projection profile is the summation of pixels vertically in an image. The distance between the upper boundary and the closest foreground pixel is considered for the upper profile and the distance between the lower boundary and the closest foreground pixel was considered here for the lower profile features. The vertical crossing is the number of foreground and background transition in an image. The value of all the profile features stated above were normalized to the range[0-1]. Profiles of a sample Bangla word are presented in Fig. 4.2.

4.3.2 400-Dimensional Gradient-Based Feature

The 400-dimensional gradient-based feature has already been presented in Section 3.3.1 of Chapter 3 and the same approach was followed for this experiment.

4.3.3 Local Gradient Histogram (LGH)

Off-the-shelf LGH feature [101] is a feature based on local information that has similarity with Lowe's SIFT descriptor [86]. In this feature extraction technique, a sliding window

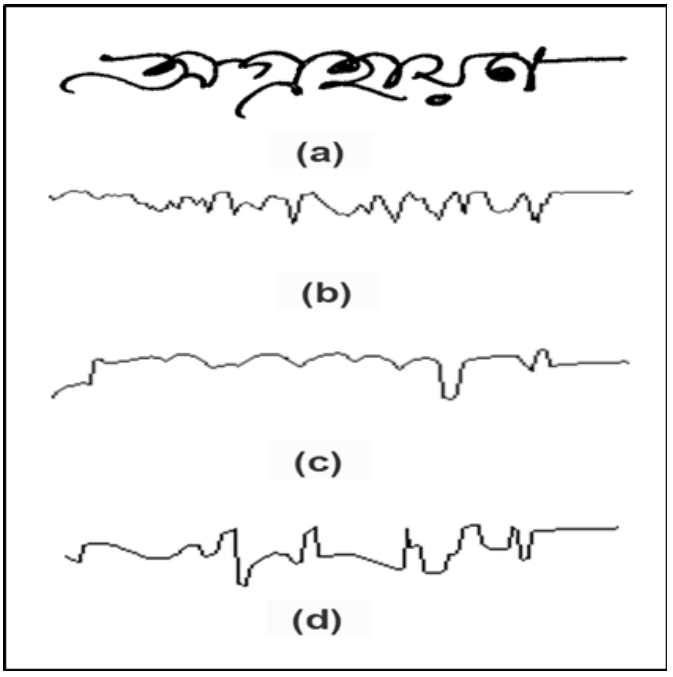


FIGURE 4.2: Profile projection of a Bangla word (a) Original image (b) Vertical projection profile (c) Upper profile (d) Lower profile

traverses the image from left to right in order to produce a sequence of overlapping sub-images.

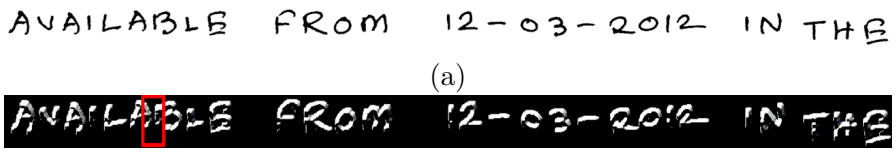
Given an image $I(x, y)$ of height H and width W, a center at each column x a window of height H and width w is considered. At each window position, a feature vector is computed that only depends on the pixels inside the window. Thus, a sequence of W feature vectors is obtained. One advantage of this sliding window approach is that it preserves the left-to-right nature of the writing.

It is reported that the LGH features [101] performed better than the previously proposed approaches. To obtain LGH features the following steps are performed.

- At first, a Gaussian filter is applied to the image $I(x, y)$ to obtain the smoothed image L(x, y).
- From the L, the horizontal and vertical gradient components G_x and G_y are determined as $G_x(x, y) = L(x+1, y)-L(x-1, y)$ and $G_y(x, y) = L(x, y+1)-L(x, y-1)$.
- Next, the gradient magnitude m and direction θ are obtained for each pixel with coordinates (x, y) as $m(x, y) = \sqrt{{G_x}^2 + {G_y}^2}$ and $\theta(x, y) = \tan^{-1}(G_y/G_x)$.

A sliding window of fixed width traverses the image from left to right to obtain a sequence of overlapping sub-images of word parts. Fig. 4.3 shows an original handwritten line along with the line after gradient computation. Each sub-image is further reduced

to the region actually containing pixels, and this region is divided into 4×4 regular cells. The gray scale LGH features are extracted from each cell. The field vector $\vec{G} = (G_x, G_y)$, is divided into L bin histogram. Each bin specifies a particular octant in the angular radian space. From all the pixels in each cell, a histogram of gradient orientations is constructed where orientations $T = 8$. Each pixel contributes to the closest bin with an amount $m(x, y)$. Alternatively, the two closest orientations can share the amount $m(x, y)$ as determined by a linear interpolation to reduce the impact of aliasing noise. The concatenation of the 4×4 histograms of 8 bins gives rise to a 128-dimensional feature vector for each window position (see Fig. 4.4). Each feature vector is scaled to have norm 1. This operation can be related to local contrast normalization and significantly improves performance in practice.



(a)

(b)

FIGURE 4.3: (a) Original handwritten line (b) Handwritten line after gradient computation (red rectangle represents a sliding window)

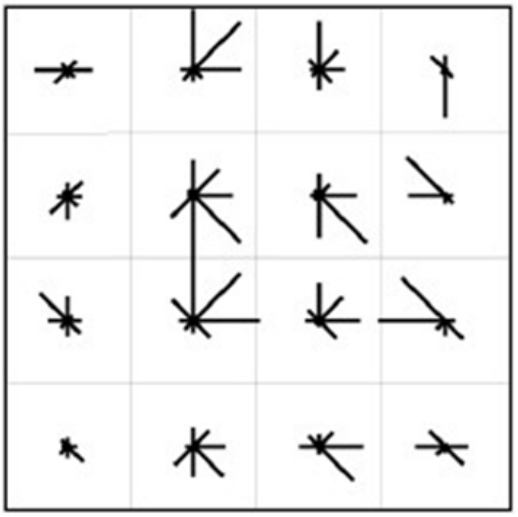


FIGURE 4.4: A sample gradient strength of 4×4 grids from a sliding window image

4.3.4 Histogram of Oriented Gradients (HOG)

HoG [102] is a robust feature descriptor which is very popular in computer vision and image processing for object detection. Dalal and Triggs [102] first described the HoG descriptors and primarily focused on pedestrian detection in static images. The basic

idea behind the HoG descriptor is that the shape and appearance of the object within an image can be described by the intensity gradient distribution or the edge directions. The HoG descriptors are typically computed by dividing an image into small spatial regions called 'cells'. A histogram of the gradient direction of the pixels within the cells is computed. The histogram bins/channels are evenly spaced over 0° to 180° or 0° to 360° based on the usage of signed or unsigned gradient values. The features are produced by combining the histogram of all the cells. HoG features suit the problem well because it operates on the localized cells and is capable of describing the shape and appearance of the handwritten numerals in the present context. Here, 8 bin/orientations are considered over 7×7 blocks for feature extraction, which resulted in a 392-dimensional feature vector.

4.3.5 Nearest Neighbours Classifier (NN)

Perhaps the most straightforward classifier in the arsenal of machine learning techniques is the Nearest Neighbour classifier where classification is achieved by identifying the nearest neighbours to a query example and using those neighbours to determine the class of the query. This approach to classification is of particular importance today because issues of poor run-time performance are not such a problem now with the computational power that is available. Amongst the various methods of supervised statistical pattern recognition, the Nearest Neighbour (NN) rule achieves consistently high performance, without a prior assumption about the distributions from which the training examples are drawn. A new sample is classified by calculating the distance to the nearest training case; the sign of that point then determines the classification of the sample. The k-NN classifier extends this idea by taking the k nearest points and assigning the sign of the majority. It is common to select k small and odd to break ties (typically 1, 3, 5 or 7). Larger k values help reduce the effects of noisy points within the training data set, and the choice of k is often performed through cross-validation. The distances can be calculated using one of the distance measures such as Euclidian, Mahalanobis, and City-block. Here, the Euclidian distance measure is used for experimentation.

4.3.6 Support Vector Machine (SVM)

The detail description of SVM [88] classifier has been given in Section 3.3.5 of Chapter 3. The Gaussian kernel SVM outperformed other non-linear SVM kernels, thus, the recognition results presented of this experiment are based on the Gaussian kernel only.

4.3.7 Hidden Markov Model (HMM)

Hidden Markov Model (HMM) is a statistical Markov model in which the systems being modelled are assumed to be a Markov process with unobserved (hidden) states. The HMM can be considered as the simplest dynamic Bayesian network. In a regular Markov model, the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters. In Hidden Markov Model, the state is not directly visible, but output, dependent on the state, is visible. Each state has a probability distribution over the possible output tokens. Therefore, the sequence of tokens generated by an HMM gives some information about the sequence of states. Note that the adjective 'hidden' refers to the state sequence through which the model passes, not to the parameters of the model; even if the model parameters are known exactly, the model is still 'hidden'.

The HMMs are especially known for their application in temporal pattern recognition such as speech, handwriting, gesture recognition, part-of-speech tagging, musical score following, partial discharges and bioinformatics. The Hidden Markov Model can be considered a generalization of a mixture model where the hidden variables (or latent variables), which control the mixture components to be selected for each observation, are related to a Markov process rather than independent of each other. A key advantage in handwriting modeling is that HMMs can cope with variable-length data and non-linear time deformations. The technical details on HMMs can be found in [75, 103]. In practice, an HMM can be employed to represent a whole word or, alternatively, sub-word units such as characters which can be concatenated to form general strings or an expression.

The feature vector sequence is processed using left-to-right continuous density HMMs [104]. One of the important features of HMMs is the capability to model sequential dependencies. An HMM can be defined by initial state probabilities π, a state transition matrix $A = [a_{ij}], i, j = 1, 2, \cdots, N$,where a_{ij} denotes the transition probability from state i to state j and output probability $b_j(O_k)$ modelled with a continuous output probability density function. The density function is written as $b_j(x)$, where x represents a k-dimensional feature vector. A separate Gaussian Mixture Model (GMM) is defined for each state of the model. Formally, the output probability density of state j is defined as

$$b_j(x) = \sum_{k=1}^{M_j} c_{jk} \mathcal{N}(x, \mu_{jk}, \sum jk) \tag{4.1}$$

where, M_j is the number of Gaussians assigned to j, and $\mathcal{N}(x, \mu_{jk}, \sum)$ denotes a Gaussian with mean μ and covariance matrix $\sum$ where c_{jk} is the weight coefficient of the Gaussian component k of state j. For a model λ, if $\mathcal{O}$ is an observation sequence $\mathcal{O}$ = ($\mathcal{O}_1$, $\mathcal{O}_2$, $\cdots$, $\mathcal{O}_T$) which is assumed to have been generated by a state sequence $\mathcal{Q}$ = ($\mathcal{Q}_1$, $\mathcal{Q}_2$,...,$\mathcal{Q}_T$), of length T, the observation probability or likelihood is calculated as follows:

$$P(\mathcal{O}, \mathcal{Q}|\lambda) = \sum_{Q} \pi_{q1} b_{q1}(\mathcal{O}_1) \prod_{T} a_{qT-1qr} b_{qT}(\mathcal{O}_T) \tag{4.2}$$

where μ_{q1} is the initial probability of state 1.

4.4 Segmentation-Based Approach for Date Extraction

A multi-stage date-field extraction approach is proposed in this section for date-based multi-script document retrieval. A block diagram of the proposed system is shown in Fig. 4.5. First, scripts were identified using foreground and background information which is described in Section 4.4.1. Water reservoir-based technique was employed to extract foreground and background information. Top and bottom reservoirs information of words was used to segment the words into primitive segments and those segments were classified using Support Vector Machine (SVM) [88] as Bangla, Devnagari or English primitive segments. Based on the majority of classified primitive segments, the script was identified. Once the script was identified, the system was trained with respective models based on the identified script. For each script, the system was trained with two (i.e. month and digit) models for classification in two stages. In the second stage, month and non-month handwritten word blocks were separated. For this purpose, words blocks were extracted using morphological operation and the segmented word blocks were classified into month and non-month classes using word block level feature analysis. The third stage of the system performed component analysis for each non-month handwritten word blocks. Isolated digits, punctuations and alphabets were identified using component level feature analysis and recognition. The components with low recognition confidence were analyzed further for touching segmentation [105]. Dynamic Time Warping (DTW)-based [19] technique was used for word block classification and SVM [88] was used in component level classification.

Finally, numeric and semi-numeric (contains month field as text string) date patterns were searched from the sequence of the labelled components. To do so, candidate lines were selected first using a voting approach and next a regular expression analysis was used to detect the date patterns. The proposed tri-lingual date extraction method can

handle five cases such as Devnagari documents with only Devnagari or English date-fields, Bangla documents with only Bangla or English date-fields and English documents with English date-fields. Two types of date-field patterns (Numeric and Alpha-numeric) were considered for all the above cases. Hence, the date extraction system will be very useful in searching and interpretation of multi-script documents.

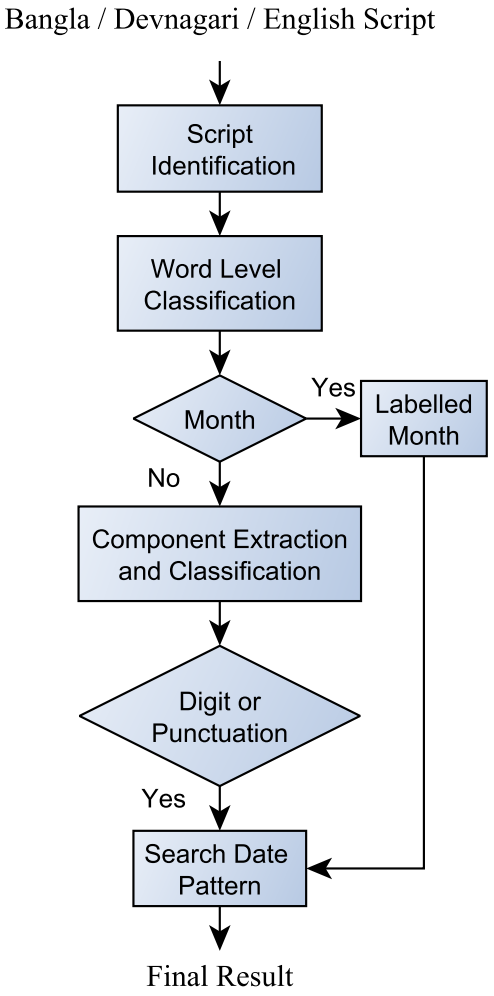


FIGURE 4.5: Block diagram of the proposed multi-script date extraction system.

4.4.1 Script Identification

A global histogram-based Otsu binarization [90] method was applied to convert the grayscale document images into binary images. Next, a smoothing algorithm [106] was used to remove noisy pixels and irregularities. The proposed date retrieval approach searches the date patterns in text line images. Hence, the binary document

was segmented into individual text lines using a line segmentation algorithm [107] and segmented lines were used for the experiment.

For line segmentation, at first the water reservoir-based [96] information was used to determine the height of text lines present in a document. The water reservoir is a metaphor to illustrate the cavity region of a component. The water reservoir principle is as follows. If water is poured from a side of a component, the cavity regions of the background portion of the component where water will be stored are considered as reservoirs of the component. The horizontal Run Length Smearing Algorithm (RLSA) was then applied on the input image. The threshold of RLSA was computed based on the height of text lines. Next, the foreground components of the RLSA applied image were eroded to generate a few seed components from the individual words of the document. Actually, the central gravity of individual words of a text line was represented by the seed components. The erosion method also helps to reduce the touching effect of character modifiers and makes the line segmentation task easier. To find the upper and lower boundary information of a text line, the erosion technique was also applied on the background region of the image. Finally, the positional information of the seed components and the boundary information was used to segment the individual lines.

Both foreground and background information was used for script identification. A water reservoir-based [96] technique was employed here for identification of English, Devnagari and Bangla scripts. Top and bottom reservoirs were used for the segmentation of the words into primitives. By top (bottom) reservoirs of a component means that the reservoirs obtained when water is poured from the top (bottom) of the component. A bottom reservoir of a component is visualized as a top reservoir when water will be poured from the top after rotating the component by 180°. Reservoir-based primitive segments (foreground) and reservoir blobs (background) were used as primitive components of these three scripts (see Fig. 4.7).

A word was segmented into a few primitive components. Segmentation points for primitives of a word are nothing but the bottommost (topmost) points of top (bottom) reservoir in a word. For primitive segmentation, the image was segmented vertically at the segmentation points. The reservoir blobs (filled with a red in Fig. 4.7) were also extracted from top and bottom reservoirs. Primitives segments and the reservoir blobs were both used for script identification.

Now, all the primitive segments and the reservoir blobs in a script were classified separately for script identification. A flow diagram of the proposed script identification system is given in Fig.4.6. A 400-dimensional gradient-based features [108] were extracted from the primitive segments and the reservoir blobs, which were subsequently fed to an SVM [88] (Gaussian kernel with Radial Basis Function) for script identification. The

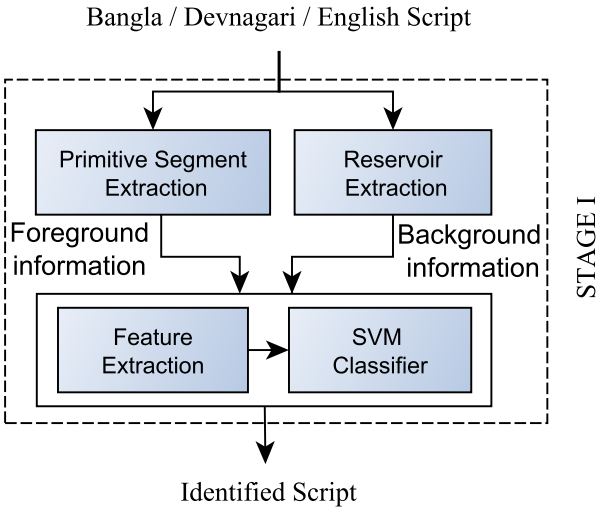


FIGURE 4.6: Flow diagram of the proposed script identification system.

400-dimentional gradient-based feature extraction technique and the SVM-based classifier used in the proposed script identification method are described in Section 4.3.2 and Section 4.3.6.

The primitive segment classifier was trained with the training data consisting of English, Devnagari and Bangla primitive segments. Likewise, a reservoir blobs classifier was also trained. Next, recognised primitive segments were counted as English, Devnagari or Bangla primitive segments and the recognized reservoir blobs were also classified into three script classes. The resultant script was identified on the basis of a majority voting of these elements. Initially the primitive segments and reservoir information were used separately on each word for its identification. Later such individual identification results were combined jointly to get final results.

For an example of combinations, let an input Devnagari word sample contains 6 primitive segments and 4 reservoirs. Out of these 6 primitive segments, let 4 primitives be classified as Devnagari and 2 primitives be classified as Bangla. Also, out of the 4 reservoirs, let 2 reservoirs be classified as Devnagari, 1 reservoir be classified as Bangla and 1 reservoir be classified as English. Since, 4 primitive segments and 2 reservoirs are classified as Devnagari, 6(4+2) is the combined count of primitive and reservoir for Devnagari. Similarly, 3(2+1) is the total count of primitive and reservoir for Bangla as 2 primitive segments are classified as Bangla and 1 reservoir is classified as Bangla. Finally, the word is recognised as Devnagari as the majority of elements (6) of the input word are classified as Devnagari.

FIGURE 4.7: Two words for each of the three different texts of (a,b) English (c,d) Devnagari and (e,f) Bangla are presented. The source image, reservoirs (reservoir blobs) from top, reservoirs from bottom and segmented points are given in left to right then top to bottom order. Reservoir blobs are marked in red and the segment points are marked by red circles.

4.4.2 Extraction of Date Field

A spotting-based approach is proposed here to find the date field in a document. The proposed date retrieval approach searched the date patterns in individual text lines of a document. First, different components of a date such as textual month, numeral,

punctuation etc. were found and this information was used to extract the date patterns. The different types of date formats were searched and the approach performed to locate those components, as well as the complete date field, are described in this section.

4.4.2.1 Components of Date

A date field in documents may have four types (numeral, textual month, punctuation and contraction) of components. Date patterns can be classified into two categories (numeric and semi-numeric dates) on the basis of different component types.

Numeric Date: A date field consisting of numerals and punctuations (Examples of numeric dates are 12/05/2010, 2/5/2010, 2-5-10 in English; ১২/০১/১৪১৮, ১-৭-১১ in Bangla; १२/०८/२००९, ३१-१२-११ in Devnagari etc.) are considered as numeric date fields. It was observed from the dataset that the total number of components in a numeric date field can vary from six to ten (if a date is written as 1/1/14 then the number of components of this date is six. If a date is written as 01/01/2014 then the number of components will be ten). The following date regular expression represents the valid formats of a numeric date:

$$(d|dd)(/.-)(d|dd)(/.-)(dd|dddd)$$

where, d represents a digit. A date has three parts or fields: date field, month field, and year field. A complete numeric date field consists of a single digit or double digit date information, single digit or double digit month information and double digit or four digit year information. Moreover, a numeric date field must have two punctuation marks to separate day, month and year information.

Semi-Numeric Date: Other date fields that consist of textual month (examples are January, জানুয়ারী and जनवरी in English, Bangla and Devnagari scripts, respectively), digit and contraction (examples are st, nd, rd in English; রা, ঠা, ই in Bangla; शे, इ in Devnagari etc.) are considered as semi-numeric dates. Examples of semi-numeric dates are ২১ শে জুলাই, ২০১০; জুলাই ২১, ২০১০ in Bangla; २३ शे जुलाई, २०१२ in Devnagari and 31st March 2011 in English etc. For semi-numeric date field extraction, the following regular expressions were searched:

$$(md|mdd)(.,-)(dd|dddd) \text{ and}$$
$$(d|dd)(contraction)(month)(.,-)(dd|dddd)$$

where, m represents a month field and d represents digit or numeral. There are two types of sequence for semi-numeric date fields depending on the position of the textual month field (Examples of date with month information in the middle position: 15th

December, 2012 and month information in the starting position: December 15, 2012). In a semi-numeric date pattern, textual month information may present in the front or in the middle of the sequence.

4.4.2.2 Month/Non-Month Identification

A conventional profile-based feature extraction and a classification process were used to identify a word as a textual month. The system performed classification on the extracted words from a line. The proposed classification technique worked as a two-class problem: month and non-month word block identification. To train the classifier, data sets with different types of month formats (examples are: জানু:, জানুয়ারী, বৈশাখ in Bangla; वैशाख, जनवरी in Devnagari and JANUARY, January, JAN in English etc.) that appear in date patterns of Bangla, Devnagari and English documents were used. Three different training datasets were used to train the classifier for the three different scripts. Once the script was identified, the respective training model was used for this purpose. The flow diagram of the month identification system is shown in Fig.4.8. The details of profile-based feature extraction procedure is described in Section 4.3.1 and the DTW-based matching steps used for this month identification process is explained as follows.

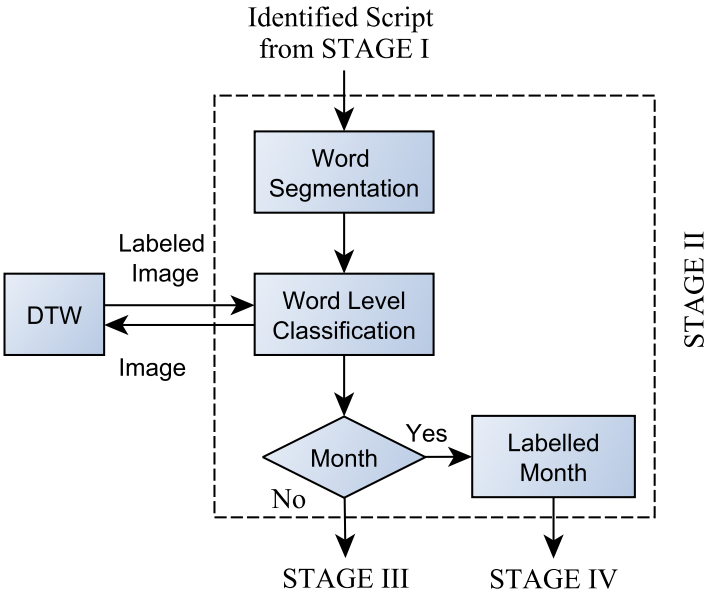


FIGURE 4.8: Flow diagram of the month-word identification system.

Classification of Month Words: The DTW-based approach applied in this experiment is similar to the method described in [19]. A DTW-based model was used for identifying month/non-month words in a text line. The similarity between two month

sequences was measured using the DTW technique. The sequences are "warped" non-linearly in the time dimension to determine a measure of their similarity independent of certain non-linear variations in the time dimension. This technique has been used widely in many applications such as speech, signatures, robotics etc. A word image was represented as 4 sequences, which were computed from different profiles of a word as described in Section 4.3.1. The DTW-based technique for measuring similarity between two sequences uses the Sakoe-Chiba band [109] to speed up the computation. The width of the word was used as a pruning criterion(i.e. the width of the word image must not be more than double the width of the other).

Here DTW was used on two signals of 4 sequences of features (f_k, where $1 \leq k \leq 4$): the upper and lower profile features, vertical projection profile and background to foreground transition of words. The DTW distance between two signals I1 and I2 was calculated using a matrix D. Where

$$D(i,j) = min\begin{pmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{pmatrix} + d(x_i, y_i) \tag{4.3}$$

$$d(x_i, y_i) = \sum_{k=1}^{4} (f_k(I_1, i) - f_k(I_2, j))^2 \tag{4.4}$$

The matching distances from 4 features of a word were added to get their cumulative distance and this cumulative distance was considered as the final matching cost of the word. Finally, this matching cost was normalized by the length of the warping path. The DTW distance and a Nearest Neighbour-based classification technique were used for month field spotting in the system. Fig. 4.9 shows some samples text lines that were classified as month and non-month blocks.

4.4.2.3 Character Level Component Identification

The words which were classified as non-month in the earlier stage were considered for the next stage. Connected component analysis was employed to segment the non-month words into different components, and component-wise classification was undertaken to extract the character/digit/punctuation components from these non-month words. Fig.4.10 shows the flow diagram of component identification system. For this purpose, connected components were fed to a component level classification process. A 400-dimensional gradient-based features [108] were extracted from the character components, which were subsequently fed to an SVM classifier [88] (Gaussian kernel with

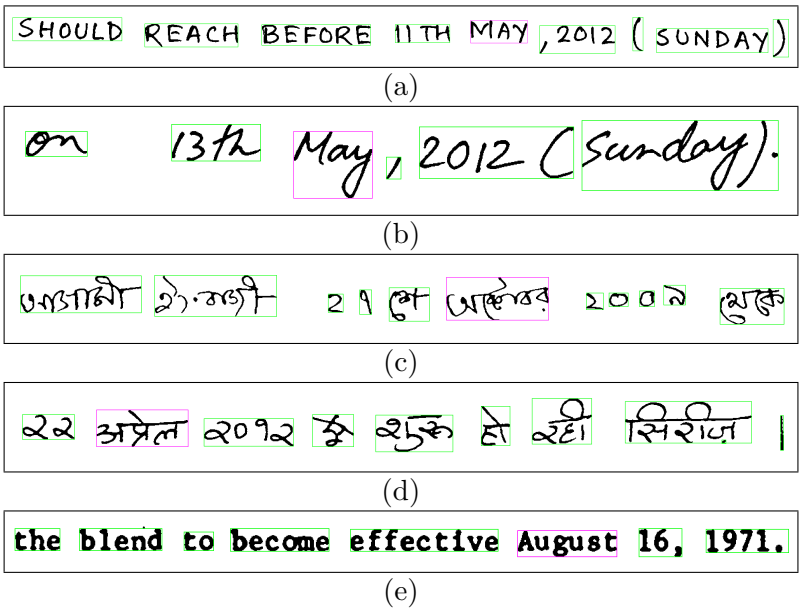


FIGURE 4.9: Text lines showing detected of month blocks (a,b) English handwritten (c) Bangla handwritten (d) Devnagari handwritten (e) English printed. In the figures, month, non-month blocks are marked in pink and green, respectively.

Radial Basis Function) for classification of character components in a word. The details of the feature extraction and classification processes are presented in Section 4.3. Through the classification process, the components were mainly classified into "digit", "punctuation", "contraction" or "text" label (see Fig. 4.11). There were some touching components available and these components could not be classified properly at this stage. Hence, the components with high recognition confidence were accepted and directly considered for date pattern matching. The rest of the components with low confidence were selected for touching component segmentation analysis. Some isolated characters which were not identified properly needed to be addressed in the touching character handling step, which is explained in the next subsection.

4.4.2.4 Touching Character Segmentation

Some touching digits or characters may exist in a component. Components with a low confidence score at the earlier stage of recognition were considered as touching and chosen for segmentation. Here, a dynamic programming-based touching character segmentation scheme [105] was used. First, different cavity regions from the touching characters were identified. The cavity regions were obtained using the Water Reservoir concept [96]. The top and bottom reservoir analysis was used to find the cavity regions in a touching component. A set of candidate segmentation points was obtained from these regions using cavity region analysis. Next, the touching component was segmented into these candidate points to find different sub-images. Using the dynamic programming, the

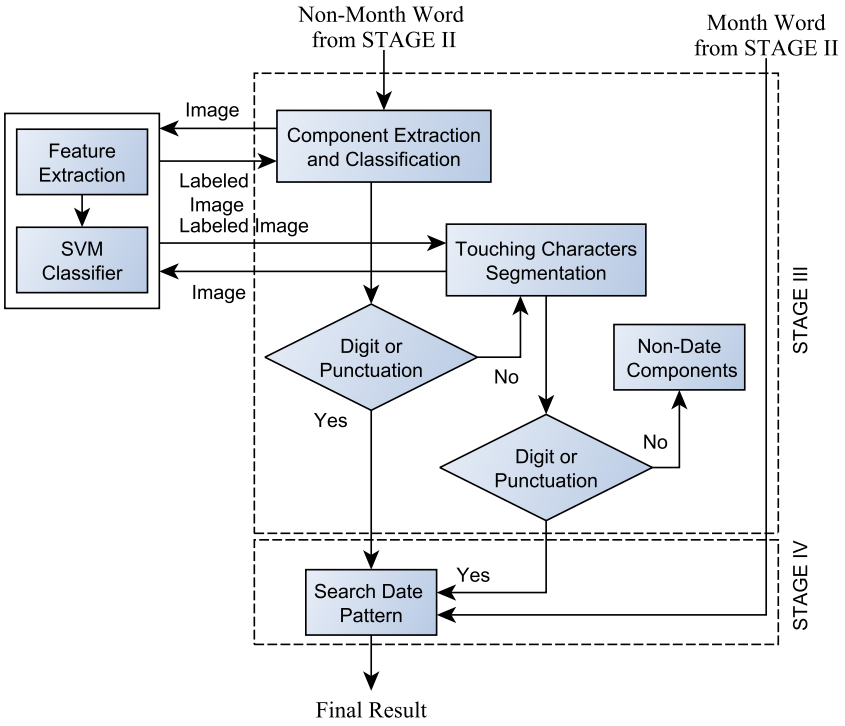


FIGURE 4.10: Flow diagram of the character level component identification system. Inputs to this system come from STAGE II.

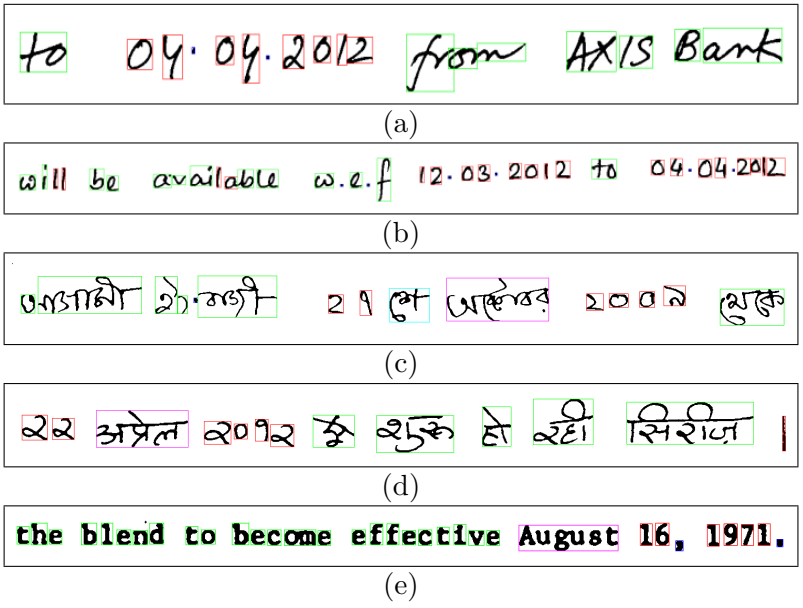


FIGURE 4.11: Component classification results from handwritten lines into digit, punctuation, contraction and text of (a,b) English handwriting (c) Bangla handwriting (d) Devnagari handwriting (e) English printed. Digit, punctuation, contraction and text are marked in red, blue, aqua and green, respectively.

recognition confidence of sub-images were analysed and the optimum segmentation path found. Finally, based on the segmentation lines, the touching component was segmented.

In Fig. 4.12, the circle shows the segmentation result of two touching digits 2 and 0.

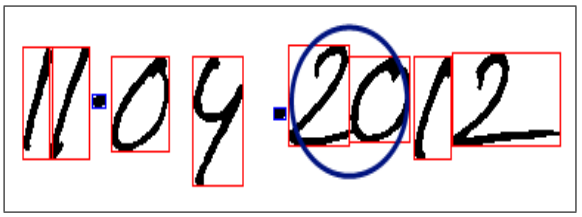


FIGURE 4.12: A segmentation result of touching digits 2 and 0. Segmented digits are marked by a blue circle.

4.4.2.5 Searching of Date Pattern

Text lines with their four different types of recognized components (month, digit, punctuation, and text) were considered for date pattern detection at this stage. The approach was divided into two parts: candidate line selection and date pattern searching.

Candidate Line Selection: The text lines that contain labelled months, digits and punctuations were selected for further analysis. For this purpose, the total number of digits, punctuation marks, and month strings of a text line were counted. Depending on the value of the counts of date elements, searching the date patterns in that text line was continued. Valid date patterns in the experimental dataset should contain at least six (examples are ৪-৬-৯০ in Bangla; ३-४-१२ in Devnagari ; 'Jan 5th, 12' etc.) elements. If the total number of date elements in a line was greater or equal to six then the line was considered as a candidate line and used for date pattern searching.

Date Pattern Searching: The components in each candidate text line were sorted in a left to right direction using the component's position (Centre of Gravity) and the positions of punctuation, digit, and month text were noted. Next, the date patterns were searched using the sequence of labelled components. Two different date patterns were considered for searching, namely: numeric and semi-numeric patterns in the proposed approach.

1) Numeric Date Searching: For numeric date extraction (e.g. ১২-০৬-২০১২,৪/৬/১০, 15/08/2012, etc.), a sub-sequence of components was matched with the numeric date regular expression (d represents digit):

$$(d|dd)(/.-)(d|dd)(/.-)(dd|dddd)$$

In the proposed date searching algorithm, first the position of the two punctuation marks was determined. If one or two consecutive digits to the left of the left punctuation, one or two digits in the middle of these two punctuations and two or four digits on the right of a right punctuation were found, the sequence was labelled as a valid numeric

date field. Devanagari numerals (१-९) and Bangla numerals (**১-৯**) were used along with English numerals for Devnagari and Bangla script, respectively. This is because people use English numerals to write Bangla and Devanagari dates.

2) Semi-numeric Date Searching: For semi-numeric date fields (e.g. **২১ শে জুলাই**, ২০১০, २३ शे जुलाई, २०१२, 31st March, 2011) extraction, the following semi-numeric regular expressions were searched:

(md|mdd)(.,-)(dd|dddd) and
(d|dd)(contraction)(month)(.,-)(dd|dddd)

The entire patterns were searched in the sequence of line components for matching with any of the above date patterns. In a semi-numeric date pattern, textual month information may present in the front or in the middle of the sequence. 44 different types of textual months such as months in upper case (JANUARY), lower case (January) and short form (JAN, Jan) were considered here for English script. 24 types of month [12 English months written in Devnagari script (e.g. जनवरी, फरवरी, मार्च) and 12 Devanagari months (e.g. वैशाख, ज्येष्ठ, आषाढ़)] were considered as Devnagari month components. 24 types of Bangla month [12 English months written in Bangla script (e.g. **জানুয়ারী, মার্চ**) and 12 Bangla months (e.g. বৈশাখ, জ্যৈষ্ঠ)] were considered here. Four types of contractions (st, nd, rd, th) were used after day information in English dates. In Bangla script, 5 contractions (রা, ঠা, ই, লা, শে) and in Devnagari script four contractions (ला, रा, शे, इ) were used.

4.5 Segmentation-Free Approach on Date Extraction

An automatic date field extraction framework from handwritten documents has been proposed in this approach. In order to design the system, sliding window-wise Local Gradient Histogram (LGH)-based features and a character-level Hidden Markov Model (HMM)-based approach were applied for segmentation and recognition. Individual date components such as month-word (month wrote in word form i.e. January, Jan, etc.), numeral, punctuation and contraction categories were segmented and labelled from a text line. Next, a Histogram of Gradient (HoG) features and a Support Vector Machine (SVM) classifier were used to improve the results obtained from the HMM-based recognition system. Subsequently, both numeric and semi-numeric regular expressions of date patterns were considered for undertaking date pattern extraction in the labelled components.

4.5.1 System Overview

A two-step approach is presented for date field extraction from handwritten documents using two classifiers, namely HMMs and SVMs. This approach performs better than the previously proposed approach [110]. It is observed that the present system works better due to the performance of the HMM on handwritten text. Unlike the previous method, here automatic segmentation of words into characters avoids the problem of pre-segmentation of date fields. The proposed approach was inspired by the method proposed by Roy et al. [111] for word recognition. The HMM was used for character level segmentation and recognition of date components such as numerals, month-word, punctuation, etc. to locate the date field in handwritten documents. The HMM-based systems performed well for the automatic segmentation of characters in handwritten words, but due to generative properties of HMMs, character recognition may fail occasionally. An SVM-based discriminative classifier trained with numerals and punctuation was applied to improve the recognition results of labelled date fields obtained from the HMM-based system. A flow diagram of date field recognition system is shown in Fig. 4.13.

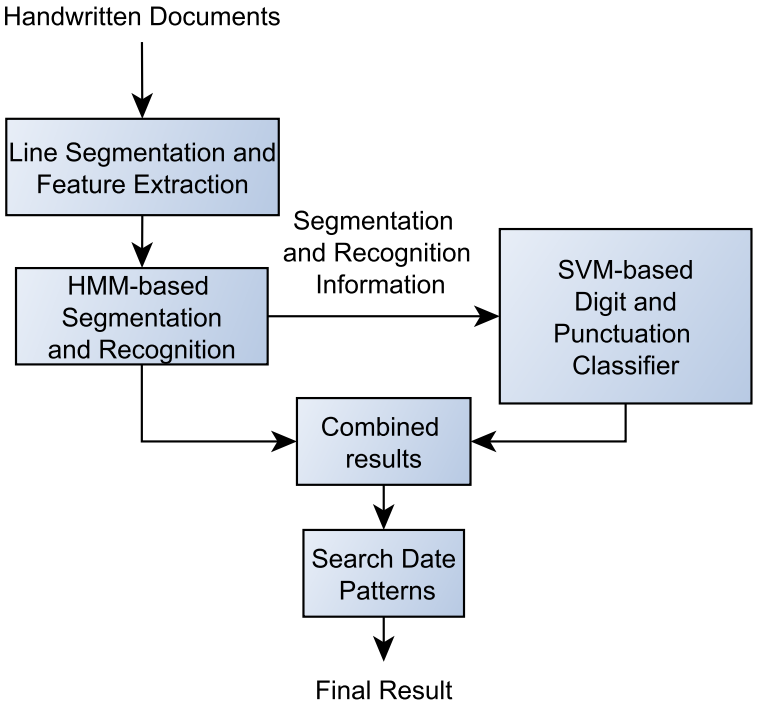


FIGURE 4.13: Flow diagram of the proposed date field recognition system.

4.5.2 Hidden Markov Model-Based Recognition System

As an initial step, Otsu's binarization method was applied to convert the grayscale images into binary images. The proposed date retrieval approach searches the date patterns in text line images. Hence, the binary document was segmented into individual text lines using a line segmentation algorithm [107] and the segmented lines were used for experimentation. Character-based HMMs [112] have been successfully used for recognition of arbitrary sets of words in English (Roman)/Latin scripts. An advantage of these systems is that they allow the recognition of unknown characters from the training data once the character models are trained. HMMs avoid the problem of pre-segmentation of words into characters so the errors of pre-segmentation can be eliminated. In the training phase, character HMMs were trained for each character of the alphabet based on transcribed text line images; since it requires date sub-fields (i.e. textual month, numerals and punctuations) samples as training material. All possible date patterns were considered in the experiment. A date pattern can consist of numerals, punctuation, and month-words (month written in word form i.e. textual month). At the recognition stage, the trained character HMMs was connected to a keyword text line model in order to calculate the likelihood score of the input text line. This likelihood score was finally normalized with respect to a general filler text line model and the length of the keyword feature vector sequence before it was compared to a threshold. Line images were represented as sequences of feature vectors $X = x1x2...xT$, also known as sequences of frames. The details of sliding window-based Local Gradient Histogram feature computation method is described in Section 4.3.3 and the details of HMM is given in Section 4.3.7. The sliding window-based feature vectors along with the line-wise transcriptions of the handwritten line images were used in order to train the character-level HMMs. The HTK toolkit [113] implementation model of the HMM developed for speech signal modelling was used in the experiments where recognition was based on the Viterbi algorithm.

Computation of character boundaries: A Viterbi forced alignment (FA) algorithm was applied to calculate the character boundaries in handwritten lines. The algorithm found the optimal alignment of a set of Hidden Markov Models. An iterative alignment and retraining process called embedded training was used to refine the character segmentation boundaries. The character segments $(S_1, S_2, \cdots, S_n)$ of a given hypothesis were obtained using the alignment algorithm. An N-best Viterbi list composed of N hypotheses was generated. From these, the best hypothesis was chosen based on the addition of a maximum likelihood (ML) segmentation at the character level. Qualitative performance of character level segmentation of this algorithm on sample handwritten lines is shown in Fig. 4.14.

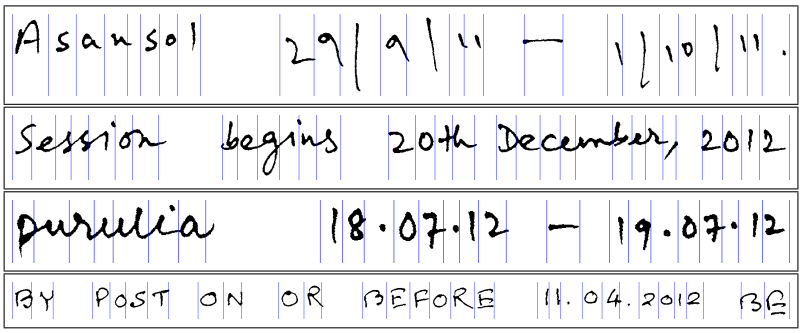


FIGURE 4.14: Qualitative performance of character segmentation results on sample handwritten lines are shown here.

4.5.3 SVM-Based Recognition System

An SVM-based recognition system was used as an isolated character recognizer. The alignment information produced by the Viterbi-forced alignment of characters was used for this stage of the system. Next, the SVM-based recognizer especially trained with numerals and punctuation was applied to refine the results obtained from the HMM-based recognition system. The semi-numeric strings were not considered by the SVM-based system and the results obtained from the HMM-based system were used in the final combination stage. The SVM-based numeral classifier initially takes the character alignment information produced by the HMM-based recognizer. The recognition errors generated by the HMM-based system due to segmentation problems were handled by the SVM-based isolated numeral and punctuation classifier. Detailed descriptions of Histogram of Oriented Gradients (HoG) feature extraction technique used with the SVM classifier is described in Section 4.3.4.

SVMs: The SVM-based [88] classifier is presented in Section 4.3.6. In these experiments, an SVM-based numeral and punctuation classifier was used. The reported recognition results in these experiments were based on the Gaussian kernel only. The hyperparameters of the SVM were set using a validation process as follows : kernel type = RBF, $\gamma = 0.04$ and $C = 3$. The best result was achieved by setting the above values for these parameters.

4.5.4 Combination of Recognition Scores

In this procedure, the inputs were the N-best list score from the individual classifiers and the score of both classifiers were usually normalized before fusion. The resultant score was calculated by a combination of scores obtained from the classifiers as follows. Let $D_1, D_2, \cdots, D_L$ be the set of L classifiers [114]. The output of the i^{th} classifier is denoted as $D_i(x) = [d_{i,1}(x), \cdots, d_{i,c}(x)]^T$, where $d_{i,j}(x)$ is the degree of support given

by classifier D_i to the hypothesis that test where class x comes from and c refers to the crisp class label. We construct $\widehat{D}$, the fused output of the L level classifier as

$$\widehat{D} = F(D_1(x), \cdots, D_L(x))$$

where, F is the aggregation rule of the maximum average and product operator. The top three likelihoods along with the confidence scores were estimated for particular segmented components from the HMM-based system. The SVM-based system estimated the confidence measure of the top three labels obtained from the HMM's output. Finally, the label of a segmented part which received a combined maximum score was considered as the final class label. Fig. 4.15 shows an example of how the combination was performed using the top three scores obtained from the classifiers.

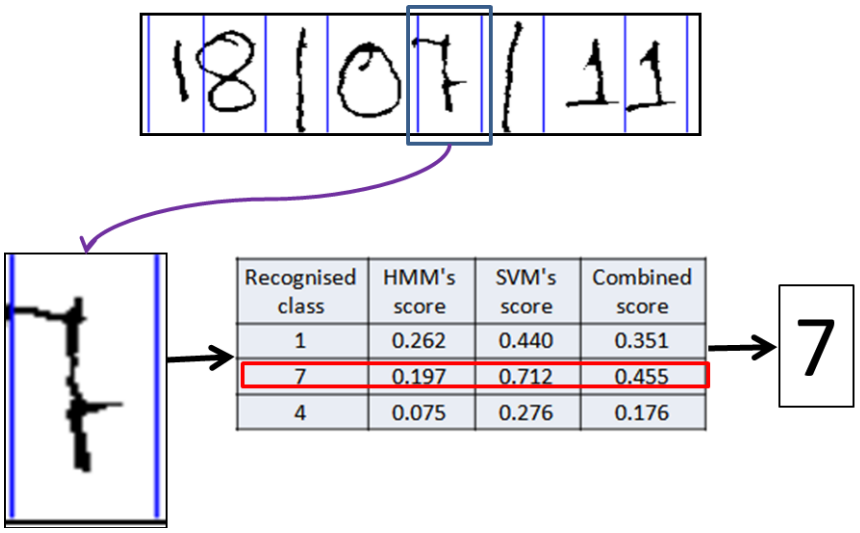


FIGURE 4.15: An example of score combination obtained from the HMM and the SVM classifiers. An average value has been considered as the combined score.

4.5.5 Matching of Date Pattern

The labelled components in each candidate text line such as punctuation, numeral and months-word (ex. Jan, Feb) were noted. Next, the date regular expressions were searched for using the sequence of labelled components. Two different date patterns for searching, namely: numeric and semi-numeric patterns were considered in the proposed approach.

Numeric and Semi-Numeric Date Matching : A date field consisting of only numerals and punctuations is considered as a numeric date field in this approach, e.g. (15/08/2012, 12-01-12 etc.). Other date fields that consist of month-word, numerals and contractions (st, nd, rd, th) are considered as semi-numeric dates e.g. 31st March 2011. For numeric date extraction a sub-sequence of components with the following date regular expression were searched:

(d|dd)(/.-)(d|dd)(/.-)(dd|dddd)

A complete numeric date field consists of at least one or at most two numerals for day information, at most two numerals of month information and a maximum of four numerals for year information. The following date regular expressions were searched for semi-numeric dates.

(md|mdd)(.,-)(dd|dddd) and
(d|dd)(contraction)(m)(.,-)(dd|dddd)

where d represents numerals, m represents a month-word and three types of punctuation in the date syntax were considered. Two types of regular expression for semi-numeric date fields were considered. In a semi-numeric date pattern, month-word may present in the front or in the middle of the sequence. Contractions can be found before alpha-numeric month information. The 'grep' command-line utility, which is available in Unix-like systems for pattern searching, was used to search all the above-mentioned date patterns from the transcribed lines.

4.6 Results and Discussion

This section evaluates the performance of different levels of the proposed approaches by considering various parameters. The different datasets used in the different levels of experiments are described in this section. Qualitative and quantitative results are detailed which show the efficiency of the proposed approaches. It is to be noted that all the experiments were performed using a PC with Intel core 2.5 GHz processor and 4 GB of RAM.

4.6.1 Performance of Segmentation-Based Approach

The dataset used to evaluate the performance of segmentation-based approach for the multi-script date extraction and the detailed experimental results of different modules of the proposed system are provided in the following sections.

4.6.1.1 Datasets

To the best of the researcher's knowledge, there exists no standard date dataset consisting of handwritten date samples to evaluate date extraction methods. Hence, different datasets were used for the training of classifiers at different levels. It should be noted that the training and test datasets were different in the experiments. Table 4.1 and Table 4.2 show the details of training and test dataset used respectively in the experiments.

Training Datasets: Row 2 of Table 4.1 shows the number of Bangla, Devnagari, and English handwritten words used to train the SVM classifier at the script identification level. This dataset was collected earlier for handwriting recognition experiments. Primitive segments and reservoir blobs were extracted from these handwritten words and used to train the classifier for identification of scripts. Row 3 of Table 4.1 shows the month-word dataset used to train the month identification system. It total, 44 classes of English months written in alphabetic letters (e.g. JANUARY, January, JAN and Jan etc.) and 24 classes each for Devnagari and Bangla scripts were considered. The month-word dataset was collected from 80 individuals, and contains different formats of the month (e.g. JANUARY, January, JAN and Jan etc. as mentioned in Section 4.4.2.5). Rows 4-6 of Table 4.1 show the volume of data used to train the classifier at the component level. English digits from the MNIST [7] dataset of handwritten digits were used for training the classifier at the component level for all the scripts. As mentioned earlier, English digits are often used in Bangla and Devnagari dates; hence, the MNIST dataset was used for all the training models along with their individual script digits.

For example, in Bangla and Devnagari script, Bangla and Devnagari numerals were used respectively for training along with English MNIST numerals. The dataset of Bangla and Devnagari numeral were collected earlier for the postal automation [115] work and used in this experiment for the training of the numeral recognition system. Here, at least 100 samples per numeral class were used from English, Devnagari, and Bangla numeral datasets.

Test Dataset: Handwritten text lines of a single script, as well as multi-script data were collected from 60 individuals of different professions and used for testing the system. Table 4.2 shows the total number of handwritten lines used from these three scripts for testing purposes.

Table 4.1: Dataset used to train the classification module of the system

Types of Data	**Bangla**	**Devnagari**	**English**
Word	2035	2079	1935
Month-word	2520	1900	2570
Numeral	3651	2137	MNIST dataset[7]
Punctuation	904	904	904
Contraction	604	578	730

TABLE 4.2: Dataset used to test the system performance

Types of Data	**Bangla**	**Devnagari**	**English**
Handwritten Line	1100	820	1240

4.6.1.2 Script Identification

Script identification experiments were performed to select the features. The computed script identification results using two features on the experimental dataset are shown in Table 4.3. It shows the SVM-based 5-fold cross-validation test accuracy for gradient and Gabor filter-based features. The Gabor filter-based features computed in this experiment are the same as those explained in [80]. It also shows the performance of both the features in the experiment. Table 4.3 shows how primitive segments (foreground) and reservoir blobs (background) information individually contribute to the script identification task. The gradient features were used finally for script identification as it was noted that gradient features outperformed Gabor in SVM-based 5-fold cross-validation experiments. Accuracy rates of 73.80% and 65.33% were obtained for gradient and Gabor, respectively when primitive segment information was used. Script identification using only reservoir blob information produced a 60.84% and 53.89% accuracy for gradient and Gabor features, respectively.

TABLE 4.3: SVM-based 5-fold cross validation results for script identification (accuracy is based on primitive segments and reservoir blobs)

Feature based on	**Primitive Segments(%)**	**Reservoir Blobs(%)**
Gradient	73.80	60.84
Gabor	65.33	53.89

The combined majority of primitive segments and reservoir blobs information was finally used for script identification in this system. The results obtained from word-wise script identification for English, Devnagari, and Bangla scripts are presented in Table 4.4. The accuracy obtained for English and Bangla scripts is encouraging. However, 7.15% and 3.12% of Devnagari scripts were confused with Bangla and English scripts, respectively. The similarity of alphabets, as well as the basic script structure between Bangla and Devnagari scripts, lead to a 7.15% confusion. If there was a tie in the total number of the recognition results during component classification, the script cannot be determined, and hence rejected. However, line-wise script identification was more accurate than word-wise script identification when the same identification technique was applied. The results of line-wise script identification for these three scripts are presented in Table 4.5.

The proposed script identification technique outperformed the previous work by Roy et al. [30]. Table 4.6 shows a comparative study with the previously proposed script identification technique. The test dataset used in the experiment for the comparison of performance is the same as the one used in [30].

TABLE 4.4: Word-wise script identification results (accuracy is given in percent)

Script	**Bangla**	**Devnagari**	**English**	**Rejection**
Bangla	**98.40**	0.60	0.00	1.00
Devnagari	7.15	**89.73**	3.12	0.00
English	0.40	0.00	**99.20**	0.40
Overall Accuracy	**94.69%**			

TABLE 4.5: Line-wise script identification results

Experimental Dataset	**Bangla(%)**	**Devnagari(%)**	**English(%)**
Handwritten lines	98.62	95.20	99.44

TABLE 4.6: Comparison of word-wise script identification results (accuracy is given in percent)

Method	**Bangla**	**Devnagari**	**English**	**Overall Accuracy**
Roy and Majumdar [30]	96.77	97.68	93.61	96.02
Proposed method	98.53	92.04	99.13	96.57

4.6.1.3 Results of Line Selection

As discussed in Section 4.4.2.5, a filtering process was used to remove lines without date patterns. Depending on the value of the total count of date components, lines were eliminated for the next level of processing. Table 4.7 shows the percent of line eliminated for all the three scripts (i.e. Bangla, Devnagari, and English) by setting the counter of the date element at 6. It was observed from the dataset that at least 6 date elements (i.e. 4 numerals and 2 punctuations) are required to represent a numeric date

pattern. Fig. 4.16 shows the overall result of line selection according to the number of date components (elements).

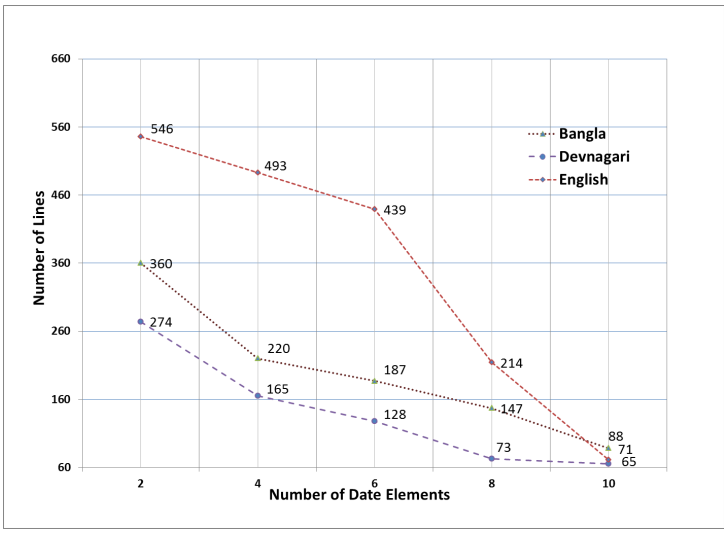


FIGURE 4.16: Results of line filter process on Bangla, Devnagari and English scripts. Lines were checked for 2 to 10 components.

It shows that the number of lines decreases in each script as the number of date elements increases. It can be seen that 71, 88 and 65 lines of English, Bangla and Devnagari scripts, respectively, contain 10 or more date elements. It can be also seen that 439, 187 and 128 lines of English, Bangla and Devnagari scripts, respectively, contain at least 6 date elements. Lines containing 6 or more date elements were considered for date pattern searching.

TABLE 4.7: Results of line selection

Script	**Lines eliminated(%)**
Bangla	83.00
Devanagari	84.39
English	64.90

4.6.1.4 Multi-Script Date Field Extraction

The measures of precision (P) and recall (R) were computed to evaluate the quantitative performance of the date extraction system for these three scripts. Depending on the ground truth of the date, extracted sequences were considered to be a valid date sequence or not. The precision-recall measure for the components of the date fields was also computed.

In this work, a DTW-based textual month-word identification approach was used. Precision-recall measures were computed for handwritten Bangla, Devnagari, English

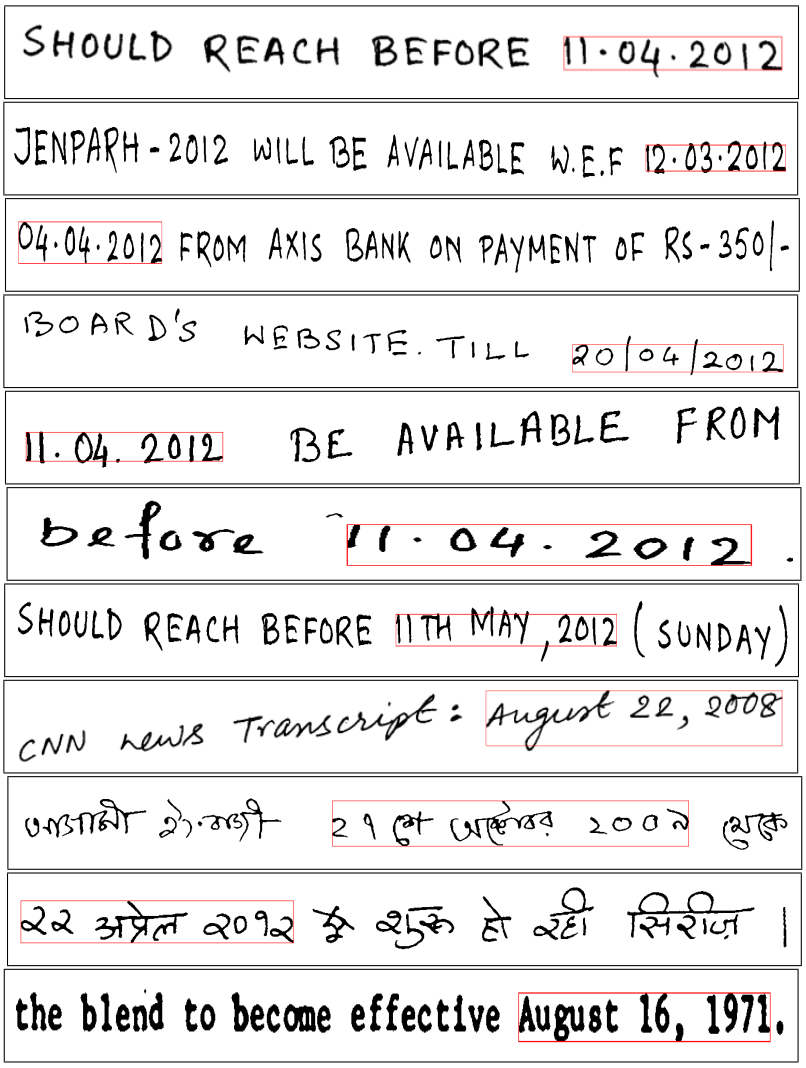


FIGURE 4.17: Qualitative results of numeric and semi-numeric dates (a-h) English handwritten, (i) Bangla handwritten, (j) Devnagari handwritten, (k) English printed. Extracted date fields are marked with a red box.

(Roman), and printed English scripts. Fig. 4.18 shows precision-recall curves of month-word identification for all the above scripts. A comparative study was also performed on the printed English dataset to show that the DTW-based month-word identification approach outperforms the SVM-based method proposed in [110] and the results are presented in Table 4.8. The measure of precision and recall of both DTW and SVM-based approaches were presented to show that the DTW-based approach outperforms the SVM-based method. The profile features and the gradient features were used for the DTW and SVM-based month-word identification approach, respectively.

The measure of precision and recall of date field component (month, numerals and punctuations) identification was also computed separately for each script. Rows 2-5 of Table 4.9 show the performance of the date retrieval technique on Bangla, Devnagari, English scripts and the overall result, respectively. To get an impression of the qualitative

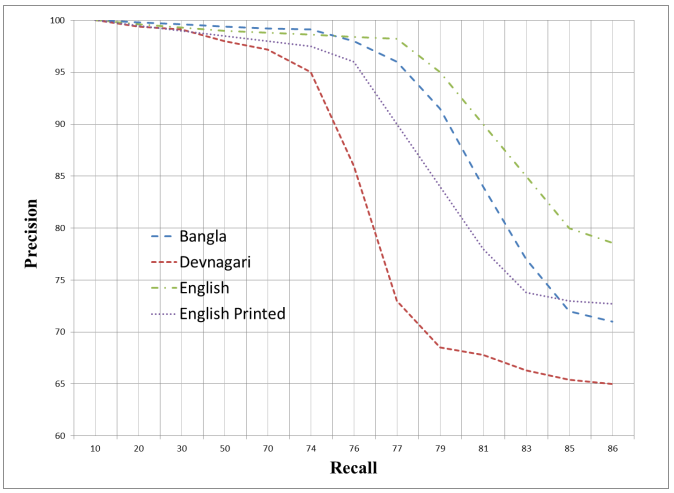


FIGURE 4.18: Precision vs recall curves for month-word extraction. Bangla, Devanagari, English and English printed PR curves are shown.

TABLE 4.8: Comparison of results for month-word identification (Q1: Fields retrieved and relevant, Q2: Relevant fields in dataset, Q3: Fields retrieved).

Method based on	**Q1**	**Q2**	**Q3**	**Precision (Q1/Q3*100)**	**Recall (Q1/Q2*100)**
DTW	96	114	129	74.42	84.21
SVM	89	114	126	70.63	78.07

TABLE 4.9: Component wise precision-recall measure (P: Precision , R: Recall)

Script	**Month**		**Numerals**		**Punctuations**	
	P	R	P	R	P	R
Bangla	70.97	86.27	88.02	93.26	92.41	95.03
Devnagari	68.75	78.57	85.07	92.83	90.15	96.21
English	80.21	85.56	83.97	95.64	93.35	95.37

results, a few examples of date field extraction of these scripts are shown in Fig. 4.17 and a precision-recall curve is also presented for all the scripts in Fig. 4.19. The same date field extraction technique was tested on the English printed documents with five hundred printed lines containing different formats (numeric and semi-numeric) of date samples. These were extracted from the documents of the 'Tobacco-800' dataset. A higher accuracy than that achieved on handwritten documents was attained in the experiment; 85.09% recall and 100% precision were achieved from this experiment.

TABLE 4.10: Precision recall measure of date field extraction (FR: Field for recogntion,Q1: Fields retrieved and relevant, Q2: Relevant fields in dataset, Q3: Fields retrieved)

FR	Q1	Q2	Q3	Precision (Q1/Q3*100)	Recall (Q1/Q2*100)
Bangla	106	134	106	100	79.10
Devnagari	82	110	82	100	74.55
English	329	419	329	100	78.52
Overall	512	663	512	100	77.22

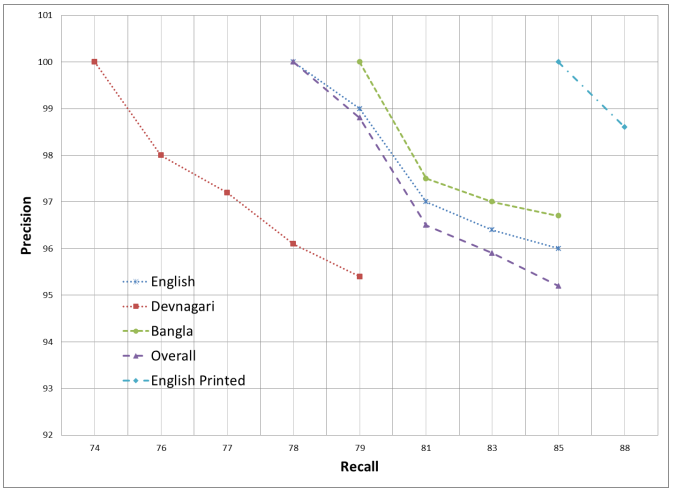


FIGURE 4.19: Precision vs Recall curves of date field extraction on Bangla, Devnagari, English and English printed scripts.

4.6.1.5 Error Analysis

It should be noted that the obtained errors were due to improper classification of textual months and non-months, and some errors are found due to incorrect touching digit segmentation and misclassification of digits. Fig. 4.20 shows some erroneous results.

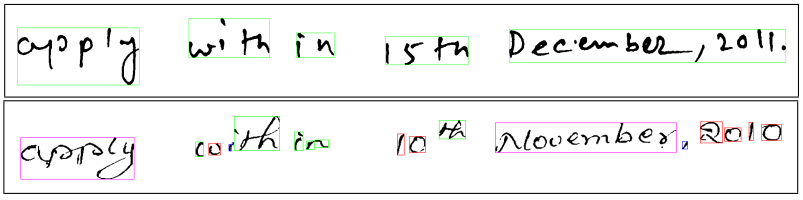


FIGURE 4.20: Erroneous results (a) the month-word 'December' is not correctly identified due to improper word segmentation (b) the word 'apply' is wrongly detected as a month field.

4.6.2 Performance of Segmentation-Free Approach

The details of the used datasets and the evaluation methods are presented in the following sections.

4.6.2.1 Datesets

Different sets of data were used to evaluate the system are presented in Table 4.11. The HMM-based system used handwritten text lines and handwritten numeral strings as the training data. The IAM English sentence dataset [8] and Indian PIN code strings which were written in English were used to train the HMM-based recognizer, whereas to train the SVM-based recognizer, English digits from the MNIST[1] dataset and handwritten punctuations were used. The test dataset used for the experiments included 1240 handwritten text lines collected from English documents which were written by different individuals from various professions. The dataset contains various date sequence of valid patterns including numeric and semi-numeric dates.

TABLE 4.11: Experimental datasets used for segmentation-free approach

Types of Data	**Volume of Data**
Handwritten line	IAM dataset [8]
Indian PIN code	1500
English digits	MNIST dataset [7]
Punctuation	904
Handwritten lines for test	1240

4.6.2.2 Performance Based on HMM

A 5-fold cross validation technique was applied to compute the recognition accuracy. A validation dataset was used to vary the HMM parameters such as the number of states, the number of Gaussian distributions and width of the sliding window. On the basis of analysing the performance, 6 states for each character model and 8 GMMs for each state were selected. In LGH feature extraction, a sliding window width of 10 pixels with a 50% overlapping ratio provided the best result in the experiment. Table 4.12 shows the performance of month-word recognition accuracy by HMMs only. Table 4.13 shows the results produced in these experiments solely by the HMM-based system. A few sample images in Table 4.15 show the qualitative results obtained from the experiments.

[1] http://yann.lecun.com/exdb/mnist/

TABLE 4.12: Performance of HMM-based month-word recognition

Total samples	**Correctly recognised samples**	**Precision (%)**	**Recall (%)**
184	160	100	86.95

TABLE 4.13: HMM-based recognition accuracy of date fields

Data	**Total samples**	**Correctly recognised samples**	**Accuracy (%)**
Numeric Date	256	208	81.25
Semi-Numeric Date	163	129	79.14

4.6.2.3 Combined Results of HMM and SVM

The results presented in Table 4.14 show the improved results obtained by the HMM-SVM hybrid rescoring system. It also reveals that HMM-SVM hybrid system performed better on numeric date extraction because of the SVM-based recognizer performed well in numerals recognition.

TABLE 4.14: Recognition accuracy based on combined results of HMM and SVM

Data	**Total samples**	**Correctly recognised samples**	**HMM+SVM Accuracy (%)**
Numeric Date	256	225	87.89
Semi-Numeric Date	163	134	82.20

4.6.2.4 Comparative Analysis

The proposed approach performed better than the approach presented in [110] on English script. Table 4.16 and 4.17 show the comparison on month-word and date field recognition using the precision-recall measure. It was observed that the Viterbi forced alignment (FA) algorithm played a vital role calculate the optimal character boundary which leads to proper character segmentation.

4.6.2.5 Error Analysis

Some errors occurred due to the segmentation problems generated by the HMM on some low-quality images. The SVM classifier also failed to recognise those characters if there were improper character alignment. Table 4.18 shows some samples of erroneous results obtained from the experiments.

TABLE 4.15: Qualitative results from the HMM-based and combined approach on sample images

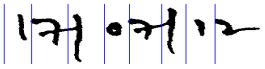
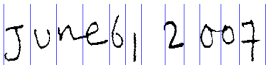
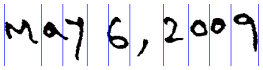
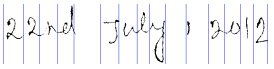
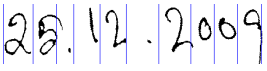
Date samples	**HMM results**	**HMM+SVM results**
	17/07/1r	17/07/12
	June6„2 007	June6, 2007
	May 6,20o9	May 6,2009
	22nd July, a0i2	22nd July, 2012
	t5.12. 2009	25.12. 2009

TABLE 4.16: Comparison with the previous method on month-word recognition.

Method	**Precision (%)**	**Recall (%)**
Mandal et al. [110]	77.41	83.72
HMM-based approach	100	86.95

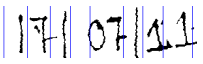
TABLE 4.17: Comparison with the previous method on complete date field recognition

Method	**Precision (%)**	**Recall (%)**
Mandal et al. [110]	89.08	74.87
HMM-based approach	100	85.68

4.7 Summary

Two different approaches for date field extraction from documents for date-based document retrieval have been proposed in this chapter. First, an automated system for date field extraction from multi-script documents has been presented. The script of a document was determined using a novel script identification technique. Next, a DTW-based method was applied in the present study to extract month-word component from scripts. The gradient-based features and SVM classifier were used for the identification of other date components such as a digit, punctuation and contraction (e.g. st, nd, rd,

TABLE 4.18: Erroneous results

Date samples	**Ground truth**	**HMM Results**	**HMM+SVM results**
	17/07/11	/07/11	17/4/18
	17/05/2007	17/05/207	17/05/209

th). Finally, different date patterns are searched from the sub-sequence of labelled components. Although this technique works for multi-lingual documents, there was some scope for improvement. In order to address the pre-segmentation problem of the date field in the second approach, the HMM-based segmentation-free approach was applied. The segmentation errors at the word and character level were decreased significantly by the HMM-based approach. To produce the final document retrieval result a simple date string matching technique needs to be applied. The accuracy achieved from the proposed date recognition system would be the retrieval accuracy eventually, because text string matching between query date and extracted document date is an error free stage. An extension on regular expressions for searching would be an interesting direction for future work. As mentioned earlier, this would foster further research to facilitate generic solutions in document analysis, with different applications in automation.

Chapter 5

Conclusions and Future Research

This chapter concludes the thesis by summarising the research contributions, limitations and suggests some areas for future work. The objective of this research was to develop a Document Image Retrieval (DIR) system using two significant pieces of key information namely signature and date. Therefore, to achieve this, in this study, the significance of document image retrieval has been investigated and the importance of signature and date used as key information were also analyzed. It appears from the investigation that although there are many published research works on DIR, only a few studies in the literature that deal with signature or date as key information for document retrieval. Though there are many potential application areas, the gap appearing in the extant literature indicates the need for more research in this field. Two independent systems were proposed in Chapter 3 and Chapter 4 for document retrieval based on signature and date information, respectively.

The proposed signature-based document retrieval technique has three main stages: signature detection, signature segmentation, and matching of signatures. The detection and segmentation of signatures from documents involve challenges due to the unconstrained nature of handwriting and the variation of the structure in signatures. The matching of the signatures are also challenging due to non-rigid shape and intra-class variance among signatures. To address these challenges, two different approaches were employed consisting of two different features extraction techniques namely gradient and SIFT-descriptor with Bag-of-Visual-Words (BoVW)-based features for the signature detection stage. The SVM was used as a classifier for both the techniques of signature detection. Next, the Density-Based Spatial Clustering approach was employed for the task of signature segmentation. Finally, SIFT-descriptor with BoVW-based features was used for signature shape coding to retrieve documents.

The extraction of handwritten dates in a document is the pivotal task in the date-based document retrieval system. However, the challenges involved in date-based document image retrieval are entirely different. A date can appear in documents with many patterns (i.e. numeric and semi-numeric) and the segmentation and recognition of elements of all patterns are quite challenging. Two different multi-stage approaches for date extraction have been presented in Chapter 4 of this thesis. The first method of date extraction is a multi-stage segmentation-based approach. First, the script was identified as Bangla or Devanagari or English. In the second stage, the date elements (i.e. month-word, numerals, and punctuation) were recognised. Finally, date patterns were searched from the identified components. The second method of date extraction is a segmentation-free approach based on HMM. In the first stage, an HMM-based text recognizer was employed. The HMM-SVM rescoring approach was employed at the second stage to refine the results obtained from the first stage. Finally, numeric and semi-numeric date patterns were searched from the recognised components. The dataset contained various types of data used for investigation at different stages of the systems.

In summary, this thesis has made the following contributions to the field of Document Image Retrieval using the signature and date information:

- Signature segmentation from documents using contextual information, Mandal et al. [80].
- Multi-script Off-line signature identification, Mandal et al. [116].
- Signature-based multi-script document retrieval using Bag-of-Visual-Words (Under revision).
- Multi-lingual date field extraction, Mandal et al. [99].
- Date field extraction using HMMs, Mandal et al. [100].

5.1 Future Research

The document retrieval techniques proposed in this thesis is focused on extraction and recognition of both, signature and date. In the near future, the work can be extended to use signature and date information jointly or individually to retrieve documents from a document repository. A flow diagram of the future system is given in Fig. 5.1. Signatures are first segmented and then the recognition process recognizes the segmented signature to determine to whom a particular signature belongs to. On the other side, if there is any date information in the document, it will also be extracted and recognised. Finally, signature and date information can be used jointly or individually to retrieve documents from a document repository.

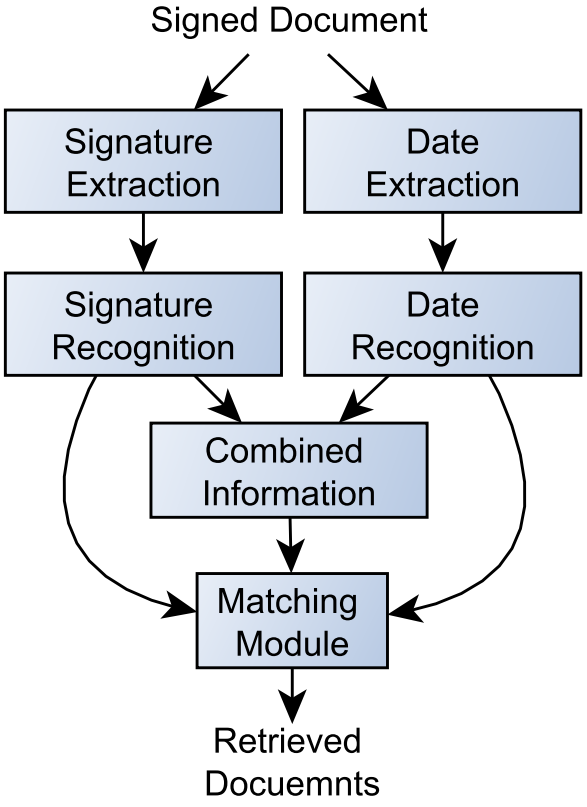


FIGURE 5.1: A block diagram of the future document retrieval system using signature and/or date information as a key.

A significant number of experiments have been conducted towards signature and date-based document retrieval. Numerous important experimental results and useful observations have been made. There are major scope for improving this research. Firstly, future research may investigate the area further in order to improve the performance of multi-script document retrieval systems. More sophisticated features based on Convolutional Neural Networks (CNN) [117] can be extracted for experiments, and to achieve more accurate results during classification if large datasets are available. Secondly, the proposed system is mainly a segmentation-based approach. A segmentation-free approach can be proposed to eliminate the error in the segmentation stages. Finally, the system can be extended towards other languages or scripts. Here, only three scripts (English, Hindi, and Bangla) are considered for the experiments. Many important other key pieces of information such as seals, logos, and words, etc. can be considered for development with document image retrieval systems.

In addition, the following important points can be considered for future work.

- The errors obtained from the present methods can be addressed.

- A segmentation-free deep learning-based approach can be proposed for signature-based document retrieval.
- Many other important key pieces of information (i.e. seal, logo, word, etc.) which are often present in administrative documents can be included to develop a generic retrieval system
- This same method can be extended for document categorization as well.
- The HMM-based date retrieval technique can be extended to other scripts.

Bibliography

[1] http://legacy.library.ucsf.edu/. The Legacy Tobacco Document Library (LTDL). University of California, San Francisco, 2007.

[2] A. L. Spitz. Determination of script, language content of document images. *IEEE Transactions on Pattern Recognition*, 19(3):235–245, 1997.

[3] U. Pal and B.B. Chaudhuri. Identification of different script lines from multi-script documents. *Image and Vision Computing (IVC)*, 20(13/14):945–954, 2002.

[4] M. Morita, L. S. Oliveira, R. Sabourin, F. Bortolozzi, and C. Y. Suen. An HMM-MLP hybrid system to recognize handwritten dates. In *Proc. International Joint Conference on Neural Networks*, 2002.

[5] Xianzhi Du, Wael Abd Almageed, and David S. Doermann. Large-scale Signature Matching using Multi-Stage Hashing. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 976–980, 2013.

[6] C. M. Travieso1 J. B. Alonso1 J. F. Vargas, M. A. Ferrer. Off-line handwritten signature GPDS-960 corpus. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 764–768, 2007.

[7] http://yann.lecun.com/exdb/mnist/. MNIST dataset.

[8] U. Marti and H. Bunke. The IAM-database: An English sentence database for off-line handwriting recognition. *International Journal on Document Analysis and Recognition (IJDAR)*, 5:39–46, 2002.

[9] J. Lladós, D. Karatzas, J. Mas, and G. Sanchez. A generic architecture for the conversion of document collections into semantically annotated digital archives. *Journal of Universal Computer Science*, 14:2912–2935, 2008.

[10] C. Chatelain, L. Heutte, and T. Paquet. A syntax-directed method for numerical field extraction using classifier combination. In *Proc. International Workshop on Frontiers in Handwriting Recognition (IWFHR)*, pages 1–10, 2004.

[11] G. Zhu and D. Doermann. Logo matching for document image retrieval. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 606–610, 2009.

[12] G. Zhu, Y. Zheng, D. Doermann, and S. Jaeger. Signature detection and matching for document image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31(11):2015–2031, 2009.

[13] S. Djeziri, F. Nouboud, and R. Plamondon. Extraction of signatures from check background based on a filiformity criterion. *IEEE Transactions on Image Processing*, 7(10):1425–1438, 1998.

[14] S. Ahmed, M. I. Malik, M. Liwicki, and A. Dengel. Signature segmentation from document images. In *Proc. International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 425–429, 2012.

[15] G. Zhu, Y. Zheng, D. Doermann, and S. Jaeger. Multi-scale structural saliency for signature detection. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

[16] S. Levy. "google′s two revolutions". Newsweek, http://www.msnbc.msn.com/id/6733225/site/newsweek/, Dec./Jan., 2004.

[17] L. Lam, C. Y. Suen, and Q. Xu. Automatic recognition of handwritten data on cheques - fact or fiction? *Pattern Recognition Letters*, 20(11-13):1287– 1295, 1999.

[18] http://code.google.com/p/ocropus/. Ocropus.

[19] T. M. Rath and R. Manmatha. Word image matching using Dynamic Time Warping. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 521–527, 2003.

[20] A. Fischer, A. Keller, V. Frinken, and H. Bunke. HMM-based word spotting in handwritten documents using sub-word models. In *Proc. International Conference on Pattern Recognition (ICPR)*, pages 3416–3419, 2010.

[21] V. Frinken, A. Fischer, R. Manmatha, and H. Bunke. A novel word spotting method based on Recurrent Neural Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 3(3):211–224, 2012.

[22] C. Zhang, J. Yang, and Z. Lou. Preprocessing of handwritten date images on chinese cheque. In *Proc. International Conference on Pattern Recognition (ICPR)*, pages 1010–1013, 2006.

[23] M. A . Ferrer, J . B. Alonso, and C. M. Travieso. Off-line geometric parameters for automatic signature verification using fixed-point arithmetic. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 27(6):993–997, 2005.

[24] A. Kholmatov and B. Yanikoglu. Identity authentication using improved online signature verification method. *Pattern Recognition Letters*, 26(15):2400–2408, 2005.

[25] F. A. Fernandez, J. Fierrez, M. M. Diaz, and J. O. Garcia. Fusion of static image and dynamic information for signature verification. In *Proc. International Conference on Image Processing (ICIP)*, pages 2725–2728, 2009.

[26] H. R. Lv, W. J. Yin, and J. Dong. Off-line signature verification based on deformable grid partition and Hidden Markov Models. In *Proc. International Conference on Multimedia and Expo (ICME)*, pages 374–377, 2009.

[27] D. Ghosh, Tulika Dube, and A. P. Shivaprasad. Script recognition : A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32:2142–2161, 2010.

[28] S. Abirami and D. Manjula. A survey of script identification techniques for multi-script document images. *International Journal of Recent Trends in Engineering (IJRTE)*, 1:246–249, 2009.

[29] P. B. Patil and A. G. Ramakrishnan. Word level multi-script identification. *Pattern Recognition Letters*, 29:1218–1229, 2008.

[30] K. Roy and K. Majumdar. Trilingual script separation of handwritten postal document. In *Proc. Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP)*, pages 693–700, 2008.

[31] A. M. Namboodiri and A. K. Jain. Online script recognition. In *Proc. International Conference on Pattern Recognition (ICPR)*, pages 736–739, 2002.

[32] J. Schenk, J. Lenz, and G. Rigoll. Novel script line identification method for script normalization and feature extraction in on-line handwritten white board note recognition. *Pattern Recognition*, 42:3383–3393, 2009.

[33] J. Hochberg, K. Bowers, M. Cannon, and P. Kelly. Script and language identification for handwritten document images. *International Journal on Document Analysis and Recognition (IJDAR)*, 2:45–52, 1999.

[34] K. Roy, A. Alaei, and U. Pal. Word-wise handwritten Persian and Roman script identification. In *Proc. International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 628–633, 2010.

[35] K. Roy, S. K. Das, and M. Obaidullah. Script identification from handwritten document. In *Proc. National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (ICVGIP)*, pages 66–69, 2011.

[36] L. Zhou, Y. Lu, and C. L. Tan. Bangla/English script identification based on analysis of connected component profiles. In *Proc. International Workshop on Document Analysis Systems (DAS)*, pages 243–254, 2006.

[37] V. Singhal, N. Navin, and D. Ghosh. Script-based classification of hand-written text documents in a multi-lingual environment. In *Proc. International Workshop Research Issues in Data Engineering: Multilingual Information Management (RIDE-MILM)*, pages 47–54, 2003.

[38] R. Sarkar, N. Das, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu. Word level script identification from Bangla and Devanagri handwritten texts mixed with Roman script. *Journal of Computing*, 2(2):103–108, 2010.

[39] G. G. Rajput and Anith H B. Handwritten script recognition using DCT and wavelet features at block level. In *Proc. Recent Trends in Image Processing and Pattern Recognition (RTIP2R)*, pages 158–163, 2010.

[40] K. Kuhnke, L. Simoncini, and Zs. M. Kovacs-V. A system for machine-written and hand-written character distinction. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 811–814, 1995.

[41] U. Pal and B.B. Chaudhuri. Automatic separation of machine-printed and hand-written text lines. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 645–648, 1999.

[42] J.K. Guo and M.Y. Ma. Separating handwritten material from machine printed text using Hidden Markov Models. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 439–443, 2001.

[43] Y. Zheng, H. Li, and D. Doermann. The segmentation and identification of hand-writing in noisy document images. In *Proc. International Workshop on Document Analysis System (DAS)*, pages 95–105, 2002.

[44] V. K. Madasu, M. H. M. Yusof, M. Hanmandlu, and K. Kubik. Automatic extraction of signatures from bank cheques and other documents. In *Proc. Digital Image Computing: Techniques and Applications (DICTA)*, pages 591–600, 2003.

[45] S. I. Jang, S. H. Jeong, and Y.S.Nam. Classification of machine-printed and handwritten addresses on Korean mail piece images using geometric features. In

Proc. International Conference On Pattern Recogniton (ICPR), pages 2:383–386, 2004.

[46] F. Farooq, K. Sridharan, and V. Govindaraju. Identifying handwritten text in mixed documents. In *Proc. International Conference on Pattern Recogniton (ICPR)*, pages 1–4, 2006.

[47] X. Peng, S. Setlur, V. Govindaraju, R. Sitaram, and K. Bhuvanagir. Markov random field based text identification from annotated machine printed documents. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 431–435, 2009.

[48] X. Peng, S. Setlur, V. Govindaraju, and R. Sitaram. Overlapped text segmentation using Markov Random Field and aggregation. In *Proc. International Workshop on Document Analysis System (DAS)*, pages 129–134, 2010.

[49] H. Srinivasan and S. N. Srihari. Signature-based retrieval of scanned documents using Conditional Random Fields. *Computational Methods for Counterterrorism*, pages 17–32, 2009.

[50] X. Peng, V. Govindaraju, S. Setlur, and R. Sitaram. Text separation from mixed documents using a tree-structured classifier. In *Proc.International Conference On Pattern Recogniton (ICPR)*, pages 241–244, 2010.

[51] A. Mazzei, F. Kalpan, and P. Dillenbourg. Extraction and classification of handwriting annotataions. In *ACM 978-1-60558-843-8/10/09*, pages 1–4, 2010.

[52] J. Kumar, J. Pillai, and D. Doermann. Document image classification and labeling using multiple instance learning. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 1059–1063, 2011.

[53] H. Dewan, W. Xichang, and L. Jiang. A content-based retrieval algorithm for document image database. In *Proc. International Conference On Multimedia Technology (ICMT)*, pages 1–5, 2010.

[54] H. Wang. Document logo detection and recognition using Bayesian model. In *Proc. International Conference On Pattern Recogniton (ICPR)*, pages 1961–1964, 2010.

[55] A. Alaei and M. Delalandre. A complete logo detection/recognition system for document images. In *Proc. International Workshop on Document Analysis Systems (DAS)*, pages 324–328, 2014.

[56] E. Özgündüz, T. Şentürk, and M. E. Karsligil. Off-line signature verification and recognition by support vector machine. In *Proc. European Signal Processing (EUSIPCO)*, pages 113–116, 2005.

[57] A. Chalechale and A. Mertins. Line segment distribution of sketches for Persian signature recognition. In *Proc. TENCON*, volume 1, pages 11–15, 2003.

[58] P.P. Roy, S. Bhowmick, U. Pal, and J. Y. Ramel. Signature based document retrieval using GHT of background information. In *Proc. International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 225–230, 2012.

[59] C. Oz. Signature recognition and verification with Artificial Neural Network using moment invariant method. In *Proc. International Symposium in Neural Networks (ISNN)*, pages 195–202, 2005.

[60] S.M. Odeh and M.K. Khalil. Off-line signature verification and recognition: Neural Network approach. In *Proc. Innovations in Intelligent Systems and Applications (INISTA)*, pages 34–38, 2011.

[61] A. Chalechale, G. Naghdy, and A. Mertins. Signautre-based document retrieval. In *International Symposium on Signal Processing and Information Technology (ISSPIT)*, pages 597–600, 2003.

[62] P. Kudlacik and P. Porwik. A new approach to signature recognition using the fuzzy method. *Pattern Analysis and Applications (PAA)*, 17(3):451–463, 2014.

[63] J.A. Rodríguez-Serrano and F. Perronnin. Handwritten word-spotting using Hidden Markov Models and universal vocabularies. *Pattern Recognition*, 42(9):2106–2116, 2009.

[64] C. L. Tan, W. Huang, S. Y. Sung, Z. Yu, and Y. Xu. Text retrieval from document images based on word shape analysis. *Applied Intelligence*, 18(3):257–270, 2003.

[65] S. Lu, L. Li, and C. L. Tan. Document image retrieval through word shape coding. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 30(11):1913–1918, 2008.

[66] Rafael C. Gonzalez and Richard E. Woods. *Extraction of Textual Information from Images for Information Retrieval.* Ph.D. thesis, 2009.

[67] C. L. Tan, W. Huang, Z. Yu, and Y. Xu. Imaged document text retrieval without OCR. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 24(6):838–844, 2002.

[68] Y. Lu and C. L. Tan. Word spotting in Chinese document images without layout analysis. *Pattern Recognition*, 3:57–60, 2002.

[69] K. Terasawa and Y. Tanaka. Slit-style HOG feature for document image word spotting. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 116–120, 2009.

[70] T. Konidaris, B. Gatos, K. Ntzios, and I. Pratikakis. Keyword-guided word spotting in historical printed documents using synthetic data and user feedback. *International Journal on Document Analysis and Recognition (IJDAR)*, pages 167–177, 2007.

[71] A. Tarafdar, R. Mandal, S. Pal, U. Pal, and F. Kimura. Shape code based word-image matching for retrieval of Indian multi-lingual documents. In *Proc. International Conference On Pattern Recogniton (ICPR)*, pages 1989–1992, 2010.

[72] G. Koch, L. Heutte, and T. Paquet. Numerical field extraction in handwritten incoming mail documents. In *Proc. International Workshop on Pattern Recognition in Information Systems (PRIS)*, pages 167–172, 2003.

[73] C. Chatelain, L. Heutte, and T. Paquet. Segmentation-driven recognition applied to numerical field extraction from handwritten incoming mail documents. In *Proc. International Workshop on Document Analysis System (DAS)*, pages 564–575, 2006.

[74] S. Thomas, C. Chatelain, L. Heutte, and T. Paquet. Alpha-numerical sequences extraction in handwritten documents. In *Proc. International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 232–237, 2010.

[75] L. R. Rabiner. A tutorial on Hidden Markov Models and selected applications in speech recognition. *IEEE*, 77(2):257–286, 1989.

[76] L. Lam C. Y. Suen, Q. Xu. Automatic recognition of handwritten data on cheques - fact or fiction? *Pattern Recognition Letters*, 20:1287–1295, 1999.

[77] Q. Xu, L. Lam, and C. Y. Suen. A knowledge-based segmentation system for handwritten dates on bank cheques. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 384–388, 2001.

[78] Q. Xu, L. Lam, and C. Y. Suen. Automatic segmentation and recognition system for handwritten dates on Canadian bank cheques. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, 2003.

[79] J. Kumar, R. Prasad, H. Cao, W. Abd-Almageed, D. Doermann, and P. Natarajan. Shape codebook based handwritten and machine printed text zone extraction. *Proc. SPIE 7874, 787406 (2011); doi:10.1117/12.876725.*

[80] R. Mandal, P. P. Roy, and U. Pal. Signature segmentation from machine printed documents using contextual information. *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, 26(7):1253003(1–25), 2012.

[81] M. Blumenstein, Miguel A. Ferrer, and J.F. Vargas. The 4nsigcomp2010 off-line signature verification competition: Scenario 2. In *Proc. International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 721–726, 2010.

[82] M. Ester, H. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 226–231, 1996.

[83] M. Shi, Y. Fujisawa, T. Wakabayashi, and F. Kimura. Handwritten numeral recognition using gradient and curvature of gray scale image. *Pattern Recognition*, 35(10):2051–2059, 2002.

[84] Chinese text location under complex background using gabor filter and svm. *Neurocomputing*, 74(17):2998 – 3008, 2011.

[85] A. Khotanzad and Y.H. Hong. Invariant image recognition by Zernike moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 12:489–497, 1990.

[86] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004.

[87] S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for recognizing natural scene categories. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 2169–2178, 2006.

[88] V. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, 1995.

[89] C. Burges. A tutorial on Support Vector Machines for pattern recognition. *Data mining and knowledge discovery*, 2:1–43, 1998.

[90] N.Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics (SMC)*, 9(1):62–66, 1979.

[91] U. Pal, S. Sinha, and B. B. Chaudhuri. Multi-script line identification from Indian documents. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 880–884, 2003.

[92] M. Ahmed and R. Ward. A rotation invariant rule-based thinning algorithm for character recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 24(12):1672–1678, 2002.

[93] D. Douglas and T. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2):112–122, 1973.

[94] L. Fei-Fei and P. Peronae. A Bayesian hierarchical model for learning natural scene categories. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, pages 524–531, 2005.

[95] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. Alvey Vision Conference (AVC)*, pages 147–151, 1988.

[96] U. Pal, A. Belaid, and Ch. Choisy. Touching numeral segmentation using water reservoir concept. *Pattern Recognition Letters*, 24(1-3):261–272, 2003.

[97] S. Pal, A. Alaei, U. Pal, and M. Blumenstein. Multi-script off-line signature identification. In *Proc. International Conference Hybrid Intelligent Systems (HIS)*, pages 236–240, 2012.

[98] http://lamp.cfar.umd.edu/. Logo dataset. University of Maryland, Laboratory for Language and Media Processing (LAMP), 2014.

[99] R. Mandal, P. P. Roy, U. Pal, and M. Blumenstein. Multi-lingual date field extraction for automatic document retrieval by machine. *Information Sciences*, 314:277–292, 2015.

[100] R. Mandal, P.P. Roy, U. Pal, and M. Blumenstein. Date field extraction from handwritten documents using HMMs. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 866–870, 2015.

[101] J . A. Rodríguez and F .Perronnin. Local Gradient Histogram features for word spotting in unconstrained handwritten documents. In *Proc. International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 7–12, 2008.

[102] N. Dalal and B. Triggs. Histogram of Oriented Gradients for human detection. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893, 2005.

[103] G. A. Fink. *Markov Models for Pattern Recognition, From Theory to Applications*. Springer, Berlin, 2008.

[104] A. Hasnat, S. M. Habib, and M. Khan. A high performance domain specific OCR for Bangla script. In *Novel Algorithms and Techniques In Telecommunications, Automation and Industrial Electronics, LNCS, Springer*, pages 174–178, 2008.

[105] P. P. Roy, U. Pal, J. Lladós, and M. Delalandre. Touching text character segmentation in graphical documents using dynamic programming. *IEEE Transactions on Pattern Recognition*, 45(5):1972–1983, 2012.

[106] U. Pal, S. Sinha, and B. B. Chaudhuri. Multi-script line identification from Indian documents. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 880–884, 2003.

[107] P. P. Roy, U. Pal, and J. Lladós. Morphology based handwritten line segmentation using foreground and background information. In *Proc. International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 241–246, 2008.

[108] U. Pal, N. Sharma, T. Wakabayashi, and F. Kimura. Handwritten numeral recognition of six popular Indian scripts. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 749–753, 2007.

[109] H. Sakoe and S. Chiba. Dynamic Programming algorithm optimization for spoken word recognition. *IEEE Transactions Acoustics, Speech and Signal Processing*, 26(1):43–49, 1978.

[110] R. Mandal, P. P. Roy, and U. Pal. Date field extraction in handwritten documents. In *Proc. International Conference on Pattern Recognition (ICPR)*, pages 533–536, 2012.

[111] S. Roy, P. P. Roy, P.Shivakumara, and U. Pal. Word recognition in natural scene and video images using Hidden Markov Model. In *Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG)*, pages 1–4, 2013.

[112] A. B. Bernard, F. Menasri, R. H. Mohamad, C. Mokbel, C. Kermorvant, and L. Sulem. Dynamic and contextual information in HMM modeling for handwritten word recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 33:2066–2080, 2011.

[113] S. Young. *The HTK Book*. Version 3.4. Cambridge University English Department, 2006.

[114] L. Kuncheva, J. C. Bezdek, and R. P. W. Duin. Decision templates for multiple classifier fusion: an experimental comparison. *Pattern recognition*, 34(2):299–314, 2001.

[115] U. Pal, R. K. Roy, K. Roy, and F. Kimura. Indian multi-script full pin-code string recognition for postal automation. In *Proc. International Conference on Document Analysis and Recognition (ICDAR)*, pages 456–460, 2009.

[116] R. Mandal, S. Pal, P. P. Roy, U. Pal, and M. Blumenstein. Spatial pyramid matching-based multi-script off-line signature identification. *International Journal of American Society of Questioned Document Examiners (ASQDE)*, 18:69–75, 2015.

[117] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.