

Segmentation of Urdu Nastalique



Session: 2012 – 2019

Submitted by:

Qurat ul Ain Akram 2012-PhD-CS-16

Supervised by:

Dr. Sarmad Hussain

Department of Computer Science

University of Engineering and Technology

Lahore Pakistan

Segmentation of Urdu Nastalique

Submitted to the faculty of the Computer Science Department of the University
of Engineering and Technology Lahore in partial fulfillment of the requirements
for the Degree of

Doctor of Philosophy
in
Computer Science.

Internal Examiner

Signature:

Prof. Dr. Sarmad Hussain

Professor, KICS

UET, Lahore

External Examiner

Signature:

Prof. Dr. Khaver Zia

Signature:

Prof. Dr. Faisal Shafait

Chairman

Signature:

Prof. Dr. Shazia Arshad

Dean

Signature:

Prof. Dr. Tahir Izhar

Department of Computer Science

University of Engineering and Technology
Lahore Pakistan

Declaration

I declare that the work contained in this thesis is my own, except where explicitly stated otherwise. In addition this work has not been submitted to obtain another degree or professional qualification.

Signed: _____

Date: _____

Acknowledgments

First of all, I would like to thank my advisor Prof Dr. Sarmad Hussain for giving me an opportunity to work in the exciting domain of document image understanding and recognition, and to develop one of the first Urdu OCR. Development of Urdu OCR was nice opportunity which allowed me to analyze the characteristics of the Nastalique, to develop conventional solution and then eventually to formulate my research using cognitive features which is not done before. I have been very fortunate to have kind support, advice and wonderful guidance of my supervisor throughout my research work. I have been extremely lucky to have a supervisor who cared so much about my work, and who responded to my questions and queries so promptly. His support, encouragement and freedom to devise the scientific solution played important role not only in the completion of my research but also to improve my abilities. His way of analyzing the problem and insight critical questions regarding the solution helped me at various stages of my research. Once again, I offer my heartfelt and sincere thanks to my advisor for his kindness and selfless support, and also for providing me an opportunity to complete my PhD thesis that was exhilarating.

I am also grateful to Mr. Irfan Ahmed Qureshi for teaching me Nastalique calligraphy and providing a detailed knowledge of calligraphy which helped me a lot to polish my research work. I really appreciate his willingness to help and to have meeting with me even during his busy schedule.

My sincere gratitude is reserved for my other teachers Mr. Shafiq-ur-Rahman, Dr. Mehreen Saeed, Dr. Muhammad Aslam and Dr. Usman Ghani who provided mentor and guidance which helped me a lot do the research in this area. I also thank all the departmental staff for cooperation during these years of my study in UET.

I am also very thankful to graduate students, researchers and other staff members of my research center especially Ms. Sana Shams, Mr. Asad Mustafa, Mr. Syed Salman Ali, Mr.Usman Ghani, Mr. Faheem Irfan, Mr. Qasim Ali, Mr. Ehsan ul Haq and Mr. Toqeer Ehsan for providing a creative, helping and friendly

environment. Many thanks to all people especially Dr. Muhammad Awais Hassan, Dr Kahif Javed and Ms. Sana Shams, for their help and suggestions to finalize the presentation of my thesis.

A big Thank you! also goes my parents for their love, prayers, efforts, concerns and motivation which cannot be expressed in words and which actually mean a lot to me to complete my work. A very special thanks to my husband Mr. Ahsan Ali Haroon. I am grateful for everything that you have done for making me able to complete my thesis. Without your help and motivation, the final stages of my thesis would not have been possible.

Many thanks to my best friend Ms. Farah Adeeba not only for her sincere help but also for being there to listen when I needed an ear, and having the hysterical laughter to make me relax. I am lucky to have such a helping, sincere and wonderful friend like you.

Last but not least, a heartfelt thanks to my supportive wonderful family, Dr Ghazala Akram, Mr Waqar Akram, Mr. Waqas Akram, Ms. Naila Akram, Mr. Zeeshan Akram, Dr Marina Akram and Dr Naz Akram for being my support in the moments of difficulties during my research. You all are always there at the time of my needs both mentally and physically. I thank Allah for introducing such people in my life.

Qurat ul Ain Akram

December, 2018

To my parents

&

Husband

Contents

Acknowledgments	iii
List of Figures	x
List of Tables	xii
Abbreviations	xiv
Abstract	xv
1 Introduction	1
1.1 Motivation	4
1.2 Research Hypothesis	5
1.3 Contribution of Thesis	5
1.4 Thesis Structure	7
2 Urdu Writing System	8
2.1 Characteristics of Arabic Script	8
2.2 Characteristics of Urdu	9
2.3 Characteristics of Nastalique Writing Style	11
2.4 Conclusion	14
3 Overview of Perceptual Analysis for Character Identification	15
3.1 Learning	16
3.2 Memory	16
3.2.1 Sensory Memory	16
3.2.2 Short Term Memory	17
3.2.3 Perceptual Analysis of Short Term Memory	18
3.2.4 Long Term Memory	19
3.2.5 Stages of Memory	20
3.2.5.1 Memory Encoding	21
3.2.5.2 Memory Storage	21
3.2.5.2.1 Mental Lexicon	21
3.2.5.3 Retrieval	22
3.3 Reading	22
3.4 Cognitive Models of Reading	24

3.4.1	Selfridge Bottom up Model	24
3.4.2	Rumelhart and McClelland Bidirectional Model	24
3.5	Letter as Basic Unit of Reading	25
3.5.1	Methods of Letter Recognition for Reading	26
3.5.1.1	Template Matching-based Letter Identification	27
3.5.1.2	Confusion Matrix-based Letter Identification	27
3.5.1.3	Feature-based Letter Identification	27
3.6	Factors Impacting Reading or Retrieval	28
3.6.1	Letter Identification	28
3.6.2	Phoneme Information	29
3.6.3	Formatting	29
3.6.4	Reading Ability and Development Age	31
3.6.5	Words Recognition	31
3.6.6	Text on Central Fixation Point	32
3.7	Conclusion	32
4	Automatic Text Recognition	33
4.1	Ligature-based Classification and Recognition	34
4.2	Character-based Classification and Recognition	35
4.2.1	Explicit Segmentation-based Recognition	36
4.2.2	Implicit Segmentation-based Recognition	37
4.3	Conclusion	41
5	Perceptual Experimentation of Urdu Character Identification	42
5.1	Related Work	43
5.2	Methodology	44
5.2.1	Hypothesis of the Study	46
5.2.2	Data Preparation	47
5.2.3	Participants	49
5.2.4	Apparatus and Stimuli	50
5.2.5	Procedure	52
5.3	Results and Discussion	54
5.3.1	Data Analysis	54
5.4	Results and Analysis of Primary, Secondary and Connectors	57
5.4.1	Single Strokes Characters	57
5.4.2	Two Strokes Characters	58
5.4.3	Three Strokes Characters	61
5.4.4	Four Strokes Characters	63
5.5	Discussion	63
5.6	Conclusion	65
6	Image Dataset of Urdu Nastalique Document Images	68
6.1	Dataset-1: Real Image and Ligature RASM Dataset Generation	69
6.1.1	Corpus Collection	70
6.1.2	Corpus Development from Books	71

6.1.3	Corpus Organization	72
6.1.4	Text Corpus as Ground Truth (GT) of Images	74
6.1.5	Semi-Automated RASM Classes Generation	74
6.1.6	Corpus Results	77
6.2	Dataset-2: Synthetic Ligature RASM Dataset	79
6.2.1	High Frequent Urdu Ligatures Selection	79
6.2.2	Image Corpus Development	83
6.2.3	Corpus naming and organization	84
6.2.4	Semi-automated RASM Classes Cleaning	85
6.3	Dataset-3: Synthetic UPTI Dataset Generation	86
6.4	Conclusion	88
7	A Novel Cognitive inspired Computational Framework for Letters of Urdu Nastalique Recognition	89
7.1	Introduction	89
7.2	Research Hypothesis	91
7.3	Proposed Cognitive Inspired Framework	91
7.3.1	Image Demon Formulation	92
7.3.2	Stroke Demon Formulation	93
7.3.3	Character/Ligature Demon Formulation	94
7.4	Conclusion	97
8	Development of Cognitive-inspired Framework for Urdu Character-based Recognition	98
8.1	Introduction	98
8.2	Image Demon	101
8.2.1	Thinning	101
8.2.2	Traversal	102
8.2.3	Computation of Features	102
8.3	Stroke Demon	103
8.3.1	Investigation and Labeling of Strokes	104
8.3.2	Training and Recognition of Cognitive Features	105
8.3.3	Analysis and Improvements	106
8.4	Character/Ligature Demon	106
8.4.1	Development of Probabilistic Model	108
8.4.2	Character Corpus Development and Probabilities Computations	110
8.4.3	Strokes Corpus Development and Probabilities Computations	110
8.4.4	Lexicon Development	111
8.5	Conclusion	114
9	Results and Discussion	118
9.1	Introduction	118
9.2	Dataset	118
9.3	Results on Dataset-1 and Dataset-2	119

9.3.1	Character Recognition Accuracy of M1	119
9.3.2	Character Recognition Accuracy of M2	120
9.3.3	Character Recognition Accuracy of M3	120
9.3.4	Character Recognition Accuracy of M4	121
9.4	Results on Dataset-3	122
10	Conclusion	125
	References	128
	Author Bibliography	128
	Author Bibliography	128

List of Figures

1.1	Text Written in Naskh Writing Style	3
1.2	Text Written in Nastalique Writing Style	3
2.1	Different Writing Styles for Arabic Script [37]	9
2.2	Urdu Characters and Coressponding RASM and IJAM	9
2.3	BEH Character Shapes at Initial Medial, Final and Isolated Positions in Naskh Writing	10
2.4	Urdu Character Set	10
2.5	Bidirectional Writing [104]	10
2.6	Urdu RASM Classes	11
2.7	Diagonality of Text in Nastalique Writing Style	12
2.8	Character Overlapping (Highlighted with Green Rectangle) and Ligature Overlapping (Highlighted with Red Rectangles) Overlapping	12
2.9	Text Written in Naskh Writing Style	12
2.10	Thick-thin Stroke Variation Across Characters in a Ligature	12
2.11	Nuqta Placement of YEY in Two Different Ligatures Highlighted with Red Rectangle	13
2.12	Same Font Size Text having Variation in Line Height	13
2.13	Examples of Fifteen Initial Shapes of Letter SWAD and FAY in Two Letter Ligatures	14
2.14	Contextual Character Shaping of Character BEH at Initial Position in Nastalique	14
3.1	Structure of Human Memory	17
3.2	Working Memory for Encoding and Retrieval [12]	19
3.3	Lexicon Network [108] According to Willem Levelt [57]	23
3.4	Layers of Visual Word Recognition [84]	25
5.1	Single Stroke Character	45
5.2	Two Strokes Characters, First and Second Strokes in Sequence Highlighted with Red and Green Colors, Respectively	45
5.3	Three Strokes Characters, First, Second and Third Strokes in Sequence Highlighted with Red, Green and Blue Colors, Respectively	46
5.4	Four Strokes Characters, First, Second, Third and Fourth Strokes in Sequence Highlighted with Red, Green, Blue and Black Colors, Respectively	46

5.5	Stroke Sequence of Urdu characters, Main Stroke is Represented by Black Color, Remaining Main Body in Gray Color and Dots are Shown with Green Color	49
5.6	Single Stroke Visual Stimuli	51
5.7	Two Strokes Visual Stimuli	51
5.8	Three Strokes Visual Stimuli	52
5.9	Four Strokes Visual Stimuli	52
5.10	Display Example of Jeem Character as Input and Jeem_S ₂ S ₃ Visual Stimulus for Recognition Task	54
5.11	Contextual Shapes of S ₂ Stroke of Character BEH Highlighted with Black Color	57
5.12	Contextual Character Shaping of Laam, Primary Stroke Highlighted with Black and Connector Highlighted with Gray Color	60
6.1	Urdu Character Set	70
6.2	Sample of Gray Scale Cropped and Un-cropped Region of Interest [9]	72
6.3	Sample of Image and Corresponding Typed Text	74
6.4	Sample image having 35 tokens of ligature RASM class	84
7.1	Architecture of cognitive based character recognition	92
8.1	Architecture of cognitive based character recognition	100
8.2	Position coded visual features extraction	101
8.3	Examples of Visual Confusion Between Strokes	107
9.1	Process Flow of Integrated Urdu OCR Framework	123
10.1	Marked strokes of character SWAD in calligrapher's book [93]	143
10.2	Marked strokes of character SWAD in calligrapher's book [80]	143

List of Tables

5.1	Characters' Strokes along with Participants Accuracy for Single Stroke Character	57
5.2	Primary, Secondary and Connectors of Single Stroke Letters	58
5.3	Characters' Strokes along with Participants Accuracy for Two Strokes Characters	59
5.4	Primary, Secondary and Connectors of Two Strokes Letters	61
5.5	Characters' Strokes along with Participants Accuracy for Three Strokes Characters	62
5.6	Primary, Secondary and Connectors of Three Strokes Letters	63
5.7	Characters' Strokes along with Participants Accuracy for Four Strokes Characters	64
5.8	Primary, Secondary and Connectors of Four Strokes Letters	65
5.9	Characters having Single Stroke as Primary Stroke represented with black color and diacritics with green color	66
5.10	Characters having Two Strokes as Primary Stroke	67
5.11	Characters having Three Strokes as Primary Stroke	67
6.1	Special Cases of Wrong Classification	76
6.2	Main Body Classes Confusions	77
6.3	Statistics of Urdu Image Corpus [9]	78
6.4	Font Wise Lines and Ligature Statistics of Corpus	79
6.5	Font Wise Instance Images Statistics	80
6.6	Instance Images of Ten Main Body Classes	81
6.7	Character Classes Mapping List	82
6.8	Broken and Diacritics-attached Examples of Ligature RASM Classes	85
6.9	Sample textlines rendered in Noori Nsatalique(a) and Alvi Nastalique (b)	87
8.1	Examples of Ligature Transcription in Terms of Strokes Sequences of Characters in Reverse Order	105
8.2	Ligature Transcription after Resolving Shape Confusions of Strokes	107
8.3	Primary, Secondary and Connector Strokes of Urdu Characters	109
8.4	Sample Entries of Strokes Sequence Corpus for M1	111
8.5	Sample Entries of Strokes Sequence Corpus for M2	111
8.6	Sample Entries of Strokes Sequence Corpus for M3	112
8.7	Sample Entries of Strokes Sequence Corpus for M4	113

8.8	Character Lexicon for Model M1	114
8.9	Character Lexicon for Model M2	115
8.10	Character Lexicon for Model M3	116
8.11	Character Lexicon For Model M4	117
9.1	Character Sequence Recognition Accuracy of M1, M2, M3 and M4 .	122
9.2	Ligature Length Wise Character Sequence Recognition Accuracy of M1, M2, M3 and M4	122
9.3	Comparison with State-of-the-art Urdu Recognition Techniques . .	124

Abbreviations

OCR	O ptical C haracter R ecognition
ICTs	I nformation and C ommunication T echnologies
HMM	H idden M arkov M odels
BW	B aum W elch
RNNs	R ecurrent N eural N etworks
LSTM	L ong S hort T erm M emory
BLSTM	B idirectional L ong S hort T erm M emory
CNN	C onvolution N eural N etwork
DCT	D iscrete C osine T ransformation
SVM	S upport V ector M achine

Abstract

The Optical Character Recognition (OCR) system is significantly used as assistive technology to convert the document images into computer editable format. After converting into the editable format, these documents can be ported online with less time and human effort. In addition, fragile historical published documents can be preserved and made accessible to local users by using this technology.

The development of a robust OCR system for Urdu language especially text written using Nastalique, is a challenging task due to its cursive nature and complex characteristics. The conventional methods for the recognition of Urdu document images are classified into the two main categories (1) ligature-based classification and recognition and (2) character-based classification and recognition. To develop the recognition systems, usually dimensional, structural and geometrical features are extracted from the images using image processing techniques, and classified using state of the art machine learning and deep learning based approaches. However, for the Arabic like complex cursive script especially for text written in Nastalique writing style, these conventional methods have some drawbacks to recognize the characters and words. The overlapping of characters and ligatures makes the system more complex especially feature computation algorithm. The computed features from horizontally sliding windows which have overlapped character strokes are actually noisy features for the classification and recognition of respective character, and eventually introduce confusions. In addition, complex contextual character shaping also adds complexity resulting in the misrecognition of characters. Due to these complexities, system generates errors by introducing the character insertions and deletions in the recognized text.

Fortunately human brains are prone to complex character shaping once human learns how to read the text. The characters' shapes similarity can easily be disambiguated by human. In addition, the overlapping of the characters and ligatures are very intelligently processed by eyes to convert the object into the sequence of features so that human brain can recognize it.

In this thesis, cognitive-based character recognition framework is presented which is inspired by the human model of reading the text. Up till now, no research exists in the literature to recognize the images of Arabic characters and ligatures using the cognitive model of reading. This framework is based on cognitive features which are used to recognize characters. This cognitive-based character recognition framework is aimed to develop a robust character recognition system for Urdu.

In this thesis, the complete perceptual experiment is described, which extracts and classifies character's features i.e. strokes into three categories (1) primary strokes which represent core shape of a character, do not change in different context and play significant role for the recognition of character, (2) Secondary strokes which do not play significant role for character identification, however, along with the primary strokes improve the confidence for the recognition of the character and (3) Connectors which give only contextual positioning information of a character. The detailed experiment is carried out on complete character set having all 21 Urdu characters' RASM classes. Response of the participants for each character is analyzed. The categorization of strokes as primary, secondary or connector is done after detailed analysis of results.

Based on findings of the perceptual experiment, a cognitive-based computational framework for the recognition of Urdu characters is developed which can also be used for the recognition of other Arabic script languages. To test the strength of the framework, two different image datasets of Urdu document images written using Nastalique writing style are used. The results show that this character recognition framework based on the primary strokes outperforms the state of the art HMMs and deep learning based Urdu recognition techniques.

Chapter 1

Introduction

Access to online information is becoming a critical factor in the development of nations due to tremendous advancements in the development of the Information and Communication Technologies (ICTs). Majority of the available online content is in English and other computationally developed languages. Urdu is spoken by 163 million people all over the world. Among them 69 million people use Urdu as first language and 94 million as second language¹. Significant amount of Urdu literature is published in the form of books, magazines and newspapers. Users from different geographical areas, socio-economic backgrounds and educational institutions must have equal access to this published Urdu content. In addition, the old published historical and cultural books which are national heritage of Pakistan, and most of which are fragile, need to preserve for future generations. Most of the available published Urdu content is in the form of images. The images are heavy in sizes which cause slow data transfer rate over the Internet. In addition, information retrieval of such content is challenging because images are not searchable.

Therefore an assistive technology i.e. OCR is required which converts Urdu document images into computer editable format. Urdu OCR will facilitate local users to port Urdu content online by accelerating the process of converting the document images into editable format. In addition, fragile historical published documents can be preserved and made accessible to local users by using this technology. This

¹<https://www.ethnologue.com/language/urd>

assistive technology integrated with Text to Speech (TTS) system will also help to give access to print disabled community; both illiterate and visually impaired. Hence the development of Urdu OCR is critical factor for socio-economic development of Urdu speaking community. The editable text can be searchable and accessible so that users can retrieve desired information through online search engine. Users can have round the clock accessibility of the information related to the language and literature. In addition, they can get authentic information from online accessible published books, magazines and newspapers. OCR systems are widely used in other fields such as mail sorting by automatic address image reading, digital conversion of published proceedings of conferences and issues of journals publications, business card and invoice images conversions etc. In short, OCR is helpful for all different types of industries which are doing printed document management and processing.

OCR has three modules; (1) Preprocessing, (2) Classification and Recognition, and (3) Post-processing. Preprocessing module deals with the processing of an input image to improve its quality and to segment image into different areas. The relevant information from these areas is extracted which is used in classification and recognition, and post-processing modules. The sub-modules of preprocessing include binarization which converts the colored/gray document images into black and white format. Then the page is segmented into figure and text areas. These text areas are sequenced into column(s) according to the reading order. Eventually text areas are segmented into text lines, words, ligatures and characters by doing image processing and using different characteristics of writing style. The classification and recognition module has two phases. The first phase called training phase deals with the training of character or ligature shapes into different classes using extracted features. In the recognition phase, the features of input shape are computed and recognized using the trained classifier. The post-processing phase deals with the formation of words and sentences using recognized characters/ligatures sequences. This module deals with the word segmentation, spell checker and Part of Speech (POS) tagger etc. sub-modules, and generates correct sequence of words to form sentences.

Arabic script is cursive in nature. Arabic language is normally written using Naskh writing style. In Naskh writing style, characters of a ligature are written along the baseline, highlighted with red line in Figure 1.1

FIGURE 1.1: Text Written in Naskh Writing Style

Urdu language also belongs to Arabic script and is bidirectional, i.e. text is written from right to left and digits are written from left to right. Nastalique is preferred writing style used to write Urdu books, magazines, newspapers and governmental official documents. It has context sensitive character shaping, diagonally growing characters in ligature causing vertical overlapping of characters and ligatures, see Figure 1.2. In addition, Nastalique has complex marks placement rules for dots and aerabs. These and other characteristics of Nastalique make Urdu text recognition a challenging task.

FIGURE 1.2: Text Written in Nastalique Writing Style

Over past decade, recognition of Urdu document images has received a lot of attention mainly focusing on the classification and recognition module of OCR [98] [74][48][5]. Different HMMs based and deep learning based techniques are used to develop state-of-the-art recognition systems. These state-of-the-art recognition techniques are divided into two approaches; (1) ligature-based recognition and (2) character-based recognition. The ligature-based recognition techniques extract features from ligature image. The features along with the sequence of ligatures' transcriptions are used to train the system using different classification techniques[5] [49] [52]. The character-based recognition techniques extract characters' features from the images. The features along with the character transcriptions are used to train the classification system. There are two categories of character-based recognition techniques; (1) Explicit segmentation-based recognition and (2) Implicit segmentation-based recognition. Explicit segmentation-based recognition techniques segment the document image into characters by using image processing

techniques and writing style characteristics. The features from the segments are extracted, and along with characters labeling are used to train the system. The implicit segmentation-based recognition techniques use the concept of sequence learning using state-of-the-art sequence learning techniques. The text images with corresponding characters/ligatures/words labels are fed to sequence learning approaches including HMMs [3][41][69][83][87], Recurrent Neural Networks (RNNs) and variants of RNNs [31] [30]. These sequence learning techniques are algorithmically strong enough to learn long context by extracting similar shape patterns and assigning labels to these patterns accordingly. The extensive research is available on use of HMMs and RNNs with variants to develop robust character recognition systems for different languages such as Arabic, Chinese, Devanagari and Urdu etc., by tweaking the feature extraction approaches, devising efficient labeling and tweaking parameters of the learning system.

1.1 Motivation

Despite state-of-the-art Urdu document image recognition approaches have promising results, still practical use of developed systems have drawbacks. Urdu character set having 38 letters of Urdu constitutes around 25,000 commonly used unique ligatures. This ligature set is not closed because addition of new word in Urdu language which can be a transliterated word of foreign language, may cause addition of new ligature. Therefore the ligature based solution for the recognition of Urdu text is not appropriate. Whenever a new ligature would be added in the language, the system would require to be retrained on the additional ligature so that this can be recognized.

To handle this issue, the character-based system seems to be an optimal solution. As Urdu character set is close set therefore addition of new ligature would not affect the performance of recognizer as new ligature will be segmented into characters which would be recognized by the recognizer. However, both character recognition techniques have some challenges. For example, the horizontally sliding window used to extract the features of the respective characters would have confusing features when a character overlaps with other character in a ligature. In addition,

the recognition of the contextual character shaping is also a challenging task. Due to these challenges, the existing character-based recognition techniques generate errors by introducing character insertions and deletions in the recognized text.

In this thesis, the main focus of the research is to solve these issues by presenting a recognition framework inspired by cognitive model of reading. This cognitive-based recognition framework is based on the recognition of characters' strokes by extracting stroke-based features of a ligature image. This technique resolves noisy feature computation due to the overlapping of characters in a ligature. Later, the recognized strokes are weighted using the cognitively inspired mathematical model to recognize the characters effectively. The character insertion and character deletion is also handled by the presented recognition framework.

1.2 Research Hypothesis

The main research hypothesis in this thesis is based on the argument that the computational model of cognitive based human reading will improve the character recognition. Human brains are prone to complex character shaping once human learns how to read the text. In addition, characters and ligatures are very intelligently processed by eyes to convert the object into the sequence of cognitive features (i.e. strokes) so that human brain can recognize it. The character shapes similarity can easily be disambiguated by human using contextual information. In most of cases, human brain focuses on the core shape of a character and generates the recognition decisions. Later the generated decisions are filtered to recognize characters and words by doing contextual processing. In the same way, the cognitive-based implicit character recognition framework will improve the character recognition results.

1.3 Contribution of Thesis

The summary of the major contributions of this thesis are given below.

- The perceptual experiment is presented to extract the cognitive features in terms of identifying the primary, secondary and connector strokes of Urdu

characters. The complete experiment is carried out and after detailed analysis of the results, strokes are classified into these categories.

- Development of comprehensive image dataset extracted from books along with ground truth information is presented. In addition, development of synthesized dataset along with the ground truth information is also presented.
- A semi-automated approach is presented to extract and clean the ligatures and main bodies for the development and evaluation of the presented and other state-of-the-art Urdu character recognition techniques.
- Development of ligature based Urdu text recognition is presented. The state-of-the-art, multilingual open source engine i.e. Tesseract is modified to adapt for the recognition of Arabic like Urdu cursive script.
- The implicit segmentation-based character recognition approach based on HMMs as classifier is also presented. The multiple contextual shapes of all Urdu characters are used for the classification and recognition.
- Another implicit segmentation-based technique is presented which divides the ligature into the sequence of characters and joiners instead of only the sequence of characters.
- A font size independent recognition technique is also presented to handle the recognition of text having multiple font sizes without developing recognizers at each font size. Instead of using text line normalization according to the line height, the ligature image is horizontally and vertically normalized to the standard font size by applying respective scaling factor to maintain the aspect ratio of the image.
- The cognitive based character recognition framework is mathematically designed and developed which is based on the findings of perceptual experiment of Urdu characters. This cognitive based recognition framework can be used for the recognition of text written in any Arabic script language. To check the strength of the presented framework, the test dataset of Urdu document images which is used in state-of-the-art Urdu character recognition techniques, is also used for evaluation.

1.4 Thesis Structure

This thesis is structured into different chapters. The current chapter discusses the basic introduction about OCR, challenges for the development of Urdu OCRs, and proposed solution which is core of this research. Chapter 2 discusses the characteristics of Urdu language and Nastalique writing style. Chapter 3 gives the background study of perceptually identification of cognitive features used for the characters recognition and eventually for reading the text. Chapter 4 discusses the detailed perceptual experimentation to extract the cognitive features of Urdu characters which normal Urdu speaker uses to identify for reading. Chapter 5 presents the conventional methods which are used for the recognition of Urdu document images using ligature-based and character-based recognition approaches. In Chapter 6, the development of comprehensive image dataset along with the ground truth information is discussed. The evaluation dataset i.e. UPTI is also discussed in this chapter. Chapter 7, which is main crux of the research, presents cognitive-inspired character recognition framework along with mathematical formulation. In Chapter 8, the implementation details to develop the cognitive-inspired recognition system for Urdu document images are presented. Chapter 9 presents the results and also presents the comparisons with other techniques. At the end Chapter 10 concludes the thesis.

Chapter 2

Urdu Writing System

2.1 Characteristics of Arabic Script

Urdu belongs to the Arabic script which is cursive in nature. Arabic script based languages are written using different writing styles. The Nastalique, Naskh, Kufi, Sulus, Deewani and Riqah are commonly used writing styles [37], text sample of each writing style is given in Figure ???. Each writing style is cursive in nature and has its own rules for character shaping in different contexts. Arabic language is normally written using Naskh writing style and Urdu is normally written in Nastalique writing style. The development of OCR for document images of Arabic script languages is a challenging task.

In Arabic script, one or more characters form ligature [60] and ligatures are grouped to form the words. Each ligature has two parts; (1) RASM and (2) IJAM [106]. RASM also called main body, is the main stroke without the associated diacritic, see Figure 2.2(b). IJAM are mandatory diacritics used to disambiguate the same RASM having different consonant behaviors as can be seen in Figure 2.2(c). A RASM can or cannot contain IJAM which is dependent on consonantal behavior. The RASM of character ر (REH) does not have any IJAM whereas RASM of character ر (Rreh) has an IJAM which is indicated in Figure 2.2(c) corresponding to ر (Rreh) character.

Naskh is a special calligraphic style for writing text of Arabic language which

نستعلیق	وَسَخَّرَ الشَّمْسَ وَالْقَمَرَ
نسخ	وَسَخَّرَ الشَّمْسَ وَالْقَمَرَ
کوفی	وَسَخَّرَ الشَّمْسَ وَالْقَمَرَ
ثلث	وَسَخَّرَ الشَّمْسَ وَالْقَمَرَ
دیوانی	وَسَخَّرَ الشَّمْسَ وَالْقَمَرَ
رقاع	وَسَخَّرَ الشَّمْسَ وَالْقَمَرَ

FIGURE 2.1: Different Writing Styles for Arabic Script [37]

ب	ک	ط
گ	ر	ط
ر	ر	ط
(a) Character	(b) RASM	(c) IJAM

FIGURE 2.2: Urdu Characters and Coressponding RASM and IJAM

is developed by Ibn Muqlah [103]. It is written using a special pen called cava pen. Naskh is commonly used to write the Arabic books including the Quran and Hadiths etc. due to its ease of legibility. In Naskh writing style, the characters are horizontally joined together on baseline to form the ligatures. Therefore ligatures are not overlapped, see Figure 1.1. Due to this property of the Naskh, the characters of the ligatures can be easily segmented on the baseline. Based on the position in a ligature, each character has four basic shapes i.e. at initial, medial final and isolated positions, see Figure 2.3, example of ب (BEH) character.

2.2 Characteristics of Urdu

Urdu language belongs to the group of languages that use Arabic script. It is inherited and influenced by Persian and Arabic languages. Urdu was developed

ب	کب	قبر	بنکر
Isolated	Final	Medial	Initial

FIGURE 2.3: BEH Character Shapes at Initial Medial, Final and Isolated Positions in Naskh Writing

by the Mughal Empire during the rule in the sub-continent [104]. Urdu has a total of 39 characters in alphabet set which is given in Figure 2.4. Urdu characters also have the diacritics which include Nuqtas and Aerabs [38]. In addition, digits, special symbols such as punctuation marks and honorific marks etc are also available in Urdu [39]. The characters are joined together to form ligatures and words. Usually each character changes its shape when it is joined to its neighboring characters. The shape of a character depends on the context in which it appears. The diacritics of each character may appear above or below the main body of the ligature.

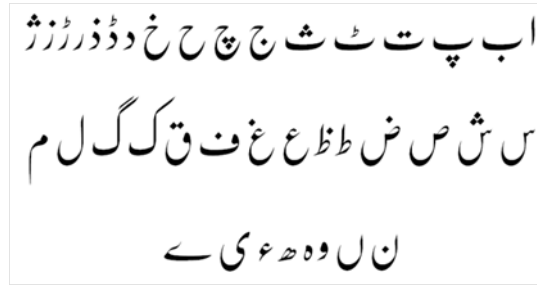


FIGURE 2.4: Urdu Character Set

Urdu is bidirectional language [104] i.e. words in the Urdu are written from right to left whereas digits are written from left to right as can be seen in Figure 2.5.

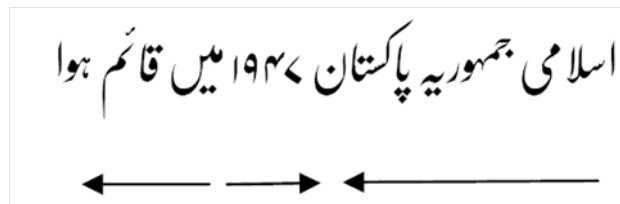


FIGURE 2.5: Bidirectional Writing [104]

Based on the shape similarity of Urdu RASMs, ligature are classified into different classes. The Urdu character set has a total of 21 unique RASMs which are given in Figure 2.6. Figure 2.6(a) specifies name of RASM which will be used throughout

this Thesis, Figure 2.6(b) indicates shape of RASM of Urdu character, and all characters with possible variations of diacritics of same character class are shown in Figure 2.6(c).

Alif Class	ا		FEH Class	ف	ف
BEH Class	ب	ب پ ت ٹ ث	QAF Class	ق	ق
JEEM Class	ج	ج چ ح خ	KAF Class	ک	ک گ
DAL Class	د	د ذ	LAAM Class	ل	ل
REH Class	ر	ر ز ژ	MEEM Class	م	م
SEEN Class	س	س ش	NOON Class	ن	ن
SUAD Class	ص	ص ض	WAO Class	و	و
TOAY Class	ط	ط ظ	HEY Class	ہ	ہ
AIEN Class	ع	ع غ	HEY DO-CHASHMY Class	ھ	ھ
YEH Class	ی	ی	HAMZA Class	ء	ء
YEH-BARI Class	ے	ے			
(a) RASM Class Name	(b)RASM Shape	(c) Characters having same RASM	(a) RASM Class Name	(b)RASM Shape	(c)Characters having same RASM

FIGURE 2.6: Urdu RASM Classes

2.3 Characteristics of Nastalique Writing Style

Most of published Urdu content i.e. printed documents, newspapers and books is written using Nastalique writing style. This writing style was developed in the 14th century CE in Iran by combining the rules of two writing styles, Naskh and Talq [33]. Due to its beauty and efficient adjustment of text on paper, this writing style is extensively used to write the content of Arabic script languages including Balochi, Kashmiri, Pashto, Persian, Punjabi, Saraiki, Turkic and Urdu in Afghanistan, India, Iran, Pakistan and other south Asian countries.

A special pen called QALAM is used to write text in Nastalique. Nastalique is a complex writing style as compared to Naskh. In Nastalique writing style, the QALAM is diagonally moved from top right to bottom left to write the characters in a ligature as can be seen in Figure 2.7, the diagonal nature of the Nastalique is

indicated with arrows. This diagonality of the writing style causes characters as well as ligatures to overlap as can be seen in Figure 2.8. The character overlapping is highlighted with green rectangle and ligature overlapping is highlighted with red rectangle.



FIGURE 2.7: Diagonality of Text in Nastalique Writing Style



FIGURE 2.8: Character Overlapping (Highlighted with Green Rectangle) and Ligature Overlapping (Highlighted with Red Rectangles) Overlapping

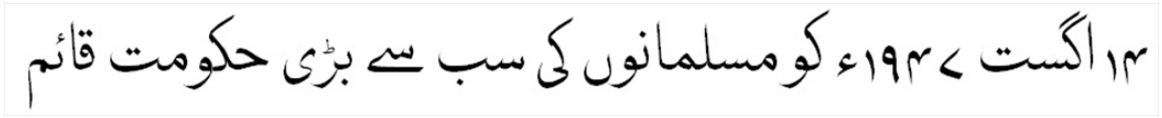


FIGURE 2.9: Text Written in Naskh Writing Style

Further, ligature stroke has thick-thin stroke variation across characters as shown in Figure 2.10. In a ligature, initially the character has thick stroke. The stroke becomes thin at the position where next character is attached and then again for actual strokes of this character, the stroke again becomes thick.

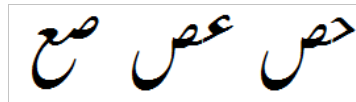


FIGURE 2.10: Thick-thin Stroke Variation Across Characters in a Ligature

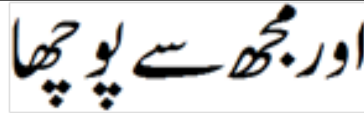
There is no hard and fast rule for placement of diacritics. The aerab and nuqta are attached above or below the RASM i.e. main body of ligature. The position of diacritics also depends on the context of a character in a ligature, see Figure 2.11, the two different placements of Nuqtas of ی (YEY) in two different ligatures are highlighted with red rectangle.



FIGURE 2.11: Nuqta Placement of YEY in Two Different Ligatures Highlighted with Red Rectangle



(a) Line Height of 111 Pixels



(b) Line Height of 87 Pixels

FIGURE 2.12: Same Font Size Text having Variation in Line Height

In addition, ligature height grows diagonally upwards as various characters are cursively joined, varying the line height. The text lines of the same font size can have varied line height due to varying height of ligatures in text line as can be seen in Figure 2.12.

In Nastalique each character has many different contextual shapes based on the position in a ligature [100], see Figure 2.13 and Figure 2.14. This contextual character shaping is based on the preceding and succeeding characters. The detailed analysis of each character shape based on its position in a ligature is reported in [40] and [100]. These contextual shapes of each character have different joining rules when attached with different characters. In Nastalique, the character shape inventory contains a total of 998 unique shapes which can be further reduced to 488 unique shapes when dealing with character RASM classes. The images of two letter ligatures showing the examples of initial shapes of SUAD and FAY [40], highlighted with black color, are illustrated in Figure 2.13.

SWAD shapes	صا	صب	صح	صر	صس	صص	صط	صح
	صف	صم	صن	صه	صه	صي	صے	
FAY shapes	فا	فب	فج	فر	فس	فض	فظ	فع
	فف	فم	فن	فو	فه	فی	فے	

FIGURE 2.13: Examples of Fifteen Initial Shapes of Letter SWAD and FAY in Two Letter Ligatures



FIGURE 2.14: Contextual Character Shaping of Character BEH at Initial Position in Nastalique

2.4 Conclusion

In this chapter, characteristics of Arabic writing style are discussed. Arabic is the cursive script. Different characters are joined to form the ligatures. Each ligature has two main components i.e. RASM and IJAM. The characteristics of Urdu writing style specifically Nastalique writing style are also discussed. Nastalique is the complex writing style. The complex characteristics of Nastalique includes diagonality of writing Urdu characters in a ligature, overlapping of characters and ligatures, thick-thin variation of stroke and complex rules for the placement of Nuqtas of characters in a ligature etc. All such characteristics make the recognition of Urdu document images a challenging task.

Chapter 3

Overview of Perceptual Analysis for Character Identification

In this chapter the background study related to the cognitive development involved in reading the written text is discussed. Reading is a cognitive process having two main components i.e. word recognition and comprehension[34]. Word recognition also called decoding, manipulates the visual input to identify the visual representation of character strokes and to convert it into sounds [102]. The conversion of letter information to the phonemic information is important for word recognition [46]. The reading comprehension is reader's ability to process the recognized text, identify the meaning of words by using the contextual information stored in memory [27]. Reader also uses contextual clues and tries to recognize and identify the meaning of unknown words[102]. Therefore, reading is a process which retrieves the stored information to recognize the sequence of strokes and characters. The words are formed using recognized sequence of strokes, characters, phonological information, lexical processing and semantic processing of the language developed in the brain [57] [91][84]. To read the text, language development is the first phase in which a language is developed. This language is developed through cognitive processes of learning in which different words are acquired along with their meanings. The utterances, usage and linguistic information of these words are also learnt [65]. Different cognitive processes including learning, storage and retrieval

are involved to read the text. Based on learning, the language is developed in the human brain. The letters, words and their linguistic information are stored in human brain in a structured format [57]. There are three main cognitive stages which are mainly involved to understand the cognitive process of reading. These three stages are (1) Learning, (2) Memory and Storage, and (3) Reading. Each of these is discussed in detail in subsequent sections.

3.1 Learning

In cognitive science the term "learning" is referred to as a process of acquiring new information which results in change of long-term memory[54]. According to Walter et al. [13], there are three modalities which are normally used by humans to acquire and process new experiences which result in learning. These are given below.

1. Visual: The demonstration of picture or shape which can be used for efficient learning
2. Auditory: The auditory such as audio(or rhymes) is another way of learning
3. Kinesthetic: Object simulation and body movement are also used to teach the children for learning

3.2 Memory

Our memory is updated or created when something is learned. Hence, there is direct relationship of learning and memory. Atkinson and Shiffrin [82] define general structure of human mind. They define that human memory has three major components i.e. sensory memory, short-term memory and long-term memory. The interaction of the memory components involved in learning is shown in Figure 3.1.

3.2.1 Sensory Memory

The environmental input information which enters in the human brain through any of the sense organ (i.e. eye, ear, nose, hand and lip) is maintained in the sensory memory [17] and [82]. Each sense organ converts the environmental input into

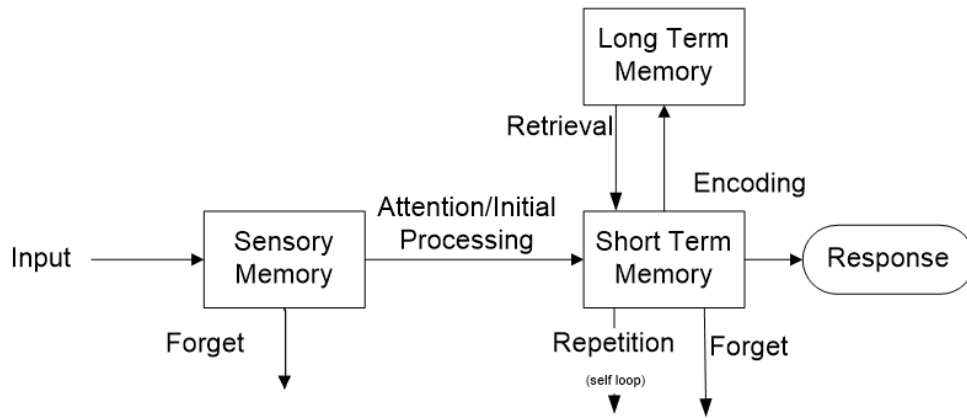


FIGURE 3.1: Structure of Human Memory

the form which brain can understand. After conversion, the information is stored in sensory memory. There are mainly two types of sensory memory i.e. echoic memory which is generated through our auditory sense and iconic memory which is generated through our vision sense (such as shape, size, color and location). The time span of the sensory memory is very limited i.e. less than 1/2 second for vision and about 3 seconds for hearing.

3.2.2 Short Term Memory

The attentional processes move selected input from sensory memory to the short term memory [22]. This is conscious memory which is generated by paying attention on the environmental input and to relate it with the internal memory information. Short term memory also does chunking which is the process of combining individual small units of information into some form of meaningful units. Short term memory has temporary storage which has storage duration of approximately 15 to 20 seconds. According to Atkinson and Shiffrin [82], the information in the short term memory can be lost due to the two reasons; (1) Decay and (2) Interference. The decay is the state when information is lost over time due to the lack of rehearsal. The Interference means when new information overrides old information.

The short term memory has working memory for storage and manipulation of

selected environmental input. Baddeley [12] presents the structure of the working memory which has four major components as can be seen in Figure 3.2. The summarized description of each component is given below.

- (I) Phonological Loop (PL): This component manipulates and stores spoken and written material. It has two sub-components;
 - (a) Phonological Store (inner ear), abbreviated as PS which holds speech based information (spoken words) for 1-2 seconds. Basically it deals with speech perception.
 - (b) Articulatory Control Process (inner voice) which rehearses and stores verbal information from the Phonological Store (PS). Articulatory control process deals with speech production.
- (II) Visuo-Spatial Sketchpad (VSS): This component manipulates and stores visual and visuo-spatial information.
- (III) Episodic Buffer: This serves as backup which links the visual, spatial and verbal information with time sequence or chronological order to form the integrated unit (e.g. memory of a story or a movie scene). It communicates with PL, VSS and long term memory to do its task.
- (IV) Central Executive: It is controller of the working memory which is responsible to monitor, control and manage coordination of Phonological Loop (PL), Visuo-spatial sketchpad(VSS) and Episodic Buffer. It also relates PL and VSS to long term memory. The central executive decides which information is important and which parts of the working memory have to process this information.

3.2.3 Perceptual Analysis of Short Term Memory

During learning, both visual and phonemic information is normally used to teach so that children or students can better learn. Hue and Erickson [36] investigate the storage component of short term memory which is used for reading. They

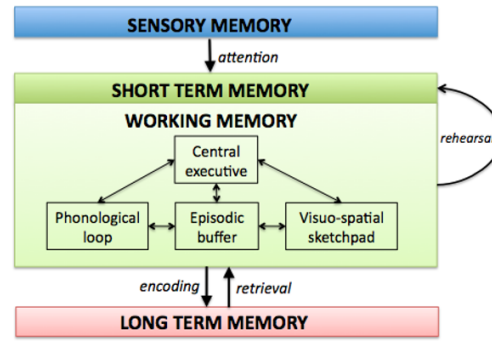


FIGURE 3.2: Working Memory for Encoding and Retrieval [12]

concluded that the characters which are known to the reader are stored in the phonological loop in verbal form otherwise it is stored in VSS in visual form.

The shape similarity at encoding and retrieval effects the performance of short term memory. Judit and Josep [62] study the processing for similar shapes in visual working memory at encoding and retrieval to examine the effect of similarity at recognition. In this experiment serial recall [32] task is used for evaluation. They conduct experiments on Chinese character set. Different sizes of lists having four combinations of similar items at encoding, similar items at retrieval, dissimilar items at encoding and dissimilar items at retrieval are presented to subjects and their recognition responses are stored. The experimental results of 48 undergraduate students response show that the large size of the list at encoding level decreases the recognition rate. Subjects response time is higher when list size is higher. Similar items at encoding level and dissimilar items at retrieval level improve recognition.

3.2.4 Long Term Memory

The long term memory retains the information for the lifetime. It has unlimited capacity which cannot be measured. The rehearsal of information in the short term memory causes information to be stored in long term memory. The long term memory is further divided into two categories i.e. declarative and procedural memory [22] which are discussed below.

1. Declarative Memory: The declarative memory stores the information which is in the form of facts and events which is communicated verbally. This is also

called explicit memory. An example of declarative memory is describing basic principle of math to students. It is further divided into two components [22]; (1) Episodic and (2) Semantic. Episodic stores the events about the life especially events of the personal history. The example includes meeting with student or friend. This involves conscious awareness of past events which is personal and autobiographical memory. The Semantic stores factual information in nature especially about the world. It deals with knowing the things such as how to subtract two numbers, differentiation of animals, birds and humans. The semantic knowledge is different from the recollection of events of episodic memory.

2. Non-declarative Memory: The memory in which information is stored in the form of skills and cognitive operations. This is implicit memory which is not consciously recollected in the form of specific events or facts [95].

Human abilities to drive a car is an example of this type of memory. This type of memory does not require any type of recollected experience. Normally this is built on the basis of day by day experience. The presence of knowledge from explicit memory is not necessary. It has three sub-categories; (1) Procedural Memory, (2) Classical conditioning, and (3) Priming. The Procedural memory involves learning from motor skills (such as how to drive a car) and cognitive skills (such as how to read). The Classical conditioning deals with retrieval/response of the stimulus based on the pairing of conditioned stimulus and unconditioned. The conditioned stimulus has conditioned response (food versus dog salivation) and unconditioned stimulus has unconditioned response. Priming deals with the primed objects and words. The Priming is the process to identify the change in the stimulus based on the prior exposure of that stimulus. This is based on the idea that primed objects and words which are previously shown are more easily recognizable as compared to the objects and words which are not primed (shown previously).

3.2.5 Stages of Memory

”Memory is the process of maintaining information over time” [63]. Memory is very important in human daily lives. Memory is important to process the information of different types such as images and sounds. Without memory human cannot

remember anything, cannot think and even cannot not learn. There are three stages of Memory; (1) Encoding, (2) Storage and (3) Retrieval [66].

3.2.5.1 Memory Encoding

Memory Encoding is the process of manipulating the input from surrounding and convert into the form the brain can process it. Usually a human takes the information in the form of image or sound. The information having forms including visual (picture), acoustic (sound) and semantic (meaning) is processed and encoded. The visual, acoustic and semantic forms are encoded in Short Term Memory and Long Term Memory.

3.2.5.2 Memory Storage

Memory storage deals with the storage of the encoded information in a defined format by processing the information in the context that what type of information and how to store in an efficient structured way so that it can be retrieved efficiently. The storage structure of the words in brain has some defined structure. The contextual relationship between the stored words constitutes the language in the brain. The spellings of the word along with the pronunciation information is stored for each word. The meaning of a word is stored at semantic level. The sequence of words are combined to form the sentences using contextual and grammatical information (also called syntactic information). To form the correct sentences, the words must be stored in a structured format so that word form, semantic and syntactic information can be used to recognize and read the sentences.

This structure is different from the ordinary dictionary which is organized according to the alphabetic order. In addition, the size of the dictionary is static which cannot be changed unless a new edition of the dictionary is published. In contrast, human memory has stored words in structured form and new words are learned on the basis of day by day experience.

3.2.5.2.1 Mental Lexicon The complete information of words which includes word form, semantic and syntactic information is maintained in mental lexicon [57]. It contains orthographic (vision-based) and phonological (sound-based) forms of words. The mental lexicon is organized in a highly efficient manner so that a

person can recognize and speak three words per second. Its size is not fixed because unused or un-rehearsed words can be forgotten and new words can be learned with the passage of time. It can access frequently used words more quickly. During recognition of the words, mental lexicon accesses the words using the neighboring effect (similarity) of sound and shape.

The mental lexicon is organized in such a way that the information specific networks for the words at lexeme, lemma and conceptual levels are defined. Levelt [57] defines the structure of the network which has word form, semantic and syntactic information of words. This architecture is defined for the storage of speech but it can also work for storage of visual object as can be seen in Figure 3.3. The architecture has three levels; (1) Lexeme level, (2) Lemma level and (3) Conceptual level. At Lexeme level the word form which has spelling and pronunciation of a word are defined. The semantic specification and grammatical properties are defined at Lemma level. The semantic specification defines the conceptual conditions (such as sense) of a word which is used to select appropriate word given the context. At Conceptual level, the semantic knowledge of the word is defined which ensures syntactic information. The conceptual conditions (senses) of words and semantic knowledge are used to select appropriate word.

3.2.5.3 Retrieval

Memory retrieval is the process of getting stored information. Failure in the retrieval is due to the fact that human did not remember the information. The efficiency in the retrieval entirely depends on way the information is stored. In STM the information is stored sequentially therefore retrieval of the information is also carried out sequentially. In LTM the information is stored and retrieved as association between the events or information.

3.3 Reading

Reading is the process which retrieves the stored information from brain. This information is used to recognize the sequence of strokes. The words are formed using recognized sequence of strokes and characters, phonological information, lexical processing and semantic processing [57] [91] [84]. Therefore for reading,

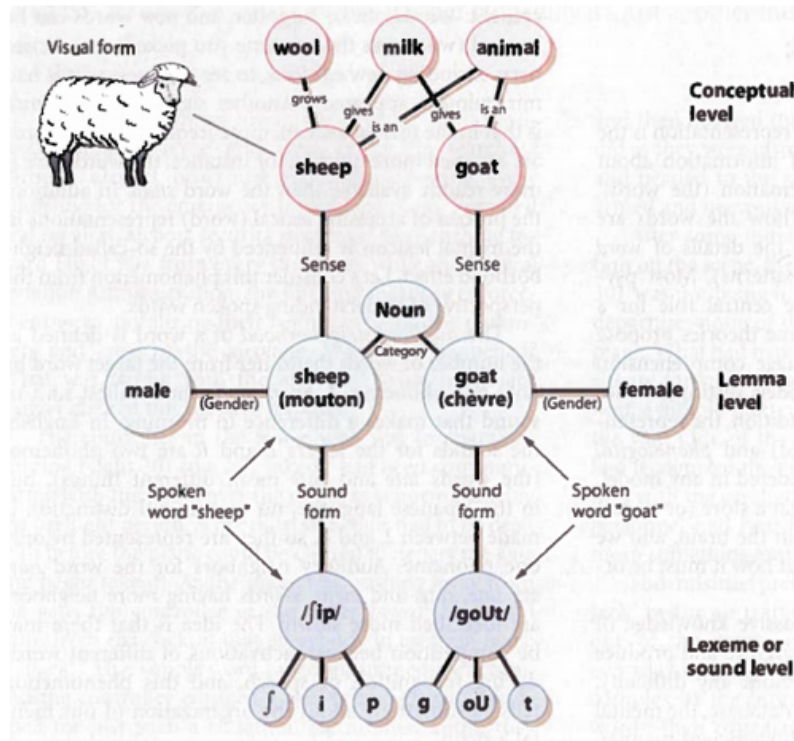


FIGURE 3.3: Lexicon Network [108] According to Willem Levelt [57]

the basic processing unit is visual pattern of letters. These visual patterns which have corresponding sounds in a language vary across different writing systems. There are three main types of visual patterns in writing [105].

- 1. Alphabetic:** In this type, each symbol of letter corresponds to phoneme. Based on degree of correspondence of letter with sound, there are two categories of orthography; (1) shallow orthography which includes the languages which have close correspondence of letters with sounds e.g. Spanish language, and (2) deep orthography which includes the languages which sometimes do not have correspondence of letters with sounds e.g. English language.
- 2. Syllabic:** In this type, each shape of symbol corresponds to a syllable. The example of this type is Japanese language which has about 100 unique syllables.
- 3. Logographic:** In this system, a unique shape is corresponded to a word or morpheme. The example of this system is Chinese language which is although not pure logographic. In Chinese language each symbol corresponds to the morphemes or phonemes. Actually Chinese is tonal language in which the meaning of a word depends on the variation of rise and fall of the pitch of vowel.

3.4 Cognitive Models of Reading

The main focus of reader of any writing style is to analyze the features of a shape of symbol (letter, syllable or word) such as strokes (horizontal, vertical lines etc.) and to try to recognize the letter. In this section two main models are discussed which are proposed for reading the written text.

3.4.1 Selfridge Bottom up Model

In this model, Selfridge [91] defines four demons(stages) for the recognition of written letter. This is unidirectional model which works from bottom to top levels to recognize the words.

1. **Image Demon:** At this stage the input image is stored as an iconic memory.
2. **Feature Demon:** At this stage, the features such as lines and curves etc. are extracted along with the locations from iconic memory.
3. **Cognitive Demon:** At this stage, the extracted features are processed to recognize possible letters.
4. **Decision Demon:** The Decision demon finally recognizes the best letter given possible options of cognitive demons.

3.4.2 Rumelhart and McClelland Bidirectional Model

Rumelhart and McClelland [84] present another perceptual model for the recognition of visual and spoken word. It is bidirectional model which means the information of higher levels such as word level, can be used to improve the recognition of a letter. This model has three levels as can be seen in Figure 3.4. These are

- **Feature Level:** At this level, features of the input shape of letter are extracted. The visual features are extracted from iconic memory and acoustic features are extracted from echoic memory.
- **Letter/phoneme Level:** Based on the extracted visual features, respective letter shape is recognized. As can be seen in Figure 3.4, for the recognition of the letter shape, the phonological information can also be manipulated

for perceptual processing of brain. In addition, based on the output at this level, cognitive processing can go backward to fine tune the features for the better recognition of letter.

- **Word Level:** At this level, the recognized letters are joined to form the words. Same as second level, feedback from letter/phoneme level is involved for the recognition of word for best match.

The main difference of Selfridge model [91] and Rumelhart and McClelland [84] is that Rumelhart and McClelland model allows usage of higher level information to improve the performance of lower level where as Selfridge model only allows bottom up flow of information.

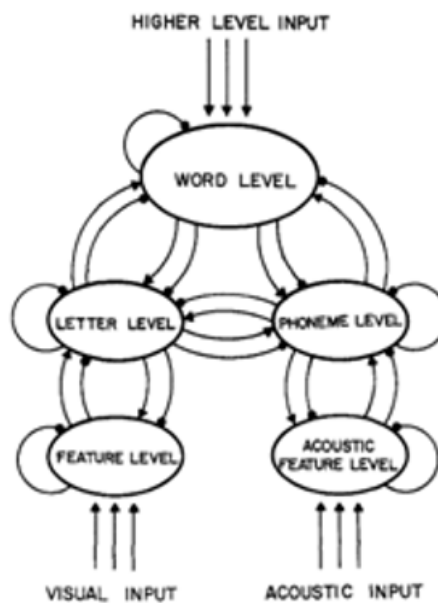


FIGURE 3.4: Layers of Visual Word Recognition [84]

3.5 Letter as Basic Unit of Reading

A lot of research theories exist in the field of cognitive focusing on identifying the factors which influence the reading of the given text. During reading, usually words are recognized by separately identifying the individual letters of the words

[79] [50]. In addition, research also argues that letters and words are recognized in parallel by doing contextual processing of pseudo words and giving feedback to the letter detectors for better recognition of words [28] [64]. The letter identification is core step for reading [28]. In addition, number of letters in words also influence the reading efficiency, as increase in number of letters reduces the reading efficiency [79]. It has been an important discussion by different researchers for many decades related to identifying the features which are used to identify the letters. Usually, sequence of visual features of a letter are transformed into the position coded identities which human brain uses to recognize the letter. To understand the alphabetic orthography of a language, letter based approaches have been extensively used for last decades. This is because, there are limited number and shapes of letters which can easily be investigated as compared to millions of shapes of words [29].

3.5.1 Methods of Letter Recognition for Reading

The process of letter recognition to read sequence of words is based on two cognitive models of reading [91] [64], discussed in previous sections. According to Selfridge [91] human mind envisions the input object as the sequences of features at Feature Demon, then these sequence of features are used to recognize one or more objects at Cognitive Demon. At Decision Demon, a single object is recognized by best matching the context of these recognized list of objects. In another well known theory of reading [64], a bidirectional model, three main levels are discussed. This model allows information of upper level e.g. Word level can be reused lower level e.g. Letter level to improve the identification.

Different methods are explored by the researcher for letter identification to read the text. These are broadly categorized into three types; (1) Template matching based letter identification, (2) Confusion matrix based letter identification and (3) Feature-based letter identification.

3.5.1.1 Template Matching-based Letter Identification

In template matching approach discussed in [55], many examples (templates) of letter shape are stored in memory, the target letter is recognized by best matching the stored template with the input one. The new template is learned when recognition of the target is not correct. This theory has clear drawback of size normalization of the target shape and all stored templates. In addition, for the better recognition accuracy, a significant number of examples and variation of templates are required to learn in memory [76]. However, up till now, the matching criteria in terms of either color intensities or feature intensities has not been discussed with consensus.

3.5.1.2 Confusion Matrix-based Letter Identification

Majority of the reported work to investigate the feature set with major focus on Roman letters is based on detailed analysis of confusion matrices. To do this, usually a stimulus is generated with different settings of masking, luminance and presentation duration [51] [24].

The reaction time and errors are stored during presentation of stimulus. For example the participant response of detection of letter 'E' when letter 'F' is presented may cause confusion between 'E' and 'F'. This is because these letters share some same features [29]. Usually such errors are used to find the confusion between the letters. In some experimental setup, it has been observed that person takes more time to disambiguate two similar objects. The main drawback of the confusion based approaches to find the feature set is the way to mask the letter (e.g. 'F' to generate the confusing letter) which may influence the nature of the confusion [29].

3.5.1.3 Feature-based Letter Identification

To resolve the issues in the confusion matrix based letter identification, another approach is used in which contrast thresholds are used to generate the stimuli instead of doing degradation of the letters. Contrast threshold means the some portion of the letter is faded using the variation of the spatial frequency using the Bubble's technique [23]. This is because, human nervous system is sensitive to the

variation of spatial frequencies [15] i.e. change in luminance.

A letter is composed of sequence of features which human brain uses to recognize. The important question for the feature based recognition is that what are main set of features which are used for the recognition or identification of letters. Majority of the reported work focused on the English letters. Actually, isolated letter identification is simple to understand how objects are recognized which is done by identification of visual features [26]. The research on letter perception reveals that letters are recognized using the identification and manipulation of feature set [45]. In later research study, it is observed that these features are organized in hierarchical layers which is processed by letter detectors of reading models [29]. Different studies exist in literature which extract important features which are used for the identification of letters. Fiset et al. [23] analyze different visual features for letter identification. By using Bubble's technique, the samples/stimulus have been generated at five different spatial scale (by using the concept that human visual perception responds against different spatial frequencies). By analyzing experimental data, features are extracted automatically which are useful for the identification of lower and upper case letters having Arial font style. Different features are identified such as curves, verticals, slants, intersections, horizontals and terminations. Among them, terminations and intersections are high prioritized for the identification of English letters. For example the letter 'W' is described as having two termination lines, one in upper left point and second on the upper right point.

3.6 Factors Impacting Reading or Retrieval

3.6.1 Letter Identification

Letter is the basic identification unit of visual word recognition and eventually for reading. Recognized letters help to recognize the complete word. According to Selfridge [91], and Rumelhart and McClelland [64], a letter can be defined or identified by using the features set which are organized as hierarchical structure. These features are strokes. Only strokes do not resolve the identification problem.

The spatial information i.e. sequence of strokes are also stored in human memory. In addition, in alphabet set, different letters have similar shapes which also add complexity for the identification of letters.

3.6.2 Phoneme Information

The usage of phoneme information during reading also helps the recognition. Hue and Erickson [36] further verify this theory of using phoneme level information during reading. They investigated that characters which have known visual and phoneme information are recognized better as compared to the characters which have only visual information.

3.6.3 Formatting

Now a days, the computer generated content is normally used for the learning and reading. Majority of the content used for learning or teaching published in either printed or online is normally generated through computerized processing. Different studies also investigate the readability on paper versus computer screen. Mills and Weldon [68] present the survey of research which investigates the reading performance of text on paper and computer screen. The factors such as foreground and background colors, formatting and letter features are highlighted which influence the reading speed and accuracy of text displayed on computer. The online content generation and dissemination is becoming important to facilitate the language readers to access and read the content online. The text formatting and style preferences also affect the reading ability. Generally in literature font style, size and alias versus dot matrix preferences are observed during reading.

Paterson and Tinker [77] compare ten font styles in context of readability. They concluded that text printed with Scotch Roman type has high reading speed as compared to other font styles, where as the reading accuracy is same for all selected font styles. Hence, based on results Scotch Roman font style performs well for both reading speed and reading accuracy. Later Sanocki [89] argued that font style and size have significant impact on accuracy. To investigate the effect of change in size and style to read text, three different experiments are conducted by changing the style, size, and both style and size. The experimental results indicate

that variation in font size and style have significant impact on accuracy. Tullis et al. [97] investigate the reading performance of different fonts of serif and sans serif. The text is generated using different font sizes range from 6.0 to 9.75 points. The reading accuracy and speed are recorded after each experiment. The results show that Arial and MS sans serif at 9.75 font size perform well for reading. Bernard et al. [14] investigate the formatting preferences of children during reading the computer generated text. The selected text having eight passages is generated using Times New Roman and Courier New (which belong to serif typeface), and Arial and Comic Sans MS (which belong to sans serif typeface) at 12 and 14 font sizes. The experiments are conducted on 27 children having age ranging from 9 to 11 years. The results show that font styles of sans serif typeface with 14 point size perform best among all in context of ease of reading and reading speed. In addition, the children also prefer sans serif typeface to read school-related material. Pelli et al. [25] investigate the impact of font style, size, duration, eccentricity, age and reading experience for identification of letters of different languages including English, Arabic, Armenian, Chinese, Devanagari and Hebrew. The results show that efficiency of letter identification is dependent on size and font style. The complex font style and/or smaller font size reduces the accuracy of letter identification. Bernard et al. [14] figure out that reading performance is influenced by three factors including font style, size and format i.e. dot-matrix and anti-aliased. The results indicate that Arial font style at 12 font size printed as dot-matrix has significant positive impact on readability as compare to other font style and size settings. The above mention research is related to the computer screen displayed text. Due to the complexity of the Arabic orthography, the perceptual load to recognize the Arabic is slow [81]. The speaker having Arabic as first language processes Arabic orthography slowly. Darroch et al. [21] present the experimental study which evaluates the font size impact on readability on handheld devices. The passages are prepared by selecting Microsoft Sans Serif with font sizes range from 2 to 16 with a difference of two. Based on reading speed and accuracy results, the font size ranges from 8 to 12 point improves the readability on mobiles.

3.6.4 Reading Ability and Development Age

Reading ability and development age affect the letter specificity during reading. The letter specific processing which includes the identification and manipulation of letters and non-letters in the text is influenced by reading skill. Burgund et al. [16] examine perceptual processing of Roman letters in subjects (participants) of different age groups from 6 to 19 years old. These letters and non-letters are presented at 36 point size. The results reveal that letter specific processing is largely influenced by increase in reading skill. Later, Burgund and Abernathy [20] also figure out that the reading ability is influenced by the development age (duration from child to adult). This is because reading ability is matured with the ability to do contextual processing while recognizing words. Adults do more visual information processing as compared to the children. They conducted the experiments on adults and children having similar less-than-expert fourth grade reading level. The experiments are generated by using stimulus of letter and non-letters. The error rate and response time indicate that adults have greater letter specificity during reading as compared to the children. Therefore, adults pay more attention on visual forms of letters as compared to the children and reading skill does not affect letter specificity.

3.6.5 Words Recognition

As discussed above, visual word recognition depends on the correct recognition of letters [91][84]. The contextual processing of words also improves the letter recognition accuracy. Different studies on Latin script evaluate the effect of reading at letter level and word level. In addition, the upper and lower part of Latin words are also investigated to check the impact on reading accuracy. Perea et al. [59] investigate impact of upper part of words for letter and word recognition. The words and pseudowords are used to generate the stimulus of primed upper and lower parts. The results conclude that upper part of the words can be used to recognize complete words. This is because only upper parts of words have higher reading accuracy as compared to the upper parts of pseudowords.

3.6.6 Text on Central Fixation Point

Researchers are also working to find out efficiency of reading and letter identification by fixation the target/stimulus at different positions [90] [107]. Nazir et al. [1] present an evidence that text appeared at positions at which our eyes are trained to read, have better reading efficiency. Stevens and Grainger [96] also predict the effect of reading at different fixations with different combinations of string lengths. They figure out that words recognition accuracy when appears at location of eye fixation is improved which depends on the visibility length of constituent letters. Tydgat and Grainger [43] also evaluate the performance of identification of letters, digits and symbols using serial positions. Based on the results, it is observed that performance is affected by receptive fields (which map iconic input from retina to the neurons), shape and size.

3.7 Conclusion

In this chapter, the background study related to the cognitive development involved in reading the text is discussed. According to the cognitive models of reading basic unit of word recognition is letter which is recognized by using visual features. The main focus of the research studies has been on identifying the factors which influence the reading of the given text. During reading, usually words are recognized by identifying the individual letters. The letters are identified by recognizing the features i.e. sequence of strokes. Later, words are recognized by using phonological information, contextual processing. A lot of research theories exist in the field of cognitive to identify the features which are used to identify the letter especially for Latin script. Different factors which influence the reading efficiency are also discussed. The majority of the existing research focuses on the English letters. However there is a gap to identify the set of visual features which are used by human brain to recognize the Urdu Nastalique text. In next chapter, the complete perceptual experiment is designed and conducted to extract the feature set of Urdu characters written in Nastalique writing style.

Chapter 4

Automatic Text Recognition

Optical Character Recognition (OCR) is a system which automatically processes the document images to recognize the text. The tremendous advancement of Information and Communication Technologies (ICT) in the local content availability highlights the significant need of porting published local content online in the form of editable text so that it can be searched and retrieved by the local users. The digitization of the published content is important to make it available online. To convert the published material into digital content efficiently, the development of (OCR) system is important.

The OCR system has three modules; (1) Preprocessing, (2) Classification and Recognition, and (3) Post-processing. Preprocessing module deals with the processing of an input image to improve its quality and to segment image into different areas using different sub modules such as binarization, layout analysis and text area segmentation etc. The relevant information from these areas is extracted which is used in classification and recognition, and post-processing modules. Classification and Recognition module has two phases. The first phase called training phase, deals with the classification of character/ligature shapes into different classes based on the shape similarity. The features of each class are extracted and used as input to a classifier for training. In the recognition phase, the features of input shape are computed and respective characters/ligatures are recognized using these features from trained classifier. The post-processing phase deals with

the formation of words and sentences using recognized characters/ligatures sequences. This module has word segmentation, spell checker and POS tagger etc. sub-modules, and outputs the correct sequence of words to form sentences.

The focus of the thesis is on the classification and recognition module. The state-of-the-art recognition techniques for Urdu like cursive script focus on the recognition of text by using two approaches; (1)ligature-based and (2) character-based recognition i.e. segmentation-based recognition. The details of techniques are discussed in sub-sequence sections.

4.1 Ligature-based Classification and Recognition

Generally ligature-based classification techniques for the cursive writing style such as Naskh and Nastalique, process text line as sequence of ligatures. That means lines have sequence of ligatures i.e. $L = l_1, l_2, l_3, \dots, l_n$, where l_i denotes i^{th} ligature in the line. The ligature-based classification and recognition deals with extraction of features from the ligature. These extracted features along with labels of ligature types are used to train the system using different classification techniques. Javed et al. [49] extract features through sliding window over ligature images and use HMMs for recognition. The reported accuracy of the system is 92% on synthesized data at 36 font size. Lehal and Rana[56] recognize Nastalique ligatures using Zoning, DCTs, Density, and Gabor features. The primary and secondary ligature strokes are separately recognized using SVM, KNN and HMM classifiers for the comparative analysis. The system recognizes 2190 types of primary strokes and 17 types of secondary strokes. It gives 98.01% accuracy for the primary stroke recognition, tested on 4380 images and 99.91% for the secondary stroke, tested on 1700 images, by using DCTs with SVM for best results. The reported computation time for recognition of 400 primary components is 77 seconds for SVM and 26 seconds for KNN. Sabbour and Shafait [85] report a dataset called UPTI having 10,000 synthetically generated Urdu text lines images written in Nastalique writing style. They develop a ligature recognition system using shape context based features of contours of main body and diacritics. The reported accuracy of the system is

91% for Urdu and 86% for Arabic, tested on UPTI test data for Urdu written in Nastalique writing style, and Arabic test dataset written in Naskh writing style respectively. Ahmad et al. [5] use a Gated Bidirectional Long-Short Term Memory (GBLSTM) networks to recognize the text line by recognizing the sequence of ligatures. The pixel values computed from ligature images along with labels of ligatures are fed to the GBLSTM for training and recognition. The GBLSTM based ligature recognition system has 96.71% accuracy tested on 29,935 ligatures of 1600 text lines of UPTI dataset. Khan et al. [52] report performance of different clustering techniques for Urdu ligatures. The K-nearest neighbor outperforms with 90% clustering accuracy. The ligature based OCR system for Urdu Nastalique writing style is developed [8] by modifying the state of the art multilingual open source recognition system [94]. The challenges of using Tesseract for the recognition of Urdu text in Nastalique writing style is analyzed and Tesseract is modified to get better recognition accuracy. The pre-processing and chopping functionality of Tesseract is disabled. Normally space is not used to define word boundary in Urdu, therefore dictionary functionality is disabled to improve the recognition accuracy and efficiency of Nastalique text recognition. A total of 1475 main body types (RASM classes) are used to train and test the system. The training data has 10 samples of each RASM classes and testing data contains 15 samples of each RAMS class. A total of 14,750 main body images are trained for 14 and 16 font sizes separately. The modified Tesseract gives 97.87 % accuracy for 14 font size and 97.71% accuracy for 16 font size tested on 22,125 instances of each font size.

4.2 Character-based Classification and Recognition

The character-based classification techniques for the cursive writing style, divide text line into sequence of character i.e. $L = c_1, c_2, c_3, \dots, c_n$, where c_i denotes i^{th} character in the line. The character-based recognition techniques extract characters' features from the images. The features along with the character transcription are used to train the classification system. There are two categories of character-based recognition techniques; (1) Explicit segmentation-based recognition and (2)

Implicit segmentation-based recognition.

4.2.1 Explicit Segmentation-based Recognition

Explicit segmentation-based recognition techniques segment document image into constituent characters or smaller primitives. The techniques developed for various scripts utilize the properties of the script by using image processing techniques. The features from the segments are extracted, and along with characters labeling are used to train the system. The letters in Devanagari script are segmented by removing its top line, but the computation of the top line also called Headline for handwritten text is not a trivial task. Shaw et al. [92] use morphological operator along with horizontal window of size 1x27 to detect the head line of Devanagari script. After segmentation of words, the Hidden Markov Models (HMMs) are used for recognition of 118 letters. The system has 81.63% character recognition accuracy and 84.31% word recognition accuracy, using 22,500 images for training and 17,200 images for testing data. For Naskh writing style, a segmentation approach for handwritten words of Arabic language is reported in [58]. The main bodies (of ligatures) are separated and then segmentation points are computed using horizontal and vertical gradients using baseline information. They address over-segmentation by ignoring the segmentation points in loops, at edges, and using characteristics of letter shapes. Testing on 200 handwritten Arabic images from IFN/ENIT database gives 92.3% segmentation accuracy. Cheung et al.[18] use the projection profile and convex dominant points (CDPs) for the segmentation of Arabic words into fragments. These fragments of the Arabic words are processed to extract smoothed chain codes for recognition. They report recognition accuracy of 90% on images of books. Mehran et al. [67] use vertical projection of the line image, first derivative of upper contour, and distance between pen tip and baseline, for the segmentation of main bodies of ligatures into characters. These junction points are used to train a Neural Networks (NNs). Then segmentation points are marked using the trained NNs. The reported accuracy of the segmentation is 98.7%. Following segmentation, normalized values of structural and statistical features are used for the recognition using Support Vector Machines (SVMs). Total of 40,000 samples of main bodies containing 175,632 graphemes

are used to train the system, with overall 98.3% character level and 90.17% word level accuracy. Very limited work has been done on explicit segmentation-based recognition for the Nastalique writing style. Safabakhsh and Abidi [86] present a segmentation-based technique for the recognition of handwritten words. They first remove ascenders and descenders from ligatures to reduce false traversal of right to left order. Then two different segmentation methods are employed and tested on 350 main bodies. First approach uses the singularities and regularities to segment the ligatures with 77.62% segmentation accuracy. The second uses the local minima, overlapping of stroke information and joining position with previous character to extract the candidate segmentation points, giving 95.68% segmentation accuracy. The Fourier descriptor, structural and discrete features are used for recognition using continuous-density variable-duration HMMs, giving 96.8% recognition accuracy. Javed and Hussain [47] thin the ligature images by applying thinning algorithm, and then segment them at branch points. These segmented thinned strokes are windowed and Discrete Cosine Transform (DCT) features are computed for each window. The features extracted from the sequence of windows for each segment are used to train the HMMs for recognition, and sequenced to formulate the ligature. The test data has synthesized images of 1,692 high frequency ligatures (printed at 36 font size) formed by the subset of six of the 21 character classes (see [100]). The system has 92.73% RASM (main body) recognition accuracy. Muaz[6] extended this segmentation-based approach to include all 21 character classes in Urdu. The system is tested on 2,494 ligatures synthesized at 36 font size, giving an accuracy of 92.19%.

4.2.2 Implicit Segmentation-based Recognition

The implicit segmentation-based recognition techniques use the concept of sequence learning using state-of-the-art sequence learning techniques. The text images with corresponding characters/ligatures/words labels are fed to sequence learning approaches including HMMs [3][41][69][83][87], Recurrent Neural Networks (RNNs) and variants of RNNs [31] [30]. These sequence learning techniques are algorithmically strong enough to learn long context by extracting similar shape

patterns and assigning labels to these patterns accordingly. The extensive research is available on use of HMMs and RNNs with variants to develop robust character recognition systems for different languages such as Arabic, Chinese, Devanagari and Urdu etc., by tweaking the feature extraction approach, devising efficient labeling and tweaking the parameters of the learning system. Sankaran and Jawahar [88] use Bidirectional Long-Short Term Memory (BLSTM) network for the segmentation and recognition of Devanagari text. They extract contextual information including top foreground pixel distance from the baseline, bottom foreground pixel distance from the baseline, ink-background transitions, number of black pixels and span of foreground pixels using sliding window. The system is tested on 67,000 words and has 94.35% and 84.87% character recognition accuracy for clean and degraded document images respectively. The corresponding word recognition accuracy is 91.38% and 77.85%. Al-Muhtaseb et al. [2] use vertical sliding window having three pixels width and height equal to the text line height. Sixteen features are computed by dividing the window into different sizes and computing the number of black pixels from each sub-window. Character transcription of a ligature is used for training of HMMs based on these features. A total of 2,500 text lines are used for training and 266 for testing, with a total of 46,062 words and 224,109 characters. Eight different fonts have been applied on these text lines and image corpora of these are synthesized separately. Text line images are normalized to have height of 80 pixels. The testing gives 99.90%, 99.68%, 99.34%, 98.78%, 98.09%, 99.70%, 98.83% and 97.86% accuracy for Arial, Tahome, Akhbar, Thuluth, Naskh, Simplified Arial, Traditional Arabic and Andalus fonts respectively. Al-Khateeb et al. [10] evaluate different features for the recognition of Arabic handwritten words. The features such as mean pixel value of overlapped blocks, first 100 DCT coefficients, first five DCT coefficients extracted from non-overlapping frames, Hu moments and wavelet transforms are computed from normalized word image. The testing is done on IFN/ENIT database with 32,492 Arabic words. The DCT features have 80.75% recognition accuracy where as overlapping blocks and moment invariant features have 77.75% and 75.75% recognition accuracy using NNs. Al-Khateed et al.[11] use intensity features for

the automatic segmentation and alignment of Arabic language handwritten words using HMMs. The images are normalized to have fixed height. They use a sliding window with three pixels width and height equal to line height. Window is further divided into 15 sub-windows vertically to extract 30 intensity features. These features are used to train the HMMs. The recognized results are ranked using structural features. They use IFN/ENIT Arabic handwritten database for training and testing. The system has 82.32% accuracy before re-ranking and 83.55% with re-ranking on v2.0p1e dataset. Khorsheed [53] extracts three different intensity features including simple intensity, horizontal intensity and vertical intensity from each sub-divided window and use HMMs for classification. The data set having 1,500 line images is generated containing 116,743 words and 596,931 letters. These images have been synthesized using different computer generated Arabic fonts. All lines have been normalized to 60 pixels height. Using training data of 1,500 and testing data of 1,000 line images, authors report 87.6%, 86.0%, 88.0%, 89.5%, 92.1% and 92.4% recognition accuracy for Thuluth, Naskh, Simplified Arabic, Traditional Arabic, Tahoma and Andalus fonts respectively. Ul-Hasan et al. [98] present the recognition system of printed Urdu Nastalique using BLSTM networks. The synthesized Urdu Nastalique data set is used to develop and test the approach. This data set consists of 10,063 synthetically generated text lines. Each text line image has been normalized to a fixed height. A 30x1 window is horizontally moved along the text line image and pixels values are used as the feature set. The character transcription is used for the segmentation-based recognition. The Recurrent Neural Networks (RNNs) are used for classification and recognition of characters. The transcription of the text line used for classification has maximum of four unique labels of shapes of a character based on the positional information in a ligature. Testing on synthesized data of 2,003 text line images gives 94.85% character recognition accuracy for trained data model. Another approach for the implicit character-based recognition is presented for Nastalique writing style of Urdu is also presented [41]. The text line based horizontal sliding window used to extract the features performs well for the Naskh writing style. However due to the complexities of the Nastalique writing style, specifically character and ligature

overlapping, such techniques cannot be applied. To handle this issue, a new traversal technique is presented to traverse along the contour of the ligature stroke. After Binarization of the image, the thinning is applied to have the single pixel contour. The start point for the traversal is computed which is actually last point of the ligature image. The local window is moved along the contour by placing center of the window in respective thinned pixel of the main body. The consistent stroke traversal is also defined to ensure that all the character strokes are explored in the same reverse order as these are written. During traversal, the DCTs as features are computed and stored as feature vector. The characters contextual shaping is analyzed in detail and a total of 250 unique characters shapes are finalized based on the context. The HMMs as classifier is used for training and recognition. The different stages including, thinning, start point detection, consistent traversal of strokes and feature vector computation are developed and fine tuned. The overall recognition accuracy is 97.11% tested on 79,093 instances of 5,249 main body classes. The proposed system is also tested on document images of different books with 87.44% main body recognition accuracy.

The statistical features extracted from sliding windows on normalized line height images are used to recognize Urdu characters sequence using Multi-dimensional Long Short-Term Memory (MLSTM) network [74]. The character recognition accuracy of the system is 94.97% tested on 1600 text lines images of UPTI dataset. The multi-dimensional long short term memory recurrent neural network (MDLSTM RNN) with connectionist temporal classification (CTC) as output layer gives 96.40% Urdu character recognition accuracy tested on 1600 text line images of UPTI [72]. Another approach uses MDLSTM RNN with a matured output layer for sequence labeling to improve the recognition results. The system has 98% character recognition accuracy tested on same UPTI dataset [73]. The hand crafted features are extracted using Convolutional Neural Networks (CNN) which are fed to MDLSTM for Urdu characters training and recognition. The system has 98.12% character recognition accuracy tested on 1,600 text lines images of UPTI dataset.

4.3 Conclusion

In this chapter, state of the art techniques to develop Urdu OCR are discussed. OCR has three modules; (1) Preprocessing, (2) Classification and Recognition, and (3) Post-processing. Over past decade, recognition of Urdu document images has received a lot of attention mainly focusing on the classification and recognition module of OCR. Different HMMs based and deep learning based techniques are used to develop state-of-the-art recognition systems. These state-of-the-art recognition techniques are divided into two approaches; (1) ligature-based recognition and (2) character-based recognition. The ligature-based recognition techniques extract features from ligature image. The features along with the sequence of ligatures' transcriptions are used to train the system using different classification techniques. The character-based recognition techniques extract characters' features from the images. The features along with the character transcriptions are used to train the classification system. There are two categories of character-based recognition techniques; (1) Explicit segmentation-based recognition and (2) Implicit segmentation-based recognition. Explicit segmentation-based recognition techniques segment the document image into characters by using image processing techniques and writing style characteristics. The features from the segments are extracted, and along with characters labeling are used to train the system. The implicit segmentation-based recognition techniques use the concept of sequence learning using state-of-the-art sequence learning techniques. The text images with corresponding characters/ligatures/words labels are fed to sequence learning approaches including HMMs, Recurrent Neural Networks (RNNs) and variants of RNNs. These sequence learning techniques are algorithmically strong enough to learn long context by extracting similar shape patterns and assigning labels to these patterns accordingly.

Chapter 5

Perceptual Experimentation of Urdu Character Identification

Arabic orthography is complex as compared to the Latin orthography due to the characteristics already discussed in Chapter 2. Each character of Arabic script has varying number and shapes of strokes. Visual perceptual analysis for the character recognition plays important role to finalize the font style and size [14] [25] [21] which are used to improve the reading ability. Different studies are also reported to define features used for character recognition by taking Latin character set as an example. Arabic has complex orthography which results in slow perceptual processing to recognize the Arabic text [81].

Text written in Nastalique writing style is also complex to process as compared to the Arabic due to characteristics such as diagonality, different contextual shaping of a character and complex rules to place diacritics of a character in ligature etc. (discussed in Chapter 2). Human can intelligently process and disambiguate the complex and confusing shapes and can recognize the text correctly. According to the experimental study reported by Fiset et al. [23], the complete letter is recognized by recognizing the important key features of letters. More specifically, human brain mainly focuses on some important features to recognize the characters, because 32% of upper case English letters and 24% of lowercase English letters are used to recognize the text [23]. In the same way, some core cognitive features

would play significant role for letter identification of Urdu written in Nastalique writing style. In this chapter, we investigate the effect of strokes used as cognitive features for the identification of Urdu character set written in Nastalique writing style. The summary of the experiment is given below.

Each character is segmented into different strokes using the calligraphic knowledge which a native Urdu writer uses to learn Urdu writing. The strokes and their sequence are eventually verified by expert calligraphers. Based on the number of strokes, characters are classified into four categories i.e. single stroke characters, two strokes characters, three strokes characters and four strokes characters. Experimental visual masks of the characters are prepared with different configuration of stroke sequences (single stroke, two strokes in sequence of writing, three strokes in sequence and four strokes in sequence). Identification accuracy of sequence of strokes used to identify respective character is analyzed to investigate and characterize the character strokes into three categories. The details of the experiment are given in sub sequent sections.

5.1 Related Work

The research studies in the field of cognitive theory focus on perceptual analysis of letters especially for Latin script. The main focus of the research has been on identifying the factors which influence the reading of the given text. During reading, usually words are recognized by identifying the individual letters [79] [50]. In addition, letters and words are recognized in parallel by doing contextual processing of pseudo words and giving feedback to the letter detectors for better recognition of words [28] [64]. The character identification is core step for reading [28]. In addition, number of letters in words also influence the reading efficiency, as increased number of letters reduces the reading efficiency [79]. It has been an important discussion by different researchers for many decades to identify the features which are used to define letters. Usually, sequence of visual features of a letter are transformed into the position coded identities which human brain uses to recognize [44]. To understand the alphabetic orthography of a language, letter based approaches have been extensively used. This is because, there are limited number

and shapes of letters which can easily be investigated as compared to millions of words. Majority of the reported work focused on the English letters. Actually, isolated letter identification is a simple way to understand how objects (letters) are recognized through the identification of visual features [26]. The research on letter perception reveals that letters are recognized using the identification and manipulation of feature set [45]. Later, research stated that these features are organized in hierarchical layers which are processed by letter detectors [29]. The important question for the feature based recognition is that what are main features which are used for the recognition of a letter. The investigation of feature set with major focus on Latin letters is based on detailed analysis of confusion matrices. For this, usually a stimuli is generated with different settings of masking, luminance and presentation duration [51] [24]. The reaction time and error rates are stored during presentation of stimulus. Usually error rates are used to find the confusion between the letters. Another recent approach to find the feature set applies contrast thresholds on the letters to generate the stimuli instead of doing degradation of the letters with different spatial frequencies using the Bubble's technique [23] [26]. This is because, human nervous system is sensitive to the variation of spatial frequencies [15] i.e. change in luminance.

By using Bubble's technique, the samples/stimulus are generated at five different spatial scale [23]. The xperimental data is analyzed to identify the features which are useful for the identification of lower and upper case Arial letters. Different features such as curves, verticals, slants, intersections, horizontals and terminations are identified. Among them, terminations and intersections are high prioritized for the identification of English letters. For example the letter 'W' is described as having two termination lines, one in upper left point and second on the upper right point [29].

5.2 Methodology

Different research studies focus on identification of feature set of Latin character set by doing detailed analysis of confusion matrices and response of participants against the stimuli. Up till now, no research has been carried out to find the

feature set of Arabic script characters used for the recognition and reading of Arabic text. The present study provides perceptual analysis of stroke of Urdu characters to analyze the feature set used for character recognition. To handle the experiment size, 21 character classes of complete Urdu character set are selected. These 21 character classes are categorized based on the shape of RASM of Urdu characters, see Figure 2.6. The strokes of each character are extracted by using the calligraphy rules of Urdu Nastalique writing. These strokes are verified from the different books of Nastalique calligraphy, some samples of characters from calligraphy books [93][80] are also given in Appendix A. The selected characters have varying number of strokes. The characters are classified into four groups based on the number of strokes. These are given in Figure 5.1, Figure 5.2, Figure 5.3 and Figure 5.4. The sequence of strokes used to write the character are also highlighted with red, green, blue and black colors for first, second, third and fourth stroke respectively.



FIGURE 5.1: Single Stroke Character

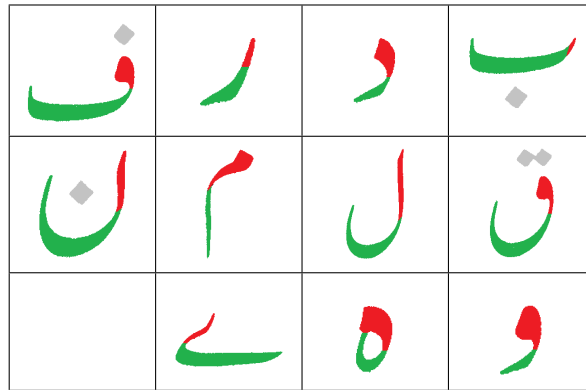


FIGURE 5.2: Two Strokes Characters, First and Second Strokes in Sequence Highlighted with Red and Green Colors, Respectively

Normally, the human brain processes features of a character image i.e. sequence of strokes and then tries to recognize the characters using the contextual information so that the best sequence of words can be recognized. In the same way, in Arabic

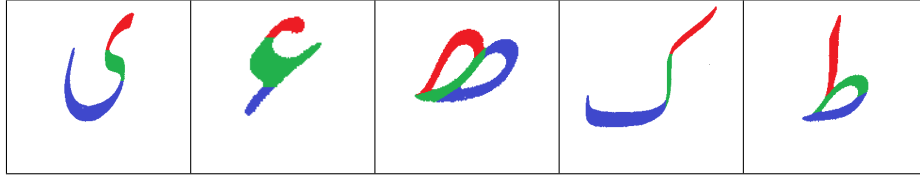


FIGURE 5.3: Three Strokes Characters, First, Second and Third Strokes in Sequence Highlighted with Red, Green and Blue Colors, Respectively

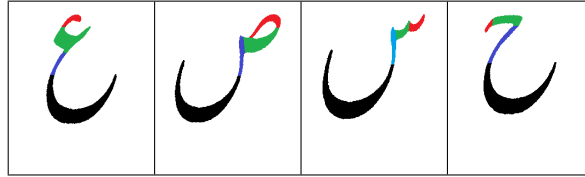


FIGURE 5.4: Four Strokes Characters, First, Second, Third and Fourth Strokes in Sequence Highlighted with Red, Green, Blue and Black Colors, Respectively

writing style specially in Nastalique, each character is written by following the special rules of calligraphy. Some sequence of strokes are used to write the character core shape and other are used to give the character's positional information [40]. For example the letters in Figure 5.4, the black stroke of all the characters are same and is used to indicate that letters are written in isolated. However, stroke having color other than black plays important role to disambiguate a letter from the other letter.

5.2.1 Hypothesis of the Study

The main motivation of this study is to do perceptual experiment to analyze the cognitive features(i.e. strokes) of Arabic letters which human brain uses to recognize. Usually human brain uses minimum strokes to recognize the characters [23]. In this study, a perceptual experiment has been conducted to extract the minimum set of strokes in sequences i.e. feature set which human brain uses to recognize Urdu Nastalique letters for reading.

The hypothesis of this study is, in Urdu Nastalique, a character has

1. **Primary Stroke(s):** The primary stroke(s) preserves core shape of character, does not change in different context and plays important role for the identification.

2. Secondary Stroke(s): Secondary strokes are supplementary for the identification of the characters. These strokes do not independently give any information for the identification of a specific character, but along with primary strokes give further cues for letter identification. There can be zero or more secondary strokes.

3. Connector Strokes: The connector is used to make contextual connection with other letters and play less important role in the identification of character.

To concretely find out the contribution of strokes working as primary, secondary and connector strokes, we need to figure out how human brain uses these strokes for the recognition of a letter. To do this, the perceptual experiment is properly designed and executed. The experiment is intended to perceptually extract the sequence of strokes which are coarsely used for the recognition of a character. Up till now, no research has been carried out to find the feature set of Arabic script in terms of strokes. This perceptual study is conducted to analyze the Urdu Nastalique orthography in terms of contribution of minimum sequence of strokes for the recognition of Urdu letters.

5.2.2 Data Preparation

The experiment is designed by taking Urdu characters as input. To handle the experiment size, 21 characters classes based on the shape similarity of RASM are used. Each Urdu character is written using the sequence of strokes by following Nastalique calligraphic rules of writing. Therefore strokes and their sequence for each character are marked according to the rules of calligraphy. Eventually these strokes are verified by the expert calligraphers of Nastalique. The characters along with their sequence of strokes are given in Figure 5.5. Each stroke is labeled with the $Char_S_{\#}$ to maintain the positional information in character $Char$. The $S_{\#}$ represents the Stroke Number e.g. Jeem_4 will be treated as 4th stroke of character Jeem. Each stroke of the character is highlighted with the black color and remaining main body image is colored with gray color showing the significance of the stroke in the overall image of the main body.

Character	Character Stroke 1	Character Stroke 2	Character Stroke 3	Character Stroke 4
ا	 Alef_S ₁			
ب	 Bay_S ₁	 Bay_S ₂		
ح	 Jeem_S ₁	 Jeem_S ₂	 Jeem_S ₃	 Jeem_S ₄
د	 Dal_S ₁	 Dal_S ₂		
ر	 Ray_S ₁	 Ray_S ₂		
س	 Seen_S ₁	 Seen_S ₂	 Seen_S ₃	 Seen_S ₄
ص	 Swat_S ₁	 Swat_S ₂	 Swat_S ₃	 Swat_S ₄
ط	 Toay_S ₁	 Toay_S ₂	 Toay_S ₃	
ع	 Aaeen_S ₁	 Aaeen_S ₂	 Aaeen_S ₃	 Aaeen_S ₄
ف	 Fay_S ₁	 Fay_S ₂		
ک	 Kaf_S ₁	 Kaf_S ₂	 Kaf_S ₃	
ق	 Qaf_S ₁	 Qaf_S ₂		
ل	 Laam_S ₁	 Laam_S ₂		
م	 Meem_S ₁	 Meem_S ₂		

























	 Noon_S ₁	 Noon_S ₂		
	 Wao_S ₁	 Wao_S ₂		
	 GolHay_S ₁	 GolHay_S ₂		
	 Hamza_S ₁	 Hamza_S ₂	 Hamza_S ₃	
	 DoChashmiHay_S ₁	 DoChashmiHay_S ₂	 DoChashmiHay_S ₃	
	 ChotiYey_S ₁	 ChotiYey_S ₂	 ChotiYey_S ₃	
	 BariYey_S ₁	 BariYey_S ₂		

FIGURE 5.5: Stroke Sequence of Urdu characters, Main Stroke is Represented by Black Color, Remaining Main Body in Gray Color and Dots are Shown with Green Color

5.2.3 Participants

A total of 30 participants are selected to conduct the experiment. The selection criteria is used to finalize the participants. All participant are graduated or graduate students of different universities. It has been ensured that participants are fluent users of computers and are normal readers of Urdu. Based on this criteria, a total of nine participants with a mean age of 23.5 years (5 Female; 4 Male) are finally selected to participate in the experiment. All were native Urdu speakers having normal or corrected to normal visual acuity.

5.2.4 Apparatus and Stimuli

The stroke based perceptual experiment has been conducted on laptop (having Intel Core i7 Processor and a 15.5" monitor). To present visual stimulus and store the response of the participant, a software in Java platform is developed. The stimuli are 21 letters of Urdu alphabets covering all shapes of Urdu RASMs classes. These are printed and scanned at 600 DPI so that images can be obtained to segment the character image and to generate visual stimuli. Strokes of each character are marked which are verified by the expert calligrapher of Nastalique writing style. Each Urdu letter has varying number of strokes. An image processing technique is applied which processes the marked segment points, extracts the marked strokes and stores as $Char_S_{\#}$ i.e. stroke number giving positional information of stroke in character $Char$. A total of 54 strokes of 21 characters are extracted which are in Figure 5.5.

Another software is also developed which automatically generates the visual stimuli (also called experimental stimuli) of the characters using extracted strokes of respective character. The process to generate the experimental stimulus i.e. pseudo letter is given below. A letter is decomposed into its constituent strokes i.e. S_1, S_2, \dots, S_n , where n is the total number of strokes of a character. First, all single strokes are extracted as visual stimuli e.g. in Figure 5.6 single stroke visual stimuli are S_1, S_2, S_3, S_4 of character Jeem. Then visual stimuli of two strokes are formed by joining the strokes which are neighbors to each other in sequence i.e. stimuli of $S_1S_2, S_2S_3, \dots, S_{n-1}S_n$ as can be seen in Figure 5.7, the S_1S_2, S_2S_3 and S_3S_4 visual stimuli are formed for Jeem character. The three strokes in sequence are also joined to form the visual stimuli such that $S_1S_2S_3, S_2S_3S_4, \dots, S_{n-2}S_{n-1}S_n$. The visual stimuli having three strokes of letter Jeem are $S_1S_2S_3$ and $S_2S_3S_4$, and are shown in Figure 5.8. At the end, last stimulus is the complete character having all n strokes joined in sequence as shown in Figure 5.9. The software automatically generates the visual stimuli of one stroke, two strokes, three strokes and so on till n strokes by joining all permutation in sequence. Hence total visual stimuli are 104 for all characters, having single stroke, two strokes, three strokes and four strokes.









		S ₁
		S ₂
		S ₃
		S ₄
Visual stimulus	Highlighted in Complete Character with Black Color	Stroke Sequence

FIGURE 5.6: Single Stroke Visual Stimuli







		S ₁ S ₂
		S ₂ S ₃
		S ₃ S ₄
Visual stimulus	Highlighted in Complete Character with Black Color	Stroke Sequence

FIGURE 5.7: Two Strokes Visual Stimuli





		$S_1S_2S_3$
		$S_2S_3S_4$
Visual stimulus	Highlighted in Complete Character with Black Color	Stroke Sequence

FIGURE 5.8: Three Strokes Visual Stimuli



		$S_1S_2S_3S_4$
Visual stimulus	Highlighted in Complete Character with Black Color	Stroke Sequence

FIGURE 5.9: Four Strokes Visual Stimuli

Each visual stimulus image is generated by center aligned at fixed sized white image having 388 pixels height and 524 pixels width (maximum height and width of the all visual stimuli). The letter stimuli are approximately subtended 1.35° of visual angle horizontally.

5.2.5 Procedure

Participants carried out the experiment individually in a room. The experiment started with the presentation of one of the 21 characters as input which is selected randomly for each participant. The duration of the presentation was one second. The participants were instructed to memorize the input Urdu character against which the next stimulus is going to be decided as minimum stroke sequence which is sufficient to recognize the input character. After that the "+" sign, pasted on

same sized white image, is shown for one second. The visual stimulus is presented to the participant for one second so that participant can encode visual stimulus as minimum and sufficient sequence of strokes to recognize respective input Urdu character. The response image i.e. 'X' is shown to user giving message to press RIGHT ARROW KEY for YES and LEFT ARROW KEY for NO. The response image remains on the screen for two seconds or less till the participant gives response. The complete iteration of the experiment is shown in Figure 5.10, displaying example of Jeem character as input and Jeem_S₂S₃ visual stimulus for the recognition task. The response of the participant is saved. For each of the randomly selected Urdu character, visual stimulus of different number of strokes i.e. single stroke having 54 visual stimuli (selected randomly), two strokes having 33 stimuli (selected randomly), three strokes having 13 stimuli (selected randomly) and four strokes having 4 stimuli (selected randomly) are presented to participant to get response. The accuracy score for each n length strokes stimulus is recorded as minimum strokes sequence used for character identification of the respective character. Against each of 21 Urdu characters, the respective participant response is gathered by showing a total of 104 visual stimuli automatically created from strokes of 21 characters. To start the experiment with each participant, complete experiment of one character is executed as trial and response is stored. The results are analyzed and feedback is discussed with the participant so that participant has complete training to execute the experiment.

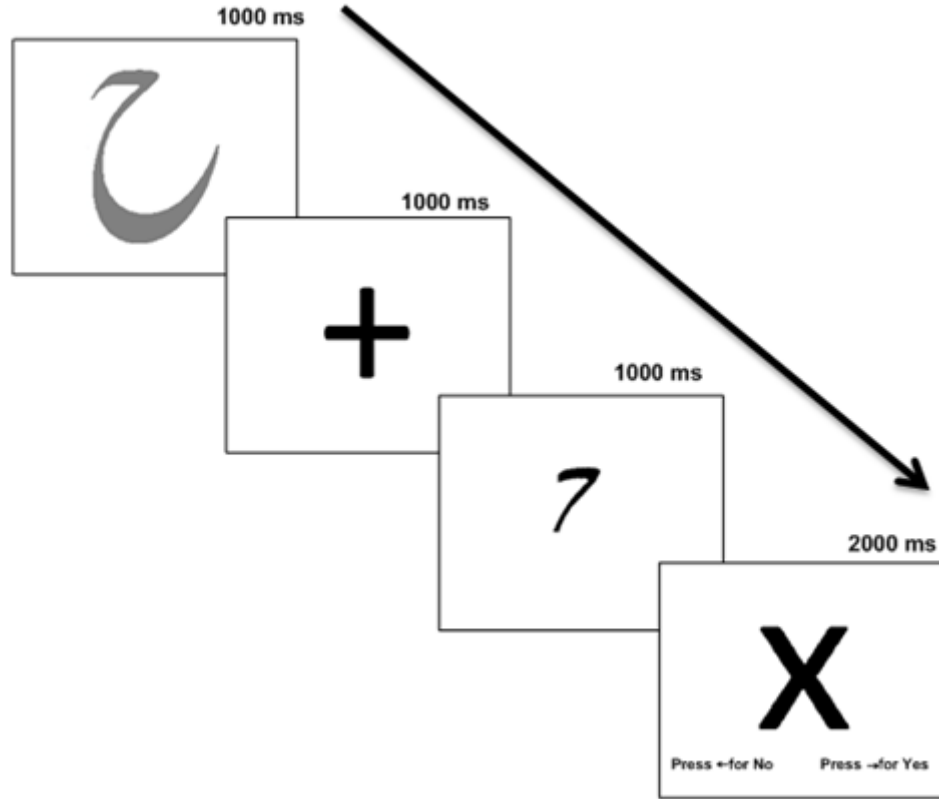


FIGURE 5.10: Display Example of Jeem Character as Input and Jeem_S₂S₃ Visual Stimulus for Recognition Task

5.3 Results and Discussion

To control the sample space, the perceptual experiment has been carried out for the single letters at isolated position. However, based on the rules of the orthography, following analysis is used to generalize the findings for each letter at all positions.

5.3.1 Data Analysis

1. The accuracy is computed as correct if sequence of strokes against which a participant presses YES is part of the respective character which is used to recognize the character. The accuracy is computed over all the participants' total correct response divided by total number of participants. The strokes which are not part of the respective character but participant presses YES as minimum number of strokes for the recognition of character are not considered. This is because, the focus of the experiment is to identify the sequence of strokes a letter which are used for identification, otherwise the confusions

between the strokes would be analyzed which is another research area.

2. All the sequences of strokes whose accuracy is greater than 50% are selected for further analysis. Only those sequences of strokes are considered for analysis against which more than 50% of the participants give response that there is correlation of sequence of strokes with the recognition of the respective character. All selected sequences of strokes of the respective character are further processed in such a way that minimum sequence of stroke having highest accuracy score is processed first.
3. Primary stroke is core stroke or sequence of strokes which represents main shape to identify the character. This shape of a character does not change in different context. A conservative measure is set heuristically at 85%. Therefore minimum stroke sequence which has more than 85% accuracy is considered as candidate for the primary stroke.
4. Secondary strokes are supplementary for the identification of the letters. Independently these strokes do not give any information for the identification of character. Such strokes do not have reasonable accuracy for the identification of letter, having very low accuracy (<50%) for the identification of the character. However along with primary strokes give further cues for letter identification, giving 100% accuracy.
5. Connector have less contribution for the recognition of the character, rather it gives contextual shaping information of the connection of a character with next character. These strokes change their shapes according to the contextual connection of a character .
6. The characters belong to set پے، ء، ر، د، ل are non-joiner character classes and do not join with next characters therefore these characters do not have connector strokes.
7. Some strokes which change their shape based on the position of a character in a ligature cannot be labeled as the primary stroke e.g. the S_2 stroke of character ب having 100% accuracy cannot be labeled as primary stroke for

the recognition of ب character. This is because, this S₂ stroke changes its shape based on the position of character ب in ligature. Such strokes are marked as connector, as can be seen in Figure 5.11. The S₂ stroke with varying shape highlighted with gray color is connector. The diacritics also play role in the recognition of a character. The reason behind the 100% accuracy of S₂ stroke of character ب is that the S₂ stroke and associated diacritic dot form clear shape of character ب. However, the primacy stroke is S₁ which is consistent for all the contextual positions of character ب (the primary stroke i.e. S₁ is highlighted with black stroke in Figure 5.11). The diacritic associated with S₁ of character ب is far from the stroke may create confusion for participant to not recognize this as meaningful stroke of character ب (BEH).

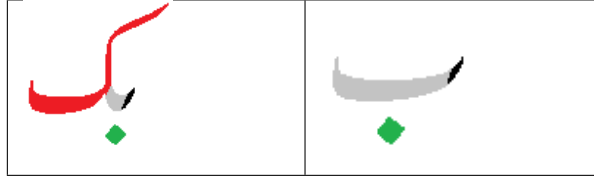


FIGURE 5.11: Contextual Shapes of S_2 Stroke of Character BEH Highlighted with Black Color

5.4 Results and Analysis of Primary, Secondary and Connectors

All visual stimuli of sequence of strokes of characters which have more than 50% accuracy are extracted and analyzed. However, the detailed accuracy of response of participants against each of the stimuli are given in Appendix-B. After doing detailed analysis based on the rules discussed above, the primary, secondary and connector stroke(s) are marked for each character. The characters are categorized into single, two, three and four strokes letter. The category wise analysis is discussed in subsequent sections.

5.4.1 Single Strokes Characters

In Urdu character set, only one character has single stroke i.e. ا (ALEF). The participant's accuracy for single stroke of letter ALEF is given in Table 5.1. This stroke has 100% accuracy resulting S_1 stroke is primary stroke of ا (ALEF), as can be seen in Table 5.2.

TABLE 5.1: Characters' Strokes along with Participants Accuracy for Single Stroke Character

Characters	Stroke Sequence	Stroke Sequence Shape in Ligature	Strokes Sequence	Se-	Accuracy
ا	ا	ا	S_1		100%

TABLE 5.2: Primary, Secondary and Connectors of Single Stroke Letters

Char	Primary Strokes	Secondary Strokes	Connectors
ﺍ	ﺍ S ₁	—	—

5.4.2 Two Strokes Characters

Urdu character set has 11 characters which are written using two strokes. The response of the participants against each visual stimulus is processed and accuracy is computed. All the visual stimulus which have more than 50% accuracy are extracted and presented in Table 5.3.

TABLE 5.3: Characters' Strokes along with Participants Accuracy for Two Strokes Characters

Char	Stroke Sequence	Stroke Sequence Shape in Ligature	Strokes Sequence	Se-	Accuracy
ب	ب	ب	S ₁		56%
ب	ب	ب	S ₂		100%
ب	ب	ب	S ₁ S ₂		100%
د	د	د	S ₁		89%
د	د	د	S ₁ S ₂		100%
ر	ر	ر	S ₁		89%
ر	ر	ر	S ₁ S ₂		100%
ف	ف	ف	S ₁		100%
ف	ف	ف	S ₁ S ₂		100%
ق	ق	ق	S ₁		89%
ق	ق	ق	S ₁ S ₂		100%
ل	ل	ل	S ₂		11%
ل	ل	ل	S ₁ S ₂		100%
م	م	م	S ₁		100%
م	م	م	S ₁ S ₂		100%
ن	ن	ن	S ₁		89%
ن	ن	ن	S ₂		67%
ن	ن	ن	S ₁ S ₂		100%
و	و	و	S ₁		89%
و	و	و	S ₁ S ₂		100%
ہ	ہ	ہ	S ₁ S ₂		100%
ے	ے	ے	S ₁ S ₂		56%

The detailed analysis of all filtered visual stimuli of each character is carried out. Started from the minimum strokes sequence, the primary, secondary and connector are marked according to the rules, already discussed. Table 5.4 gives details of the primary, secondary and connector strokes against each character.

1. The primary stroke of letter ب (BEH) is S_1 by applying rule 7 discussed in Data Analysis sub-section .
2. The character ج (Laam) is also the special case to identify the primary and connector strokes. Based on accuracy, the S_1S_2 both strokes can be considered as the primary strokes. However, based on shape analysis of ج (Laam) at different contextual positions and consistent stroke shape, the S_1 is the primary stroke. However, the isolated shape of S_1 is confused with primary stroke of ا (ALEF) due to the shape similarity. However, this confusion is resolved by doing contextual processing of strokes. The character ج (Laam) has a primary stroke i.e. S_1 and connector i.e. S_2 .The S_2 stroke changes its shape based on the position in a ligature as can be seen in Figure 5.12, whereas primary stroke ا (ALEF) does not have a connector. As can be seen in Figure 5.12 the S_1 shape remains consistent, highlighted with black color, that means it is primary stroke. The S_2 is the connector, which changes its shape based on the contextual connection as can be seen in Figure 5.12.

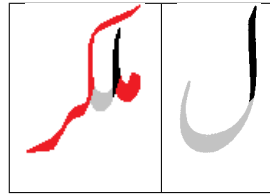























FIGURE 5.12: Contextual Character Shaping of Laam, Primary Stroke Highlighted with Black and Connector Highlighted with Gray Color

3. ا (Gol-HAY) character is special case which requires both S_1S_2 stroke sequence for the recognition of character Gol-HAY. Each single stroke has accuracy less than 50% that means S_1 and S_2 separately do not contribute in character identification.

TABLE 5.4: Primary, Secondary and Connectors of Two Strokes Letters

Char	Primary Strokes	Secondary Strokes	Connectors	Char	Primary Strokes	Secondary Strokes	Connectors
ب	 S ₁	—	 S ₂	م	 S ₁	—	 S ₂
د	 S ₁	 S ₂	—	ن	 S ₁	—	 S ₂
ر	 S ₂	 S ₁	—	و	 S ₁	 S ₂	—
ف	 S ₁	—	 S ₂	ہ	 S ₁ S ₂	—	—
ق	 S ₁	—	 S ₂	ے	 S ₂	 S ₁	—
ل	 S ₁	—	 S ₂				

5.4.3 Three Strokes Characters

A total of five character are written using three strokes. In the same way, results of each letter visual stimulus are shortlisted by applying accuracy threshold. The shortlisted sequence of strokes along with accuracy are given in Table 5.5. Based on detailed analysis, the primary secondary and connector strokes are mentioned in Table 5.6.

TABLE 5.5: Characters' Strokes along with Participants Accuracy for Three Strokes Characters

Char	Stroke Sequence	Stroke Sequence Shape in Ligature	Strokes Sequence	Se-	Acc
ب	ب	ب	S ₁ S ₂		100%
ب	ب	ب	S ₁ S ₂ S ₃		100%
ک	ک	ک	S ₁		56%
ک	ک	ک	S ₁ S ₂		100%
ک	ک	ک	S ₂ S ₃		78%
ک	ک	ک	S ₁ S ₂ S ₃		100%
د	د	د	S ₁ S ₂ S ₃		100%
ع	ع	ع	S ₂		89%
ع	ع	ع	S ₁ S ₂		100%
ع	ع	ع	S ₂ S ₃		100%
ع	ع	ع	S ₁ S ₂ S ₃		100%
ی	ی	ی	S ₁		78%
ی	ی	ی	S ₂		67%
ی	ی	ی	S ₁ S ₂		100%
ی	ی	ی	S ₂ S ₃		78%
ی	ی	ی	S ₁ S ₂ S ₃		100%

TABLE 5.6: Primary, Secondary and Connectors of Three Strokes Letters

Char	Primary Strokes	Secondary Strokes	Connectors
	 S ₁ S ₂	 S ₁	—
	 S ₁ S ₂	—	 S ₃
	 S ₁ S ₂ S ₃		
	 S ₂	 S ₁ and S ₃	—
	 S ₁ and S ₂	—	 S ₃

5.4.4 Four Strokes Characters

The last category is the four strokes characters which are a total of four characters. Each character has ten visual stimuli. These are shortlisted based on the threshold of accuracy which is 50%. The shortlisted strokes are presented in Table 5.7. The primary, secondary and connector strokes are further analyzed. Table 5.8 gives details of strokes sequence along with accuracy.

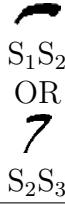








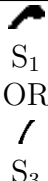

5.5 Discussion

Each stroke has perceptually different behavior for the recognition of characters to read the text. Based on the functionality/behavior, these strokes are categorized into three classes; (1) Primary strokes, (2) Secondary strokes and (3) Connectors.

TABLE 5.7: Characters' Strokes along with Participants Accuracy for Four Strokes Characters

Char	Stroke Sequence	Stroke Sequence Shape in Ligature	Strokes Sequence	Se-	Acc
ﺚ		ﺚ	S ₂		56%
ﺚ		ﺚ	S ₁ S ₂		100%
ﺚ	7	ﺚ	S ₂ S ₃		100%
ﺚ	7	ﺚ	S ₁ S ₂ S ₃		100%
ﺚ	ﺚ	ﺚ	S ₂ S ₃ S ₄		100%
ﺚ	ﺚ	ﺚ	S ₁ S ₂ S ₃ S ₄		100%
ﺱ		ﺱ	S ₁ S ₂		100%
ﺱ	ﺲ	ﺱ	S ₁ S ₂ S ₃		100%
ﺱ	ﺱ	ﺱ	S ₁ S ₂ S ₃ S ₄		100%
ﺹ		ﺹ	S ₁ S ₂		100%
ﺹ	ﺺ	ﺹ	S ₂ S ₃		56%
ﺹ		ﺹ	S ₁ S ₂ S ₃		100%
ﺹ	ﺹ	ﺹ	S ₁ S ₂ S ₃ S ₄		100%
ﻉ		ﻉ	S ₂		89%
ﻉ		ﻉ	S ₁ S ₂		100%
ﻉ	ﻊ	ﻉ	S ₂ S ₃		100%
ﻉ	ﻋ	ﻉ	S ₂ S ₃		100%
ﻉ	ﻉ	ﻉ	S ₂ S ₃ S ₄		100%

TABLE 5.8: Primary, Secondary and Connectors of Four Strokes Letters

Char	Primary Strokes	Secondary Strokes	Connectors
چ	 S ₁ S ₂ OR S ₂ S ₃	—	 S ₄
س	 S ₁ S ₂	 S ₃	 S ₄
ص	 S ₁ S ₂	 S ₃	 S ₄
ع	 S ₂	 S ₁ OR S ₃	 S ₄

The findings of this perceptual experiment can be further generalized according to minimum length of strokes sequence called primary strokes, which are mainly used to recognize the character. The primary stroke for Urdu character set having single, two and three strokes are given in Table 5.9, Table 5.10 and Table 5.11 respectively. The actual number of strokes against each character is also mentioned. As can be seen, a total of 54 strokes (adding Total Strokes of Characters of Table 5.9, Table 5.10 and Table 5.11) are reduced to 30 primary strokes (adding Primary Stroke(s) of Table 5.9, Table 5.10 and Table 5.11) which a human brain uses to recognize the Urdu characters. The secondary strokes can also be used to further improve the recognition results. Finally results also reveal that each character stroke has different behavior as primary, secondary and connector strokes, see Table 5.2, Table 5.4 and Table 5.6.

5.6 Conclusion

In this chapter, strokes of Urdu characters are perceptually analyzed. Based on the results, three categories of strokes are finalized; (1) primary, (2) secondary and (3) connector strokes. It is being observed that participants recognize the stimulus as

TABLE 5.9: Characters having Single Stroke as Primary Stroke represented with black color and diacritics with green color

Character	Primary Stroke	Total Strokes of Character
ا	ا	1
ب	ب	2
د	د	2
ر	ر	2
ع	ع	4
ف	ف	2
ق	ق	2
ل	ل	2
م	م	2
ن	ن	2
و	و	2
ع	ع	3
ک	ک	2
Total Strokes	13	28

a character when primary stroke is present. This perceptual experiment suggests that complexity of recognizing each character having multiple contextual shapes can be drastically reduced to the limited set of primary strokes. So, a total of 54 strokes of all Urdu characters can be reduced to 30 primary strokes. Moreover, recognition output based on the primary strokes can be further processed using secondary and connector strokes to improve the recognition confidence.

TABLE 5.10: Characters having Two Strokes as Primary Stroke











Character	Primary Stroke	Total Strokes
ح	 OR 	4
س		4
ص		4
ط		3
ک		3
ہ		2
ی		3
Total strokes for recognition	14	23

TABLE 5.11: Characters having Three Strokes as Primary Stroke

Character	Primary Stroke	Total Strokes
		3
Total strokes for recognition	3	3

Chapter 6

Image Dataset of Urdu

Nastalique Document Images

The accuracy of the OCRs depends to a great degree on thorough analysis of the variations of the document images and effectiveness of the developed techniques on large dataset. In addition, a large amount of standard data is also required covering all real varieties of document images so that different techniques can be evaluated, compared and matured. The standard datasets of different languages [61] [70] [78] are used to compare the performance of existing state of the art techniques. Such datasets play an important role for the development and maturation of the algorithms which result in overall performance improvement of OCR systems. Hence, there is a significant need of the standard corpus in pattern recognition and OCR research. The accessibility of the standard corpus not only facilitates the researchers to do research but also provides a platform to evaluate different techniques. This is because initially, most of the researchers report results tested on their own datasets. Currently, for the development of Urdu OCR, Urdu Printed Text Image (UPTI) dataset [85] is commonly used [98] [74] [72] [73]. This dataset is synthetically generated by selecting more than 10,000 text lines from different domains including politics, social and religion, copied from Daily Jang Urdu newspaper. The text lines are written in Alvi Nastaleeq font style. This font

style was created in 2007 by Amjad Hussain Alvi¹.

The tremendous research in the field pattern recognition shows that significant amount of training data covering variety of context improves the overall recognition accuracy. To develop a robust Urdu Nastalique character recognition system, a large amount of image data is required covering contextual variations of ligatures and characters. In addition, the corpus must cover the real examples of images having different paper and printing qualities.

The focus of this thesis is to develop a recognition system for published Urdu books and magazines written in Noori Nastalique. A comprehensive image dataset is required covering variety of font sizes, paper and printing quality variations. The variation of character and ligature context is also required to be covered in the dataset. The standard data set is also required to compare the performance of presented OCR technique with state of the art Urdu OCR systems. Hence, to meet this image dataset requirement for Noori Nastalique font style, the following datasets are used. The details of each of the datasets are given in sub-sequent sections

1. Dataset-1: Real Image and Ligature RASM Dataset
2. Dataset-2: Synthetic Ligature RASM Dataset
3. Dataset-3: Synthetic Noori Nastalique UPTI Dataset

6.1 Dataset-1: Real Image and Ligature RASM Dataset Generation

Dataset-1 is collected from the books written using Noori Nastalique writing style [9]. This image dataset is generated from different books to cover variety of paper and printing qualities. Most of the Urdu books and magazines are written using Noori Nastalique writing style having 14 to 40 font sizes. The normal text of the Urdu books is written using 14 and 16 font sizes. In children books, the normal text is written in larger font sizes range from 18-22 font sizes. The remaining font

¹<https://fonts2u.com/alvi-nastaleeq.font>

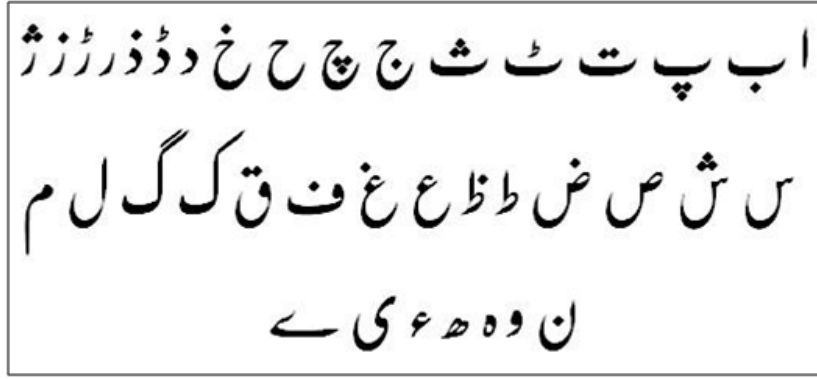


FIGURE 6.1: Urdu Character Set

sizes are used to write headings. Therefore three categories of image dataset are defined (1) Normal text images, (2) Normal text images for children and poetry books and (3) Headings text images. The complete process is designed for books collection, image corpus acquisition, image corpus labeling and ground truth (GT) data generation. A semi automatic approach is used to extract image dataset of RASM classes. Each of the sub-processes is detailed in subsequent sections.

6.1.1 Corpus Collection

Corpus collection process is divided into two main phases i.e. corpus design and corpus development [9]. The selection criteria to select the books and pages for scanning, is defined in the corpus design phase. The scanning of pages to generate the image dataset, organization of the images and generation of the ground truth (GT) information of the images are carried out in corpus development phase. Details are presented in subsequent sections. The selection of books is based on the criteria given below.

1. Character Set and Symbols: The image corpus should cover Urdu alphabet given in Figure 6.1 below, Latin digits (0, 1, 2, 3, 4, 5, 6, 7, 8, 9), English characters (A-Z, a-z), Urdu digits (۰, ۱, ۲, ۳, ۴, ۵, ۶, ۷, ۸, ۹), Urdu aerab (آ, اُ, اِ, اَ, اِ, اُ, اِ, اَ) and other symbols of Urdu.
2. Font Size and Style: Most of the Urdu books are written using Noori Nastalique font therefore the target font style is Noori Nastalique. The selected font sizes are 14-40 because normal text in books is written using 14 and 16 font sizes,

children's books are written using 18-22 font sizes and heading size ranges from 24-40.

3. Multiple Domains: Books are selected from multiple domains for each font size category to address the requirement of a balanced corpus.
4. Publishers and Publication Date Variety: Books published from multiple publishers of different cities are selected. In addition within a city, variety of publishers is also considered. In addition, publication date also affects printing as well as paper quality of books. Therefore while selecting the books, this parameter is also considered and books having variety of publication dates are selected.
5. Page/Printing Quality: Paper and printing qualities also affect image quality. To develop the standard dataset for Urdu image, all these varieties are included in the Urdu text image corpus.

6.1.2 Corpus Development from Books

Based on the availability of books according to the above mentioned criteria, the number of books and pages are selected according to font wise category. For each of the normal font size i.e. 14 and 16 font sizes, at least 100 books are selected and five pages from each book are tagged to be scanned to generate image dataset. For the second category i.e. children books, at least 30 books for each of the shortlisted font sizes i.e. 18, 20 and 22, are selected. At least five pages from each book are selected to scan. To generate image corpus of third category i.e. heading, at least 20 books for each of 24, 28, 32, 36 and 40 font sizes are selected and at least 10 headings from each book are marked to scan.

The selected pages of each book are scanned at 300 DPI using HP Scanjet G3110 scanner. During scanning, two types of images are scanned in gray scale format; (1) image without cropping the region of interest and (2) image with cropping the region of interest, both samples are given in Figure 6.2. The images without cropping the region of interest are scanned for the researchers who want to do research on page frame detection of Urdu document images. To generate image corpus for



(a) Gray Scale Un-cropped (b) Gray Scale Cropped

FIGURE 6.2: Sample of Gray Scale Cropped and Un-cropped Region of Interest [9]

headings, only heading textual area is extracted and saved during scanning. All the images are saved in JPG file format.

6.1.3 Corpus Organization

The intelligent labeling of the image dataset is essential for the research and development on the desired data type. It is normally done manually and is time consuming task to ensure error free data labeling. The data labeling helps to extract the desired data automatically. To maintain the presented Urdu image corpus in an orderly manner, image corpus for each font size is maintained separately. The cropped (edited) and un-cropped (unedited) images of gray scale are also maintained. The naming convention to save the images is defined so that any of the four versions, book information and font size etc. can be extracted by simply manipulating the images' names. Each image name for normal text (for 14 to 22 font sized text images) has following tags.

A.B_*C.D_E.F.G.jpg

e.g. G_UE_B13_R_P26_F14.jpg where

1. **A** represents the image format information i.e. gray scale represented by **G**.
2. **B** tag represents whether the scanned image is cropped (edited) represented by **E**, or un-cropped (unedited) represented by **UE**.
3. ***C**: When **B** tag is E then **C** tag is used to indicate editing type i.e. image is cropped for this corpus. Image name does not have C tag when B tag is UE that means image is Un-Edited.

4. **D** tag defines a unique book number (assigned manually) of the image from which it is scanned. The book number has **B** as prefix letter indicating book. This book number can be used to get further information about book including book name, author, publisher, publications date and domain which is maintained in separate file.
5. **E** indicates that type of the content of scanned image. The image can have normal (or regular) text represented as **R**, figure represented as **I**, table of contents represented as **T**.
6. **F** is used to represent the page number of book which is scanned to generate the image. The page number is defined with letter **P** as prefix.
7. **G** is last tag which is used to represent the font size of text image. Depending on the font sizes appeared in the text of the image, there can be multiple entries of font size, each is defined with prefix **F**.

The image corpus for each font size of headings is also maintained separately. As heading images are actually cropped from the document image during scanning. Therefore cropped and un-cropped versions are not maintained explicitly. The naming convention for the heading image is defined as

A_D_H_F_H#_G.jpg

e.g. G_B149_H_P34_H1_F32.jpg where

A, **D**, **F** and **G** tags are same as mentioned above. The **H** is used to define the type of the image i.e. **H** indicating the image is of heading. There can be more than one headings on same page. The **H#** is used to define the sequence number of heading in the document image from which the heading image is extracted. The heading number is defined with prefix letter **H** as can be seen in the example.

The complete information of each font size corpus is also maintained manually in separate file during scanning of images. This file contains information of each scanned image such as book ID, book name, author name, publisher, year of publication, city, total number of pages, domain, image name, available font sizes

in image, and columns (either 1 or 2) etc. This information is cross verified to generate the error free details of an image.

6.1.4 Text Corpus as Ground Truth (GT) of Images

To process and recognize document images of the reported image corpus, parallel typed version of each image is also generated as ground truth data. This GT data will assist the researchers to extract training and testing data for classification and recognition by developing systems for segmentation of lines and ligatures. In addition, this parallel text corpus is also helpful for the researchers to develop language models using contextual information for post-processing of OCR system to improve the accuracy. Each scanned document image is typed by two typists. The instructions are given to type text as is to have exact typed context of the image. This means the number of lines in text files must be same as number of lines in document image, see Figure ???. A total of 2,843 images are typed. Both versions of typed data are manually verified and mistakes are removed.

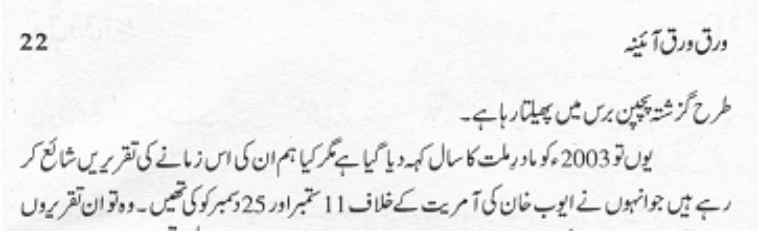
Image	
Typed Text	<p>ورق ورق آئینہ 22</p> <p>طرح گزشتہ پچپن برس میں پھیلتا رہا ہے۔</p> <p>یوں تو 2003ء کو مادر ملت کا سال کہہ دیا گیا ہے مگر کیا ہم ان کی اس زمانے کی تقریریں شائع کر رہے ہیں جو انہوں نے ایوب خان کی آمریت کے خلاف 11 ستمبر اور 25 دسمبر کو کی تھیں۔ وہ تو ان تقریروں</p>

FIGURE 6.3: Sample of Image and Corresponding Typed Text

6.1.5 Semi-Automated RASM Classes Generation

The pages scanned from different books are further processed to generate the image dataset of the RASM classes. The instances of RASM (main body) classes are also extracted for each font size image corpus. To do so, a semi automatic process is defined to reduce time and human effort. A basic classification and recognition

system is developed using Dataset-2, discussed in next section. Tesseract, an open source multilingual OCR [94] is used as classification engine to develop Urdu RASM classes recognition system. All the RASM classes of Dataset-2 are used for the training of the Tesseract. The trained Tesseract takes main body image as input. Tesseract recognizes and classifies the input main body image into the respective class. An automatic system is develop which takes gray scale Urdu document image as input. The input document image is binarized using algorithm defined in [71]. The connected components are extracted from the binarized image and categorized into diacritics and main bodies using dimensional features. The main bodies are given to the Tesseract-based classification system which classifies the main body into respective class. The classified image is placed in directory with the name of the recognized main body class. While generating the instance images, the name of main body image is defines such that the positional information of bounding box is also appended with RASM instance image name. This positional information is helpful in the manual pass of verification to open the actual image and verify in case of confusions.

After classification, a manual pass is required remove the misclassification issues of the Tesseract. The printed main body class string as name of the directory helps to verify the images accurately and quickly. There are some cases when, the image is misclassified and does not correspond to the sting of classified main body class and hence these are removed from the respective directory. Some cases of misclassification are; (1) figure connected components are classified as main bodies, labeled as Figure error category (2) broken main body labeled as Broken MB error, (3) Attached connected components labeled as MB Attached with another MB and MB Attached with diacritic errors and (4) special symbols labeled as Other Font style error. The sample image of each error category is given in Table 6.1. The misclassification of RASM images is also due to the confusing shapes of different ligatures, some example images are given in Table 6.2. To verify such cases, the instance images are verified from the actual document image (using positional information stored in image name) and are classified according to actual ligature. All the erroneous cases are deleted. to clean the training and

testing dataset for the development of RASM based classification and recognition system. In second pass, all unclassified main bodies are classified manually to have significant number of instances of main body classes for the classification and recognition of Urdu document images.

TABLE 6.1: Special Cases of Wrong Classification




















Error category	Instance Image Sample	Line Image of Document Image
MB Attached with Another MB		شہد کی اسی خوبی وجہ سے جن علاقوں میں شہد کا استعمال بطور غذا کیا
MB Attached with Diacritic		اب تو دکھ سے گفتاریں رو پڑیں اور بولیں۔ ”ہائے!..... اب تو ہم شکار کے قابل
Broken MB		کتاہیں لکھنے والے شوگر پر کتاب لکھ کر اس میں سے قوم کو نجات دلانے
Other Font Style		ایں سعادت بزور بازو نیست
Figure		

TABLE 6.2: Main Body Classes Confusions

MB Class	Image	Confusing Class	MB	Image
د		و	و	
با		ما	ما	
بد		مد	مد	
ے		بے	بے	
س		سن	سن	
مل		بل	بل	
سر		سبر	سبر	

6.1.6 Corpus Results

The summary of the statistics of Dataset-1 are summarized in this section, further details are given in [9]. The published Urdu books are selected to cover the variety of domains including literature, poetry, religion, biography, novel, interviews, culture/travel, history, autobiography, science, short stories and character representation. During development, the complete information about the books such as ID, name, author name and domain is maintained. The summarized information of number of books, domains and authors for each of the selected font size is given in Table 6.3.

Parallel typed text corpus of respective image corpus is also generated for each font size. The characters are joined together to form the ligatures. For each font size, lines and ligatures information is also computed from typed text corpus which is given in Table 6.4.

TABLE 6.3: Statistics of Urdu Image Corpus [9]

Font size	Book/Magazine count	Number of document images	Domains	Authors
14	101	593	18	76
16	116	595	19	100
18	30	150	10	23
20	45	149	2	24
22	56	151	2	21
24	21	461	18	24
28	21	202	6	21
32	23	186	9	21
36	31	226	7	22
40	26	199	7	22

The document images of each font size are further processed to extract the main body instance images. A semi-automatic process is used for this purpose . The details of main body instance images for each font size are given in Table 6.5. An example subset of instance images of ten main body classes are given in Table 6.6.

TABLE 6.4: Font Wise Lines and Ligature Statistics of Corpus

Font Size	Total document images	Lines	Total Ligatures	Unique Ligature	Average Lines per image	Average Ligatures per Line
14	591	13,712	386,648	6,452	23	28
16	528	11,080	306,080	5,938	20	27
18	150	2,622	60,056	2,872	18	23
20	149	2,318	54,657	2,204	16	24
22	151	1,857	43,121	1,865	12	23
24	461	463	3,961	883	1	9
28	202	203	1,424	502	1	7
32	186	274	2,874	616	2	11
36	226	260	1,776	537	1	7
40	199	222	1,510	498	1	7

6.2 Dataset-2: Synthetic Ligature RASM Dataset

A subset of main body types from the complete set is extracted for different font sizes from the dataset of books, see Table 6.5. Synthetic Ligature RASM dataset is developed to have significant ligature RASM classes dataset. To meet the dataset requirement covering a wider variety of character and ligature coverage, the efficient way is to select text content which covers all variation and then synthesize that content to generate the image dataset. This is conventional way to quickly generate the dataset and to formulate basic research in computer vision and pattern recognition [85]. Hence, the ligatures' images dataset is synthetically generated along with ground truth information. This dataset is developed for each of the selected font size i.e. 14, 16, 18, 20, 22, 24, 28, 32, 36, 40 and 44. The detailed process to generate the dataset is illustrated below.

6.2.1 High Frequent Urdu Ligatures Selection

Different Urdu text corpora including 18 million words corpus[42] and CLE Urdu Digest Corpus of 100K words corpus [99] are processed to extract unique words

TABLE 6.5: Font Wise Instance Images Statistics

Font Size	Number of Main Body Classes	Number of Instance Images
14	2,666	95,437
16	1,705	35,687
18	1,619	46,576
20	1,221	43,019
22	1,033	33,467
24	529	3,124
28	276	1,176
32	262	1,237
36	246	1,115
40	279	1,388

along with their frequency. The words frequency of CLE Urdu Digest Corpus is multiplied by 180 to balance the frequency ratio of 100K words versus 18 million words. All the words which have frequency greater than 100 are selected for further processing. The unique ligatures are extracted from the selected words. The frequency of ligatures appearing in different words are added to rank the ligature according to frequency. The character sequence of a ligature is converted to character class sequence (see Figure 2.6) sequence to convert the ligature to respective ligature RASM class, using rules given in Table 6.7. The conversion of the ligature to ligature RASM class is done by applying the character class conversion rules based on the position in the ligature. As can be observed in Table 6.7, some characters at specific position in a ligature take the shape of different character classes. The corpus has a total of 149,465 unique words of Urdu, which are formed by a total of 23,951 unique ligatures having 10,374 unique RASM classes.

After applying character class mapping rules, all ligatures are mapped to the ligature RASM classes. The unique list of ligature RASM classes are extracted along with the respective valid ligature string. This ligature string is then used

TABLE 6.6: Instance Images of Ten Main Body Classes




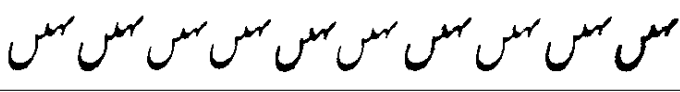
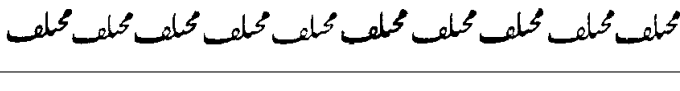
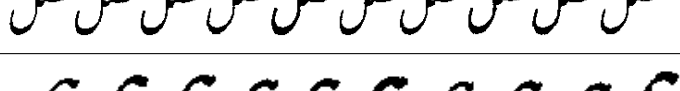

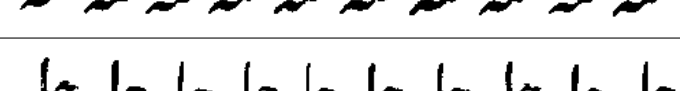


Sr. no	Instance Images of Main Body Classes
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

TABLE 6.7: Character Classes Mapping List

Character	Position (initial, medial, final and isolated)	Mapping Character Class
پ پ ت ث	All Positions	ب
ن ق ح خ	All Positions	ج
ڈ ڈ	All Positions	د
ر ز ز ش	All Positions	ر
س ش	All Positions	س
ص ض	All Positions	ص
ط ظ	All Positions	ط
ع غ	All Positions	ع
ف	All Positions	ف
ق	Final and Isolated	ق
ق	Initial and Medial	ف
ک گ	All Positions	ک
ل	All Positions	ل
م	All Positions	م
ن ل	Final and Isolated	ن
ن ل	Initial and Medial	ب

و	All Positions	و
و	All Positions	و
و	All Positions	و
و	All Positions	و
و	All Positions	و
و	All Positions	و
و	Initial and Medial	و
و	Final and Isolated	و

to generate the image dataset of ligature RASM classes. Top 5586 high frequent ligature RASM classes are selected to generate the dataset. These ligature RASM classes, also referred to as High Frequency Ligatures (HFL), cover 131,000 high frequency words.

6.2.2 Image Corpus Development

The next step is the development of image dataset of these 5586 ligature RASM classes. A total of 35 samples of valid ligature string of each class are generated in Inpage in Noori Nastalique writing style. Inpage software is usually used to publish Urdu books, magazines and newspaper in Noori Nastalique. Inpage renders ligatures instead of characters hence only those ligatures can be printed from the Inpage which exist in the valid ligature list to have properly rendered shape of that ligature. Therefore validity check of the ligature class is important to have proper shape of the ligature which is ensured by selecting the valid ligature string for each ligature class. To efficiently manage the dataset of each class, each ligature string is copied 35 times on the single page separated by space. The next ligature string samples are placed on the next page. Then these ligatures are printed and

scanned at 300 DPI having gray scale version to generate the image corpus. Each image has ligature samples of single ligature class so that an automatic system can be developed to extract, organize and clean the ligatures. The sample image of a ligature is given in Figure 6.4.

6.2.3 Corpus naming and organization

The image naming convention is defined and applied manually in an intelligent manner so that automatic data extraction techniques can be applied to extract the desired data. The naming convention for High Frequency Ligature (HFL) image corpus is also defined for each image to get relevant information from the image. The tags of HFL ligature image name is elaborated below:

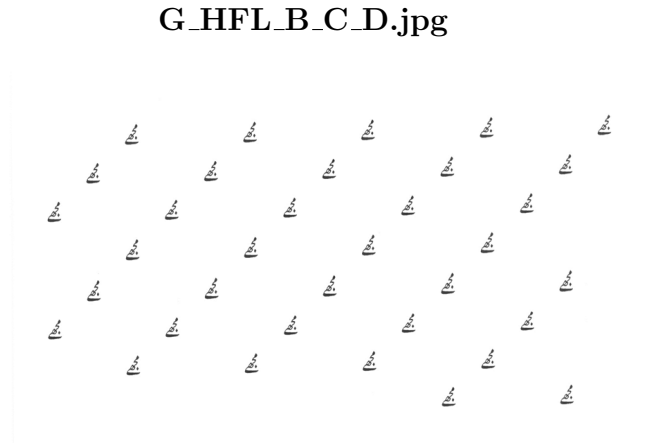


FIGURE 6.4: Sample image having 35 tokens of ligature RASM class

G defines that format of the image is gray scale (**G**)

HFL defines that the image is of **high frequency ligature**

B defines the ligature string of image

C defines serial number of ligature string in high frequency ligature list

D have prefix **F** and defines the font size of the ligature string at which it is generated

The metadata information is also maintained in separate file which contains information about each image in the corpus including ligature class, image name, printed ligature and its high frequency ligature serial number. This dataset is also publically available at <http://www.cle.org.pk/clestore/imagecorpora.htm>

6.2.4 Semi-automated RASM Classes Cleaning

An automatic system is developed which binarized the document images using [71]. The connected components are extracted. The dimensional features are used to segregate the diacritics and main bodies. As the ligature class information is saved in image name. The system automatically extracts the main bodies using dimensional information and place in the directory having same name as ligature RASM string. Later, a manual pass is carried out to clean each directory to have correct shapes of ligature RASM class in less time. During manual cleaning, it has been observed that some main bodies have attached diacritics. In addition, due to the thick thin stroke transition of ligature RASM stroke, sometimes main body stroke is broken. Some such examples of attached diacritics with RASM and broken main bodies, shown in Figure 6.8. All such types of erroneous images are deleted to clean the RASM images dataset. The ligature RASM class page is re-printed if the total number of tokens are less than 35 after manual pass. In the second pass, the other valid ligature string of the respective class is selected for printing to avoid diacritics attached and broken issues.


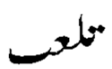


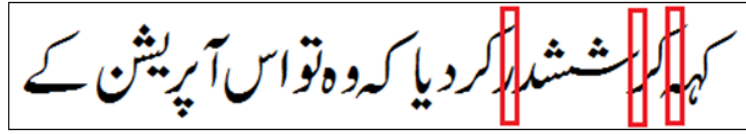
	
	
(a) Examples of Broken Main Body	(b) Examples of Attached Diacritics with Main Bodies

TABLE 6.8: Broken and Diacritics-attached Examples of Ligature RASM Classes

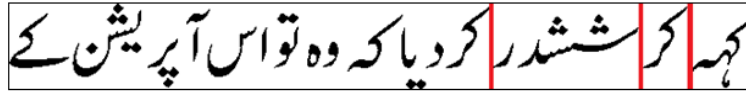
6.3 Dataset-3: Synthetic UPTI Dataset Generation

The Urdu Printed Text Image (UPTI) dataset [85] is commonly used to evaluate the performance of the technique with existing Urdu state of the art recognition techniques [98][74][72][85][75][5]. This dataset is synthetically generated by selecting more than 10,000 text lines from different domains such as politics, social and religion, of Daily Jang Urdu newspaper. The text lines are written in Alvi Nastaleeq font style. This font style was created in 2007 by Amjad Hussain Alvi². Publishing industry prefer Inpage software to publish Urdu books and magazines in Noori Nastalique writing style. Noori Nastalique is complex writing style as compared to the Alvi Nastalique. In Noori Nastalique, the ligatures are more compactly written as compared to Alvi Nastalique. The examples of text lines rendered in Noori Nastalique and Alvi Nastalique are given in Figure 6.9.

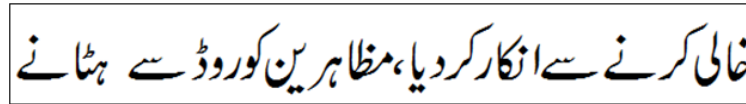
²<https://fonts2u.com/alvi-nastaleeq.font>



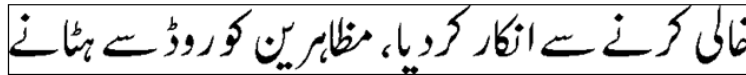
(a1) Text line in Noori Nastalique,
ligature overlapping highlighted with red rectangles



(b1) Text line in Alvi Nastalique,
no ligature overlapping, highlighted with red lines



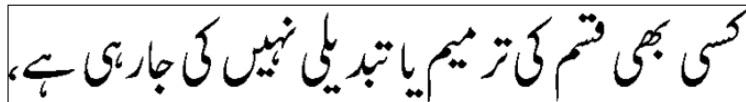
(a2) Text line in Noori Nastalique



(b2) Text line in Alvi Nastalique



(a3) Text line in Noori Nastalique



(b3) Text line in Alvi Nastalique

TABLE 6.9: Sample textlines rendered in Noori Nastalique(a) and Alvi Nastalique (b)

In this thesis, the main focus is to develop the Urdu text recognition techniques used for the digitization of Urdu books and magazines. However a dataset is required which can be used to compare the performance of the proposed techniques with the state of the art Urdu recognition techniques. Therefore subset of UPTI dataset is also rendered in Noori Nastalique font style of Inpage software. These text lines are printed and scanned at 300 DPI having gray scale version to generate the image corpus, namely Noori-UPTI dataset. This dataset has a total of 1600 text lines.

6.4 Conclusion

In this chapter, a comprehensive image dataset of Nastalique writing style is presented. Three different datasets are developed; (1) Dataset from scanned Urdu books, (2) synthetically generated RASM (main bodies) classes, and (3) Noori-UPTI, a subset of UPTI rendered in Noori Nastalique writing style. The dataset scanned from different books is developed to have actual examples of text images having paper and printing variations. To cover the variety of character and ligature context coverage, the Dataset-2 is developed. Dataset-1 and Dataset-2 are used to develop the system and Dataset-3 is eventually used to compare the performance of the system with state of the art Urdu recognition techniques.

Chapter 7

A Novel Cognitive inspired Computational Framework for Letters of Urdu Nastalique Recognition

7.1 Introduction

To develop the recognition systems, usually the dimensional, structural and geometrical features are extracted, termed as visual features ¹, from the images using image processing techniques. These visual features along with labels are classified using state of the art machine learning and deep learning approaches. The conventional methods for the recognition of Urdu document images are classified into the two main categories (1) ligature-based classification and recognition and (2) character-based classification and recognition. Despite recognition of state of the art learning approaches of Urdu document images gives promising results, still practical use of developed systems have drawbacks. Urdu character set having 39 letters of Urdu constitutes around 25,000 commonly used unique ligatures. This ligature set is not closed because addition of new word in Urdu language which

¹Visual features are extracted by doing some image processing techniques used to classify the image whereas cognitive features are set of strokes used to recognize characters.

can be a transliterated word of foreign language, may cause addition of new ligature. Therefore the ligature based solution for the recognition of Urdu text is not appropriate. Whenever a new ligature would be added in the language, the system would require to be retrained on the additional ligature so that this can be recognized. To handle this issue, the character-based system seems to be an optimal solution. As Urdu character set is close set therefore addition of new ligature would not affect the performance of recognizer as new ligature will be segmented into characters which would be recognized by the recognizer.

Both implicit character recognition and explicit character recognition techniques for Nastalique writing style have some issues. The overlapping of characters and ligatures make the system more complex especially feature computation module. The horizontal sliding window is normally used to extract the features of the respective character. When a character overlaps with other character in a ligature, noisy features are computed for training and recognition of respective character and cause confusions for the machine learning system. In the same way, the recognition of the contextual character shaping is also a challenging task requiring significantly more training data to learn. Due to these challenges, the existing character recognition techniques generate errors by introducing character insertions and deletions in the recognized text.

However, human can intelligently process and disambiguate the complex and confusing shapes and can recognize the text correctly. According to the experimental study reported by Fiset et al. [23], the complete letter is recognized by recognizing the important key features of letters. More specifically, human brain mainly focuses on some important features to recognize the characters, because 32% of upper case English letters and 24% of lowercase English letters are used to recognize the text [23]. In the same way, some core cognitive features also play significant role for letter identification of Urdu written in Nastalique writing style, see Chapter 4. Hence to solve recognition issues due to the complexities of Nastalique, a novel framework of character recognition inspired by cognitive model of reading is presented in this chapter. Human brains are prone to complex character shaping once human learns how to read the text. The characters are very intelligently

processed by eyes to convert the object into the sequence of cognitive features (i.e. sequence of strokes) so that human can recognize [19]. The complete letter is recognized by recognizing the important key features of letters [23]. More specifically, human brain mainly focus on some more important cognitive features i.e. sequence of strokes to recognize English characters [23]. To check this phenomenon for Urdu letter identification, effect of cognitive features i.e. strokes of Urdu character set written in Nastalique writing style is perceptually investigated, discussed in Chapter 4. The experimental results showed that Urdu character strokes can be categorized into three categories; (1) primary, (2) secondary and (3) connector. The perceptual experiment suggests that complexity of recognizing each character having multiple contextual shapes can be drastically reduced to the limited set of primary strokes.

The cognitive-inspired recognition framework based on cognitive features of Urdu is modeled to recognize the sequence of characters in a ligature. This technique resolves noisy feature computation due to the overlapping of characters in a ligature. In addition, the recognition of multiple contextual shapes is drastically reduced by focusing on recognized strokes of respective character. Later, these recognized strokes can be weighted to further improve the character recognition.

7.2 Research Hypothesis

The main research hypothesis is based on the argument that the computational model based on cognitive model of reading will improve the character recognition. Up till now, no research exists in the literature to recognize the Arabic letters using the cognitive model of reading. This model is based on cognitive features of Urdu Nastalique characters which human brain uses to recognize during reading, see Chapter 4 for further details. In the same way, the cognitive-inspired implicit character recognition framework will improve the recognition results.

7.3 Proposed Cognitive Inspired Framework

The proposed computational framework of character recognition is inspired by the cognitive model of reading, discussed in [91] [84]. During reading, the visual input

is processed to recognize the cognitive features i.e. the sequence of strokes. These cognitive features are further processed to recognize characters and words using phonological information, lexical processing (using lexicon and strokes corpus) and semantic processing (using lexicon and character corpus). The proposed computational model of character recognition is based on the bidirectional model of reading [91] to improve the recognition results using feedback. The bidirectional model uses the feedback from previous stage to improve the recognition results of the current stage. This feedback strategy eventually improves the overall recognition accuracy. The architecture of cognitive inspired computational model of recognition is given in Figure 7.1 with parallel architecture of cognitive model of reading [91]. The details of the demons(stages) of document image recognition are given below.

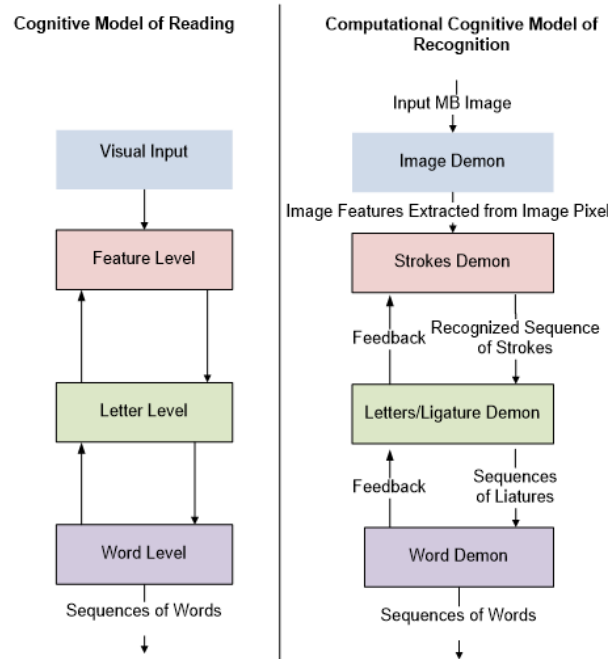


FIGURE 7.1: Architecture of cognitive based character recognition

7.3.1 Image Demon Formulation

At this stage, image stored as 2D array i.e. $I(x,y)$, is processed to compute the visual feature set i.e. V_F so that cognitive model can be trained to recognize the cognitive features, i.e. sequence of strokes. The local window is traversed according to the movement of the pen to write the ligature to extract the position

coded visual features. These features are computed by using image processing techniques. Hence a 2D array i.e. $I(x,y)$ is converted into visual feature set $V_F = (v_{f_1}, v_{f_2}, v_{f_3}, \dots, v_{f_n})$

7.3.2 Stroke Demon Formulation

This stage focuses on the classification and recognition of cognitive features using sequence of strokes. It is important to note that the sequence is also important for the recognition of strokes by doing contextual processing. The computed position coded visual feature set V_F is used to classify sequence of strokes. Hence, the strokes sequence $S = (s_1 s_2 s_3 \dots s_m)$ is recognized given visual features set $V_F = (v_{f_1}, v_{f_2}, v_{f_3}, \dots, v_{f_n})$. The objective is to recognize the strokes sequence S using computed visual features V_F , therefore Equation 7.1 will be

$$P(S|V_F) = P(s_1 s_2 s_3 \dots s_m | v_{f_1} v_{f_2} v_{f_3} \dots v_{f_n}) \quad (7.1)$$

By applying Bayes rule Equation 7.1 can be written as

$$P(S|V_F) = \frac{P(V_F|S) \cdot P(S)}{P(V_F)} \quad (7.2)$$

To determine maximal probable stroke sequence of a character from multiple possibilities, Equation 7.2 becomes

$$S = \underset{S \in C}{\operatorname{argmax}} \frac{P(V_F|S) \cdot P(S)}{P(V_F)} \quad (7.3)$$

Where C represent the character which belongs to the Urdu character set and have sequence of strokes S . Equation 7.3 is simplified using the fact that $P(V_F)$ is common in all possible stroke sequences, therefore it can be ignored to have Equation 7.4.

$$S = \underset{S \in C}{\operatorname{argmax}} P(V_F|S) \cdot P(S) \quad (7.4)$$

The computation of S is not straight forward, therefore $P(S)$ is simplified by

taking the Markov assumption that the stroke is only dependent on its previous stroke (bigram probability). In addition, $P(V_F|S)$ can also be simplified by assuming that each visual feature V_{f_i} in feature set V_F is dependent only on its corresponding stroke s_i . Therefore, Equation 7.4 can be approximated as Equation 7.5

$$S = \underset{S \in C}{\operatorname{argmax}} \prod_{i=1}^n P(v_{f_i}|s_i) \cdot P(s_i|s_{i-1}) \quad (7.5)$$

The visual feature set $V_F = (v_{f_1}, v_{f_2}, v_{f_3}, \dots, v_{f_n})$ and the sequence of strokes labels i.e. $S = (s_1 s_2 s_3 \dots s_m)$ are used to train the model. The model recognizes top k Strokes' Sequences $S_{StrokesSeq} = (S_1, S_2, S_3, \dots, S_n)$. To handle the feedback to this stage, the top k ranked list of sequences of strokes i.e. $S_{StrokesSeq}$ is forwarded to the next stage i.e. Character/Ligature Demon.

7.3.3 Character/Ligature Demon Formulation

At this stage, the recognized sequence of strokes i.e. $S = s_1 s_2 s_3 \dots s_m$ is computationally processed to recognize sequence of characters i.e. $C = c_1 c_2 c_3 \dots c_n$ of a ligature (**Lig**). Therefore, the next step is to recognize the character sequence i.e. $C = c_1 c_2 c_3 \dots c_n$ which from recognized stroke sequence $S = s_1 s_2 s_3 \dots s_m$, where n represents number of characters and m represents the number of strokes. Therefore Equation 7.6 will be

$$P(C|S) = P(c_1 c_2 c_3 \dots c_n | s_1 s_2 s_3 \dots s_m) \quad (7.6)$$

This equation also represents that m number of strokes can be assigned to n number of characters of a ligature. Equation 7.6 is rewritten by applying Bayes rule as Equation 7.7.

$$P(C|S) = \frac{P(S|C) \cdot P(C)}{P(S)} \quad (7.7)$$

Equation 7.7 is further generalized to generate maximal probable character sequence of a ligature (**Lig**) to have Equation 7.8

$$P(C|S) = \underset{C \in Lig}{argmax} \frac{P(S|C) \cdot P(C)}{P(S)} \quad (7.8)$$

$P(S)$ remains constant for all possible character sequences therefore it can be ignored to have Equation 7.9

$$P(C|S) = \underset{C \in Lig}{argmax} P(S|C) \cdot P(C) \quad (7.9)$$

Where

$$P(S|C) = P(s_1 s_2 s_3 \dots s_m | c_1^n)$$

$$P(S|C) = \prod_{i=1}^m P(s_i | c_1^n s_1^{i-1}) \quad (7.10)$$

$P(S|C)$ is simplified by taking the Markov assumption that stroke s_i only depends on the previous stroke s_{i-1} i.e.

$$P(S|C) = \prod_{i=1}^m P(s_i | c_1^n s_{i-1}) \quad (7.11)$$

in addition, another assumption is taken that s_i only depends on its character instead of whole character sequence. Therefore Equation 7.11 can be written as Equation 7.12

$$P(S|C) = \prod_{i=1}^m P(s_i | c_k s_{i-1}) \quad (7.12)$$

Since we take assumption that s_i depends on c_k , a character in which s_i appears, it gives always value of 1 and does not contribute in Equation 7.12, therefore Equation 7.13 is formulated.

$$P(S|C) = \prod_{i=1}^m P(s_i | s_{i-1}) \quad (7.13)$$

Now solving $P(C)$ of Equation 7.9 we have

$$P(C) = P(c_1) \times P(c_2|c_1) \times P(c_3|c_1^2) \times \dots \times P(c_n|c_1^{n-1}) \quad (7.14)$$

$$P(C) = \prod_{j=1}^n P(c_j|c_1^{j-1}) \quad (7.15)$$

Now using Markov assumption that probability of a character depends only on the previous character which allows Equation 7.15 to be represented as Equation 7.16

$$P(C) = \prod_{j=1}^n P(c_j|c_{j-1}) \quad (7.16)$$

Now putting values of Equation 7.13 and Equation 7.16 into Equation 7.9, we have

$$P(C|S) = \underset{c_1^n \in Lig}{argmax} \left(\prod_{i=1}^m P(s_i|s_{i-1}) \right) \times \left(\prod_{j=1}^n P(c_j|c_{j-1}) \right) \quad (7.17)$$

Equation 7.17 gives the maximum probable character sequence among all alternative character sequences in the ligatures set i.e. *Lig*.

where

$P(c_j|c_{j-1})$ and $P(s_i|s_{i-1})$ are estimated character bigram and stroke bigram probabilities, calculated using Equation 7.18 and Equation 7.19 respectively.

$$P(c_j|c_{j-1}) = \frac{Count(c_{j-1}c_j)}{Count(c_{j-1})} \quad (7.18)$$

$$P(s_i|s_{i-1}) = \frac{Count(s_{i-1}s_i)}{Count(s_{i-1})} \quad (7.19)$$

To compute these probabilities, three types of resources are required; (1) character corpus having sequences of characters of a ligature, (2) strokes corpus having sequence of strokes of characters and (3) character lexicon of characters having sequence of strokes. Same as third demon of reading, the contextual processing

using character's lexicon, strokes model and character model is formulated to recognize the best sequence of characters i.e. ligature (Lig).

7.4 Conclusion

In this chapter, a novel framework of the character recognition is proposed which is inspired by cognitive model of reading. All the phases of cognitive model of reading are formulated separately to develop equivalent model of Urdu character recognition. At first demon the 2-dimensional image is processed to generate the position coded visual feature same as first demon of cognitive model. In second demon, human brain recognizes the cognitive features i.e. sequence of strokes. In the same way the visual features are modeled to have sequences of strokes in Stroke Demon. In the third demon of reading model, the characters are recognized from sequence of strokes by doing contextual processing. The stroke model and character model are generated for strokes corpus and characters corpus respectively. Same as third demon of reading, the contextual processing using character's lexicon, strokes model and character model is formulated to recognize the best sequence of characters i.e. ligature (Lig). In the next chapter, the proposed mathematical model is used to develop the system.

Chapter 8

Development of Cognitive-inspired Framework for Urdu Character-based Recognition

8.1 Introduction

In Chapter 7, the cognitive inspired character recognition framework is presented along with mathematical formulation of each demon. This formulation works for the character recognition of text of any language belongs to Arabic script. Up till now, no research exists in the literature to recognize the Urdu letters using the cognitive model of reading. This cognitive based character recognition framework is aimed to develop a robust character recognition system to handle complex contextual character shaping and to resolve character insertion and deletion issues. According to the cognitive model of reading [91][84], the cognitive features (i.e. sequence of strokes) are processed from iconic input. These cognitive features are used to recognize the characters by using contextual information so that correct words can be recognized. In the same way, in Arabic writing style specifically

in Nastalique, each character is written by following the special rules of calligraphy. The sequence of strokes is used to write the characters. In Nastalique, each character has many contextual character shapes based on the position in the ligature.

The complete perceptual experiment has been conducted on complete character set having all 21 Urdu characters' RASM classes. The response of the participants for each character is analyzed. After detailed analysis of results, the strokes are categorized into three categories (1) primary strokes which represent core shape of the character, do not change its shape in different contexts and play significant role for the recognition of characters, (2) secondary strokes which do not play significant role for character identification, however, along with the primary strokes improve the confidence for the recognition of the character and (3) connector strokes which give only contextual positioning information of the character. The perceptual study of Urdu Nastalique character set drastically reduces the complexity of recognizing each character having multiple contextual shapes to the limited set of primary strokes.

Based on the findings of the perceptual experiment, a cognitive based computational framework is developed which will be used for the recognition of Arabic script languages. In this chapter, the implementation details of this framework are discussed by taking Urdu Nastalique ligature image as an input. The four different options are used to develop probabilistic model of character recognition, to see the significant of primary, secondary and connector for the recognition of a character. To test the applicability/strength of the framework, the image dataset of Urdu text images is used which is written using Nastalique writing style. The developed framework outperforms the state of the art Urdu recognition system.

As discussed in Chapter 7, the cognitive inspired framework for recognition of cursive script text has four demons given in Figure 8.1. The summary of each of demons are discussed here.

1. Image Demon: The input image stored as 2D array i.e. $I(x,y)$, is processed to compute the visual feature set i.e. V_F

2. Stroke Demon: Classification and recognition of the cognitive features i.e. sequence of strokes is performed using the visual features.
3. Character/Ligature Demon: The recognized sequence of strokes are statistically processed to recognize sequence of characters of a ligature.
4. Words Demon: Sequence of ligatures are processed to form the best sequence of words. This stage is same as the post processing phase of OCR. This is out of the scope of the current study.

The development details first three demons of framework are given in the subsequent sections.

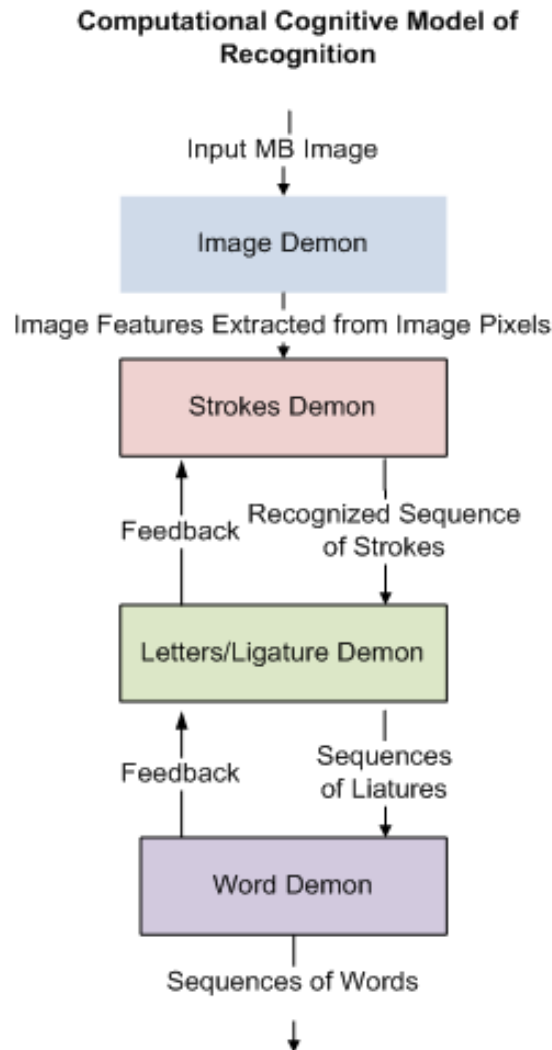


FIGURE 8.1: Architecture of cognitive based character recognition

8.2 Image Demon

Image demon processes the 2D image i.e. $I(x,y)$ and extracts the position coded visual features V_F so that cognitive model can be trained. The representation of 2D image in the form of position coded visual features is a challenging task. The input image of the system is the ligature image. To control the data size of the huge number of ligatures images, the ligature RASM is used to develop the system. The input ligature image is converted to the binarized form using binarization algorithm defined in [71]. The diacritics and RASM (main body) of the image is separated using dimensional features. The local stroke based windowing algorithm is used to traverse the RASM stroke to extract the position coded features according to the movement of QALAM. The process flow to extract position coded stroke features from the RASM image is given in Figure 8.2 with an example of image. The details of each step of this demon are given below.

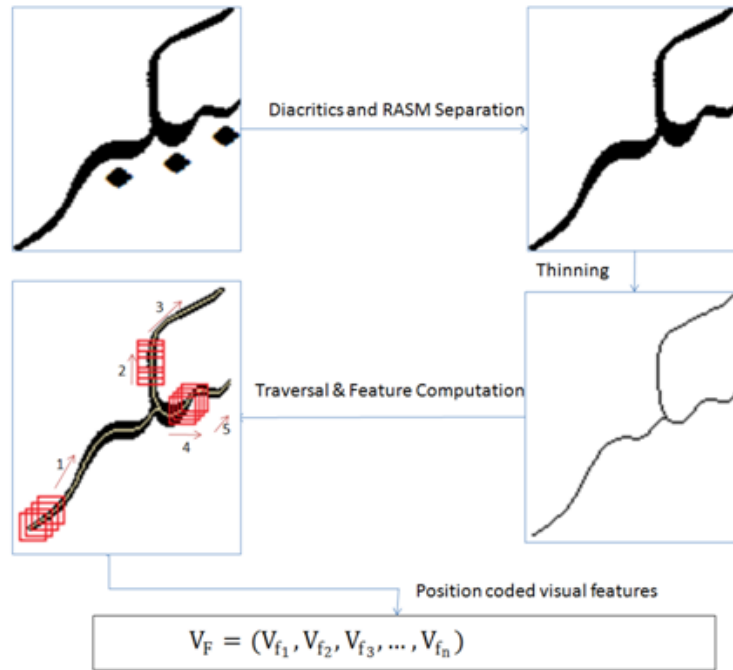


FIGURE 8.2: Position coded visual features extraction

8.2.1 Thinning

Thinning algorithm converts the RASM image into the skeletonized single pixel stroke. Before applying the thinning, salt and pepper noise is removed. This noise

causes the thinning algorithm to generate an incorrect thinned contour of the RASM. The thinning algorithm defined in [35] is applied on the RASM image to generate the thinned contour i.e. single pixel stroke, see Figure 8.2. This thinned contour will be used to traverse the window along the characters' strokes using consistent traversal.

8.2.2 Traversal

Traversal is the second step which will be used to slide the window so that (1) local stroke level features can be computed without covering characters overlapping context and (2) traversal should be in same order which human uses to move QALAM to write characters in ligatures. Using, the thinned contour, a consistent traversal algorithm [41] is applied to capture the position coded features of the main body stroke. The traversal algorithm traverses the ligature RASM in reverse order of its strokes sequence starting from the last stroke [41]. Therefore start point of the traversal is computed which is the last contour point of the last stroke of the last character. In Nastalique writing style, the movement of the pen to write the characters in ligature introduces junction points which need to be handled in consistent manner. Therefore, the priority rules are defined for the each direction of the branch contour to ensure the sequence of the strokes of a character is traversed in the same order as they are written. The thinned contour is overlayed on the actual stroke of the ligature image to apply the traversal algorithm. The centre point of the window is placed on the thinned contour such that the complete window should cover the stroke portion. The window is placed on the start point and the portion of the stroke, covered by the window is used to compute the features. In the same way, by applying the traversal rules, the complete stroke is traversed and window based features are computed and stored in sequence [41]. The traversal and windowing applied on the Urdu RASM is shown in Figure 8.2.

8.2.3 Computation of Features

In the third step, the discrete cosine transform (DCT) are computed from each window to extract the features. The DCTs performs well as compared to the structural and dimensional features [56]. The low frequency components of DCT

extract meaningful information from the windowed image. The DCTs features are computed from window of size $w \times w$. To ensure that the center point of the window is on thinned contours of the RASM, the window size is selected as an odd number which is $w=13$. This size is finalized by doing experiments to ensure that complete information of the stroke is computed even at extreme thick portion when complete QALAM is used to write the segment of the stroke. The window is placed on the RASM stroke by placing the center of the window on the thinned contour. The DCT features of the window placed at first contour point are computed and top-left three DCT coefficients are used as visual features v_{f_1} . The window is moved along the next contour point and DCT features from the respective window placed on RASM stroke are computed and stored as v_{f_2} , see Figure 8.2. The complete RASM stroke is traversed and the feature set $V_F = (v_{f_1}, v_{f_2}, v_{f_3}, \dots, v_{f_n})$ is computed. This approach of computing the features ensures the conversion of 2D image into the position-coded visual features.

8.3 Stroke Demon

At stroke demon, the cognitive features are recognized using the extracted features. This is important to note that the developed recognition system should recognize the strokes by doing contextual processing. Therefore the sequence of the strokes is important. The computed position coded visual feature set V_F is used to classify the cognitive features which are sequence of strokes. The complete classification and recognition system is developed which recognizes the sequence of cognitive features. The sequence of labels of strokes i.e. $S = (s_1 s_2 s_3 \dots s_m)$ along with the visual features i.e. $V_F = (v_{f_1}, v_{f_2}, v_{f_3}, \dots, v_{f_n})$ are used to classify so that these can be recognized using Equation 8.1

$$S = \underset{S \in C}{\operatorname{argmax}} \prod_{i=1}^n P(v_{f_i} | s_i) . P(s_i | s_{i-1}) \quad (8.1)$$

There are three phases to implement the stroke demon, the details are given below.

1. Investigation and labeling of character strokes of a ligature

2. Training and recognition

3. Improvements

8.3.1 Investigation and Labeling of Strokes

For the classification and recognition of strokes sequence, the first step is the labeling of the strokes of a ligature. The input to the system is ligature image which is composed of sequences of characters. The sequence of strokes of each character of the ligature is analyzed and unique shapes of strokes are extracted. Hence, the detailed analysis of ligatures inventory is carried out, unique shapes of each character strokes are extracted and transcribed. The analysis of strokes of Nastalique ligatures is carried out in an intelligent manner. First, single-letter ligatures are analyzed and unique shapes of character strokes are observed and their sequence of labels is transcribed. Then the ligatures of length two are investigated efficiently in such a way that all two letters ligatures are classified separately which have that specific analyzed character (or sequence of characters) as postfix character (or sequence of characters) of ligature. To further illustrate, ligatures such as جا، صا، سا، با are extracted separately as already analyzed postfix letter ا (ALEF) and جب، صب، سب، لب ligatures are separately listed as already analyzed letter ب (BEH) as postfix. As the $n-1$ length of ligature contexts have been analyzed e.g. ا (ALEF) and ب (BEH), therefore only remaining context i.e. strokes of first character need to analyze and write labels transcription accordingly. The sequence of strokes transcription of each ligature is defined. The analyzed context of ligatures of length $n-1$ is used to analyze the context of ligatures of length n . In the same way, the complete ligature RASM classes inventory is analyzed and the sequence of strokes of character labels' transcription is defined. As already discussed, the traversal of the ligature stroke starts from the last stroke of the last character and continues in reverse order, therefore the transcription of the strokes' labels is also defined in reverse order of writing as can be seen Table 8.1. Some examples of ligature transcription in terms of strokes labels are given in Table 8.1. Each stroke label is separated by space in transcription. Each Stroke (S) label is

defined as $S_char\#$ where, $char$ represents specific character having $\#$ as stroke number of that specific stroke S in $char$.

TABLE 8.1: Examples of Ligature Transcription in Terms of Strokes Sequences of Characters in Reverse Order

Ligature	Transcription
ا	S_{Alef1}
ب	$S_{Bay2} S_{Bay1}$
ج	$S_{Jeem4} S_{Jeem3} S_{Jeem2} S_{Jeem1}$
س	$S_{Seen4} S_{Seen3} S_{Seen2} S_{Seen1}$
ص	$S_{Swat4} S_{Swat3} S_{Swat2} S_{Swat1}$
ط	$S_{Toay3} S_{Toay2} S_{Toay1}$
ع	$S_{Aaeen4} S_{Aaeen3} S_{Aaeen2} S_{Aaeen1}$
ف	$S_{Fay2} S_{Fay1}$
ک	$S_{Kaf3} S_{Kaf2} S_{Kaf1}$
ق	$S_{Qaf2} S_{Qaf1}$
ل	$S_{Laam2} S_{Laam1}$
م	$S_{Meem2} S_{Meem1}$
ن	$S_{Noon2} S_{Noon1}$

8.3.2 Training and Recognition of Cognitive Features

The visual feature set $V_F = (v_{f_1}, v_{f_2}, v_{f_3}, \dots, v_{f_n})$ and the sequence of strokes labels i.e. $S = (s_1 s_2 s_3 \dots s_m)$ are used to train the HMM model using the Sphinx toolkit [101] which generates a trained HMM model i.e. $\lambda = (A, B, \pi)$. In the HMM model, A, B and π correspond to transition probabilities, emission probabilities and initial state probabilities, respectively. During training, the HMM uses the Baum Welch algorithm to maximize the visual feature sequence probabilities $P(V_F|\lambda)$ incrementally.

After training, the next step is to recognize the sequence of strokes from the input visual feature set of the RASM image. For this, the Viterbi-type beam search algorithm is used. This algorithm takes the feature set as input, uses the optimized HMM $\lambda = (A, B, \pi)$, and computes trigram based ranked list of highly probable

sequence of strokes. A corpus of sequences of strokes is generated so that probabilities $P(s_i|s_{i-1})$ can be computed. This corpus is generated by computing the ligatures' probabilities from the text corpus [4]. The sequence of strokes of characters in a ligature is repeated n times, where n is the frequency of the respective ligature in the text corpus. HMMs compute the prior probability i.e. $P(s_i|s_{i-1})$ of Equation 8.1 by using this corpus. The weighted prior probability is combined with the likelihood model of the visual feature sequence. The weight threshold of prior probability is set to 20 which is also computed experimentally. The visual feature set computed from the input ligature image i.e. V_F is fed to the trained HMM which recognizes top k strokes' sequences $S_{StrokesSeq} = (S_1, S_2, S_3, \dots, S_n)$. To handle the feedback to this stage, the top k ranked list of sequences of strokes i.e. $S_{StrokesSeq}$ is forwarded to the next stage i.e. Character/Ligature Demon.

8.3.3 Analysis and Improvements

After training and recognition, the recognition results are processed and confusions between the strokes are analyzed to further improve the stroke recognition results. The detailed analysis of stroke confusions reveals that some strokes of different characters have same shapes, as can be seen in Figure 8.3. To resolve this confusion, the same label is assigned to that stroke, see Table 8.2. The contextual recognition model of the strokes automatically ranks the respective stroke of the character by doing contextual processing. The strokes which share same shape across different characters (see Figure 8.3) are labeled as the S and all characters along with stroke numbers separated by underscore (-) are listed in postscript e.g. $S_{Bay2_Fay2_Kaf3}$ indicates that the second stroke of character Bay, the second strokes of character Fay and the third stroke of character Kaf are same.

8.4 Character/Ligature Demon

At this stage, the recognized sequence of strokes are computational processed to recognize sequence of characters of a ligature. Therefore the next step is to recognize the character sequence i.e. $C = c_1c_2c_3 \dots c_n$ from recognized stroke sequence $S = s_1s_2s_3 \dots s_m$, where n represents number of character and m represents the number of strokes. For this Equation 8.2 is used.


























Conf_1	Conf_2	Conf_3	Conf_4	Conf_5
 Bay_S2	 Fay_S2	 Kaf_S3		
 Jeem_S4	 Aaeen_S4			
 Seen_S3	 Swat_S3			
 Seen_S4	 Swat_S4	 Qaf_S2	 Laam_S2	 Noon_S2

FIGURE 8.3: Examples of Visual Confusion Between Strokes

TABLE 8.2: Ligature Transcription after Resolving Shape Confusions of Strokes

Ligature	Transcription
	S _{Alef1}
	S _{Bay2_Fay2_Kaf3} S _{Bay1}
	S _{Jeem4_Aaeen4} S _{Jeem3} S _{Jeem2} S _{Jeem1}
	S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Seen3_Swat3} S _{Seen2} S _{Seen1}
	S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Seen3_Swat3} S _{Swat2} S _{Swat1}
	S _{Toay3} S _{Toay2} S _{Toay1}
	S _{Jeem4_Aaeen4} S _{Aaeen3} S _{Aaeen2} S _{Aaeen1}
	S _{Bay2_Fay2_Kaf3} S _{Fay1}
	S _{Bay2_Fay2_Kaf3} S _{Kaf2} S _{Kaf1}
	S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Qaf1}
	S _{Laam2} S _{Laam1}
	S _{Meem2} S _{Meem1}
	S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Noon1}

$$P(C|S) = \underset{c_1^n \in Lig}{argmax} \left(\prod_{i=1}^m P(s_i | s_{i-1}) \right) \times \left(\prod_{j=1}^n P(c_j | c_{j-1}) \right) \quad (8.2)$$

Equation 8.2 gives the maximum probable character sequence among the all alternative character sequences in set *Lig.* where $P(c_j|c_{j-1})$ is estimated character bigram probabilities calculated using Equation 8.3.

$$P(c_j|c_{j-1}) = \frac{\text{Count}(c_{j-1}c_j)}{\text{Count}(c_{j-1})} \quad (8.3)$$

$P(s_i|s_{i-1})$ is estimated stroke bigram probabilities calculated using Equation 8.4.

$$P(s_i|s_{i-1}) = \frac{\text{Count}(s_{i-1}s_i)}{\text{Count}(s_{i-1})} \quad (8.4)$$

To compute these probabilities, two different corpora are required; (1) Character corpus and (2) Strokes corpus. In addition a character lexicon is also required to get the respective character against the sequence of strokes. These corpora and lexicons are generated by processing and analyzing all ligatures having single character, two characters, three characters and four characters in length. As majority of the character's contextual shapes are covered in ligatures having four characters therefore the presented models will recognize the ligatures upto eight characters. The details of development of these resources are given in the subsequent sections. For the understanding of the approach, examples of upto two characters ligatures are discussed.

8.4.1 Development of Probabilistic Model

In the section, the development of resources to develop the probabilistic model of character sequence using Equation 8.2 is discussed. The sequence of strokes is used to write Urdu characters. According to the findings of the cognitive experiment for Urdu characters, discussed in Chapter 4, there are three categories of strokes; (1) primary strokes, (2) secondary strokes and (3) connector strokes. The summary of the cognitive features classified into primary, secondary and connector strokes for the Urdu characters is given in Table 8.3.

To see the impact of these different categories of the strokes, four different probabilistic models are developed using four different options of strokes categories. The details of these models are given sub-sequent sections.

TABLE 8.3: Primary, Secondary and Connector Strokes of Urdu Characters

Char	Primary Strokes	Secondary Strokes	Connectors
ا	S _{Alef1}	—	—
ب	S _{BEH_1}	—	S _{BEH_2}
ج	S _{JEEM1} S _{JEEM2} OR S _{JEEM2} S _{JEEM3}	—	S _{JEEM4}
د	S _{DAL1}	S _{DAL2}	—
ر	S _{REH1}	S _{REH2}	—
س	S _{SEEN1} S _{SEEN2}	S _{SEEN3}	S _{SEEN4}
ص	S _{SWAT1} S _{SWAT2}	S _{SWAT3}	—
ط	S _{TOAY1} S _{TOAY2}	S _{TOAY3}	—
ع	S _{AAeen2}	S _{AAeen1} OR S _{AAeen3}	S _{AAeen4}
ف	S _{FAY1}	—	S _{FAY2}
ق	S _{QAF1}	—	S _{QAF2}
ک	S _{KAF1} S _{KAF2}	—	S _{KAF3}
ل	S _{LAAM1}	—	S _{LAAM2}
م	S _{MEEM1}	—	S _{MEEM2}
ن	S _{NOON1}	—	S _{NOON2}
و	S _{WAO1}	S _{WAO2}	—
ہ	S _{GHAY1} S _{GHAY2}	—	—
ء	S _{HAMZA2}	S _{HAMZA1} OR S _{HAMZA3}	—
ہ	S _{DCHAY1} S _{DCHAY2} S _{DCHAY3}		
ی	S _{CYHEY1} S _{CYHEY2}	—	S _{CYHEY3}
ے	S _{BYEY2}	S _{BYEY2}	—

1. M1: Primary, secondary and connector strokes are used to recognize the characters
2. M2: Primary and secondary strokes are used to recognize the characters
3. M3: Primary and connectors strokes are used to recognize the characters
4. M4: Primary strokes are used to recognize the characters

The character corpus, strokes corpus and character lexicon of each model are separately developed to build the probabilistic model.

8.4.2 Character Corpus Development and Probabilities Computations

The characters are joined together to form the ligature. The frequency of the ligatures is extracted from 37 million words corpus [4]. The characters sequence of a ligature is repeated textitn times where textitn is frequency of ligature. After developing this corpus, the character bigram probabilities are computed using formula given in Equation 8.3.

8.4.3 Strokes Corpus Development and Probabilities Computations

Strokes sequences of characters of a ligature are used to develop the strokes corpus. After developing this corpus, the strokes bigram probabilities, i.e. the occurrence of the stroke given the previous stroke, are computed. The strokes corpora for M1, M2, M3 and M4 are computed separately. To see the impact of the cognitive based strokes for character recognition, four different strokes corpora for M1, M2, M3 and M4 are generated. The strokes corpus of M1 contains primary, secondary and connector strokes sequence of characters. For M2 model, the primary and secondary strokes of characters are used to develop the strokes corpus. The M3 model which will show the impact of the primary strokes, the strokes corpus is developed by modifying the M1 in such a way that additional entries of character sequences of ligature are added having the primary and connector strokes. The M4 is developed to see the impact of only the primary strokes. Therefore, M3 model is modified by adding the strokes entries of primary strokes of character sequences. The sample entries of strokes corpora of M1, M2, M3 and M4 are given Table 8.4, Table 8.5, Table 8.6 and Table 8.7 respectively. For understanding of the strokes sequence for each ligature, the ligature against each strokes sequence entry is also given in each table. The actual strokes corpus only contains the sequence of strokes. After developing strokes corpora, the bigram stroke probability for each

strokes given the previous stroke, is computed using formula given in Equation 8.4.

TABLE 8.4: Sample Entries of Strokes Sequence Corpus for M1

Strokes Sequence of Ligature	Ligature
S _{Jeem4_Aaen4} S _{Jeem3} S _{Jeem2} S _{Jeem1}	ح
S _{Jeem4_Aaen4} S _{Aaen3} S _{Aaen2} S _{Aaen1}	ع
S _{BEH2_Fay2_Kaf3} S _{BEH1} S _{2_Fay_With_BEH} S _{Fay1}	فب
S _{BEH2_Fay2_Kaf3} S _{BEH1} S _{3_Swat_With_BEH} S _{Swat2} S _{Swat1}	صب
S _{DAL2_1} S _{DAL1_1} S _{3_Swat_With_DAL} S _{Swat2} S _{Swat1}	صد
S _{REH2_1} S _{REH1_1} S _{3_Swat_With_REH} S _{Swat2} S _{Swat1}	صر
S _{DAL2_1} S _{DAL1_1} S _{2_FAY_With_DAL} S _{Fay1}	فد
S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Seen3_Swat3} S _{Seen2} S _{Seen1} S _{2_BEH_With_SEEN} S _{BEH1_1}	بس

TABLE 8.5: Sample Entries of Strokes Sequence Corpus for M2

Strokes Sequence of Ligature	Ligature
S _{Jeem3} S _{Jeem2} S _{Jeem1}	ح
S _{Aaen3} S _{Aaen2} S _{Aaen1}	ع
S _{BEH1} S _{Fay1}	فب
S _{BEH1} S _{Swat2} S _{Swat1}	صب
S _{DAL2_1} S _{DAL1_1} S _{Swat2} S _{Swat1}	صد
S _{REH2_1} S _{REH1_1} S _{Swat2} S _{Swat1}	صر
S _{DAL2_1} S _{DAL1_1} S _{Fay1}	فد
S _{Seen3_Swat3} S _{Seen2} S _{Seen1} S _{BEH1_1}	بس

8.4.4 Lexicon Development

The sequence of strokes are joined to form the character. The character lexicon is required to get the respective character against the sequence of strokes. Therefore the character lexicon is developed which contains sequence of strokes along with the respective character. The lexicon is extracted from the transcription of the ligature in terms of the sequence of strokes of characters. Different lexicons for different models are developed separately. For M1, the sequence of all categories of strokes i.e. primary, secondary and connector are used to recognize the character. The

TABLE 8.6: Sample Entries of Strokes Sequence Corpus for M3

Strokes Sequence of Ligature	Ligature
S _{Jeem4_Aaeen4} S _{Jeem2} S _{Jeem1}	ح
S _{Jeem4_Aaeen4} S _{Jeem3} S _{Jeem2}	ح
S _{Jeem4_Aaeen4} S _{Jeem3} S _{Jeem2} S _{Jeem1}	ح
S _{Jeem4_Aaeen4} S _{Aaeen2}	ع
S _{Jeem4_Aaeen4} S _{Aaeen2} S _{Aaeen1}	ع
S _{Jeem4_Aaeen4} S _{Aaeen3} S _{Aaeen2}	ع
S _{BEH2_Fay2_Kaf3} S _{BEH1} S _{2_Fay_With_BEH} S _{Fay1}	فب
S _{BEH2_Fay2_Kaf3} S _{BEH1} S _{3_Swat_With_BEH} S _{Swat2} S _{Swat1}	صب
S _{DAL1_1} S _{3_Swat_With_DAL} S _{Swat2} S _{Swat1}	صد
S _{DAL2_1} S _{DAL1_1} S _{3_Swat_With_DAL} S _{Swat2} S _{Swat1}	صد
S _{REH1_1} S _{3_Swat_With_REH} S _{Swat2} S _{Swat1}	صر
S _{REH2_1} S _{REH1_1} S _{3_Swat_With_REH} S _{Swat2} S _{Swat1}	صر
S _{DAL1_1} S _{2_FAY_With_DAL} S _{Fay1}	فد
S _{DAL2_1} S _{DAL1_1} S _{2_FAY_With_DAL} S _{Fay1}	فد
S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Seen2} S _{Seen1} S _{2_BEH_With_SEEN} S _{BEH1_1}	بس
S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Seen3_Swat3} S _{Seen2} S _{Seen1} S _{2_BEH_With_SEEN} S _{BEH1_1}	بس

lexicon for M2 contains the sequence of primary and secondary strokes along with respective characters. The M3 model is based on M1 models having additional entries of sequence of primary strokes and connectors for respective characters. In the same way, the M4 model is based on the M2 having additional entries of primary strokes of the respective characters. The some lexical entries for M1, M2, M3 and M4 models are given in Table 8.8, Table 8.9, Table 8.10 and Table 8.11 respectively.

The algorithm starts by joining a recognized stroke with the previous with two possibilities; (1) current stroke is part of the previous sequence of strokes and (2) current stroke is first stroke of the next character. This algorithm computes the correct sequence of characters by building the binary tree. The first stroke is added as the root node of the tree. The second stroke is attached with first strokes

TABLE 8.7: Sample Entries of Strokes Sequence Corpus for M4

Strokes Sequence of Ligature	Ligature
S _{Jeem2} S _{Jeem1}	ج
S _{Jeem3} S _{Jeem2}	ج
S _{Jeem3} S _{Jeem2} S _{Jeem1}	ج
S _{Aaeen2}	ع
S _{Aaeen2} S _{Aaeen1}	ع
S _{Aaeen3} S _{Aaeen2}	ع
S _{BEH1} S _{Fay1}	فب
S _{BEH1} S _{Swat2} S _{Swat1}	صب
S _{DAL1.1} S _{Swat2} S _{Swat1}	صد
S _{DAL2.1} S _{DAL1.1} S _{Swat2} S _{Swat1}	صد
S _{REH1.1} S _{Swat2} S _{Swat1}	صر
S _{REH2.1} S _{REH1.1} S _{Swat2} S _{Swat1}	صر
S _{DAL1.1} S _{Fay1}	فد
S _{DAL2.1} S _{DAL1.1} S _{Fay1}	فد
S _{Seen2} S _{Seen1} S _{BEH1.1}	بس
S _{Seen3} S _{Swat3} S _{Seen2} S _{Seen1} S _{BEH1.1}	بس

with two possibilities; (1) attached with the previous stroke i.e. stroke is part of previous character stroke sequence, by adding the left child of the root, and (2) strokes is start of next character stroke, by adding node to the right child. At each level, the node probability is calculated. In the same way, for the second stroke, four possibilities are generated. This can be better visualized as the tree, where each N^{th} level shows the attachment of N^{th} stroke with previous strokes with two possibilities already discussed, causing 2^{N-1} nodes at N^{th} level. Hence, the node probability is computed using the developed probabilistic model by computing the characters sequence probabilities and strokes sequence probabilities. To handle the feedback to the previous level and get next sequence of strokes to form the characters, top k recognized sequences of strokes are processed separately using probabilistic model. Then the next sequence of strokes is processed and sequence

TABLE 8.8: Character Lexicon for Model M1

Lexical Entries	Char
S _{Alef1}	ا
S _{BEH2_Fay2_Kaf3} S _{BEH1}	ب
S _{Jeem4_Aaen4} S _{Jeem3} S _{Jeem2} S _{Jeem1}	ج
S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Seen3_Swat3} S _{Seen2} S _{Seen1}	س
S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Seen3_Swat3} S _{Swat2} S _{Swat1}	ص
S _{Toay3} S _{Toay2} S _{Toay1}	ط
S _{Jeem4_Aaen4} S _{Aaen3} S _{Aaen2} S _{Aaen1}	ع
S _{BEH2_Fay2_Kaf3} S _{Fay1}	ف
S _{BEH2_Fay2_Kaf3} S _{Kaf2} S _{Kaf1}	ک
S _{2_Fay_With_BEH} S _{Fay1}	فا
S _{3_Swat_With_BEH} S _{Swat2} S _{Swat1}	ص
S _{3_Swat_With_DAL} S _{Swat2} S _{Swat1}	ص
S _{DAL2_1} S _{DAL1_1}	د
S _{3_Swat_With_REH} S _{Swat2} S _{Swat1}	ص
S _{REH2_1} S _{REH1_1}	ر
S _{2_FAY_With_DAL} S _{Fay1}	فا
S _{2_BEH_With_SEEN} S _{BEH1_1}	باب

of characters along with the probabilities is stored. In the same way, the k^{th} sequence of strokes is processed to generate the sequence of characters. The top n sequences of characters are ranked having highest probabilities.

8.5 Conclusion

In this chapter, the implementation details of the cognitive inspired character recognition are presented. To see the impact of the categories of the cognitive features, four different probabilistic models are developed based on four different options of contributions of the strokes to recognize the characters. In next chapter, the recognition results on test data are also presented.

TABLE 8.9: Character Lexicon for Model M2

Lexical Entries	Char
S _{Alef1}	ا
S _{BEH1}	ب
S _{Jeem3} S _{Jeem2} S _{Jeem1}	ج
S _{Seen3_Swat3} S _{Seen2} S _{Seen1}	س
S _{Seen3_Swat3} S _{Swat2} S _{Swat1}	ص
S _{Toay3} S _{Toay2} S _{Toay1}	ط
S _{Aaeen3} S _{Aaeen2} S _{Aaeen1}	ع
S _{Fay1}	ف
S _{Kaf2} S _{Kaf1}	ک
S _{Swat2} S _{Swat1}	ص
S _{DAL2.1} S _{DAL1.1}	د
S _{REH2.1} S _{REH1.1}	ر
S _{BEH1.1}	ب

TABLE 8.10: Character Lexicon for Model M3

Lexical Entries	Char
S _{Alef1}	ا
S _{BEH2_Fay2_Kaf3} S _{BEH1}	ب
S _{Jeem4_Aaen4} S _{Jeem2} S _{Jeem1}	ج
S _{Jeem4_Aaen4} S _{Jeem3} S _{Jeem2}	ج
S _{Jeem4_Aaen4} S _{Jeem3} S _{Jeem2} S _{Jeem1}	ج
S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Seen2} S _{Seen1}	س
S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Seen3_Swat3} S _{Seen2} S _{Seen1}	س
S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Swat2} S _{Swat1}	ص
S _{Seen4_Swat4_Qaf2_Laam2_Noon2} S _{Seen3_Swat3} S _{Swat2} S _{Swat1}	ص
S _{Toay2} S _{Toay1}	ط
S _{Toay3} S _{Toay2} S _{Toay1}	ط
S _{Jeem4_Aaen4} S _{Aaen2}	ع
S _{Jeem4_Aaen4} S _{Aaen2} S _{Aaen1}	ع
S _{Jeem4_Aaen4} S _{Aaen3} S _{Aaen2}	ع
S _{Jeem4_Aaen4} S _{Aaen3} S _{Aaen2} S _{Aaen1}	ع
S _{BEH2_Fay2_Kaf3} S _{Fay1}	ف
S _{BEH2_Fay2_Kaf3} S _{Kaf2} S _{Kaf1}	ک
S _{2_Fay_With_BEH} S _{Fay1}	فا
S _{3_Swat_With_BEH} S _{Swat2} S _{Swat1}	ص
S _{3_Swat_With_DAL} S _{Swat2} S _{Swat1}	ص
S _{DAL1.1}	د
S _{DAL2.1} S _{DAL1.1}	د
S _{3_Swat_With_REH} S _{Swat2} S _{Swat1}	ص
S _{REH1.1}	ر
S _{REH2.1} S _{REH1.1}	ر
S _{2_FAY_With_DAL} S _{Fay1}	فا
S _{2_BEH_With_SEEN} S _{BEH1.1}	با

TABLE 8.11: Character Lexicon For Model M4

Lexical Entries	Char
S _{Alef1}	ا
S _{BEH1}	ب
S _{Jeem2} S _{Jeem1}	ج
S _{Jeem3} S _{Jeem2}	ج
S _{Jeem3} S _{Jeem2} S _{Jeem1}	ج
S _{Seen2} S _{Seen1}	س
S _{Seen3} S _{Swat3} S _{Seen2} S _{Seen1}	س
S _{Swat2} S _{Swat1}	ص
S _{Seen3} S _{Swat3} S _{Swat2} S _{Swat1}	ص
S _{Toay2} S _{Toay1}	ط
S _{Toay3} S _{Toay2} S _{Toay1}	ط
S _{Aaen2}	ع
S _{Aaen2} S _{Aaen1}	ع
S _{Aaen3} S _{Aaen2}	ع
S _{Aaen3} S _{Aaen2} S _{Aaen1}	ع
S _{Kaf2} S _{Kaf1}	ک
S _{Fay1}	ف
S _{DAL1.1}	د
S _{DAL2.1} S _{DAL1.1}	د
S _{REH1.1}	ر
S _{REH2.1} S _{REH1.1}	ر
S _{BEH1.1}	ب

Chapter 9

Results and Discussion

9.1 Introduction

In Chapter 8, the implementation details of cognitive-inspired framework for Urdu character-based recognition are discussed. To see the impact of the each category of cognitive features, four different probabilistic model are developed to recognize the character sequence using recognize sequence of strokes. In this chapter, character recognition results are presented for each of the four different probabilistic models.

9.2 Dataset

To test the system, two datasets are used. The Dataset-1 and Dataset-2 are used to prepare the training and testing datasets. The instances of 2,850 high frequent RASM classes of 2,156,484 ligature instances computed from the 37 millions words corpus [4], are selected from Dataset-2. The selected main bodies cover 38,207 high frequency unique Urdu words. In addition, the real samples of the selected main body classes are also extracted from Dataset-2 to develop training and testing datasets. A total of 30 samples are gathered to train the system. Where available, 15 real samples of Dataset-1 and 15 synthesized samples of Dataset-2 are used. If insufficient number of real samples is available, additional synthesized samples are added to keep the total sample count to 30. In addition, at least 15 non-overlapping samples for each main body are also to test the system. Real samples

are preferred to be used for testing when insufficient real data is available to cover both training and testing.

9.3 Results on Dataset-1 and Dataset-2

The testing data is used to test the stroke demon for the classification and recognition of strokes of Urdu characters of a ligature. The recognition output of the stroke demon is passed to the character/ligature demon for the recognition of character sequence using probabilistic model. To see the impact of the categories of cognitive features of Urdu characters, four probabilistic models for the recognition of Urdu characters using the recognized stroke sequences are developed, as discussed in previous chapter. In this section, these four models are evaluated on test dataset. The character sequence recognition accuracy is computed for each of the ligature image. The details of the recognition results are given in subsequent sections.

9.3.1 Character Recognition Accuracy of M1

The strokes corpus of M1 is developed by using strokes sequences of characters of a ligature. The primary, secondary and connector strokes sequence of characters of ligature are repeated m times where m is the frequency of the characters computed from corpus [4]. The character lexicon is required to recognize the character against the specific sequence of strokes. This lexicon is extracted from the transcription of the ligature in terms of the sequence of strokes and their respective characters. For M1 probabilistic model, the sequence of all categories of strokes i.e. primary, secondary and connector are used to recognize the character. After development of the probabilistic model, recognized stroke sequences of the stroke demon are used to recognize the character sequence. The output of this demon is the ranked list of character sequences. The accuracy is computed as the desired sequence of characters is found in the ranked list. The character recognition accuracy tested on the test data is 72.70%. The detailed analysis of the results shows that in majority of the misrecognition scenarios, the misrecognition of the connector causes the error.

9.3.2 Character Recognition Accuracy of M2

The second probabilistic model i.e. M2 is developed using primary and secondary strokes. The strokes corpus is developed by using primary and secondary strokes sequences of characters of a ligature. All the connector strokes are removed from the strokes corpus of M1 to develop the strokes corpus for M2. The character lexicon of M2 is also developed by removing all the connector strokes entries from the character lexicon of M1. During recognition, the recognized strokes sequences of the stroke demon are processed to remove all the connector strokes. The recognized character sequences of the character demon are used to compute the character recognition accuracy. The character recognition accuracy of M2 is 97.45% which is drastically improved from the M1. As can be seen from the recognition results, primary and secondary strokes based character recognition model significantly improves the character recognition results. These results further highlight the strength by showing that connector strokes play less role for the recognition of characters. The connectors preserve the information of contextual connection shape. The many contextual connectors shaping add the complexity for recognition. The sequence of correct primary, correct secondary and incorrect connectors will not be able to recognize the respective character in M1.

9.3.3 Character Recognition Accuracy of M3

Strokes corpus and character lexicon for the probabilistic model of M3 is developed by modifying the strokes corpus of M1. The additional entries of the characters in M1 strokes corpus which have secondary stroke are added. In each additional entry of the respective character, the secondary strokes are removed so that if the secondary strokes are not recognized, then the model should be able to recognize the desired character sequences. The testing of the model is carried out on the testing data and recognized sequences of the character strokes are fed to M3 probabilistic model. The character recognition accuracy of the system is 73.10%. The recognition results of M3 are slightly improved from the M1 indicating that misrecognition of the secondary stroke has impact on the character recognition. However the M2 outperforms M1 and M3. This is because the misrecognition of

the connector strokes deteriorates the character recognition accuracy significantly.

9.3.4 Character Recognition Accuracy of M4

The resources i.e. strokes corpus and character lexicon for the last probabilistic model i.e. M4 are also developed separately. The strokes corpus and character lexicon of M2 is modified by adding additional entries of the characters which have secondary stroke. In each additional entries of the respective character, the secondary strokes are removed so that if the secondary strokes are not recognized, then the model should be able to recognize the desired character sequences. It is important to note that in this model the connector strokes are ignored and secondary strokes are treated as optional to recognize the characters. The testing of 15 instances of each of 2,850 RASM types, the character recognition accuracy is 98.01% which out performs the all the probabilistic models. Similar to the processing of human brain to recognize the text by focusing on core features of characters, the computational model of character recognition based on primary stroke also outperforms. The character recognition accuracy of each model is given in Table 9.1. The test data contains ligatures of multiple lengths. The ligature length wise character recognition accuracy of each model is given in Table 9.2. As can be seen in Table 9.2, upto eight characters ligatures are tested by each of the model.

The model M4 with 98.01% outperforms all the models and gives promising results. The details analysis of the errors is carried out. Some of the errors are due to some variation in image quality causing some traversal issues after thinning resulted in the false features computation. In addition, primary strokes of different characters e.g. ALEF and LAM are also confused. To further improve the recognition accuracy of the Sphinx based Stroke Demon, deep learning based approaches can be used. In addition, such type of confusions can also be resolved at Character Demon by applying contextual heuristics such as ALEF is a non-joiner which cannot appear at initial or medial position in ligature, and if that is recognized at these places then the recognition stroke will be LAAM not ALEF.

TABLE 9.1: Character Sequence Recognition Accuracy of M1, M2, M3 and M4

Models	Char Seq. Recognition Accuracy (%)
M1	72.70
M2	97.45
M3	73.10
M4	98.01

TABLE 9.2: Ligature Length Wise Character Sequence Recognition Accuracy of M1, M2, M3 and M4

Lig. Length	Instances	Unique Lig.	M1 Acc. (%)	M2 Acc. (%)	M3 Acc. (%)	M4 Acc. (%)
1	195	13	68.72	96.41	74.87	96.41
2	2265	151	75.41	98.68	76.78	99.16
3	9210	614	75.11	97.46	75.44	98.85
4	16860	1124	74.14	97.67	74.64	98.10
5	10665	711	69.85	97.44	69.96	97.62
6	3000	200	67.10	95.20	67.20	95.50
7	480	32	64.79	98.75	65.00	98.96
8	75	5	60.00	96.00	60.00	96.00

9.4 Results on Dataset-3

Based on the results, M4 model is selected for the recognition of characters. This cognitive framework is also tested on UPTI dataset to do the comparative study. This system is integrated as RASM classification and recognition module with the Urdu Nastalique OCR framework [7], highlighted with blue color in Figure 9.1, to test the text line images. A total of 1600 text line images are used to test the system.

The images are binarized using algorithm defined in [71] to convert the gray scale image into binarized format. The RASM and diacritics are disambiguated using dimensional features. The diacritics association information with their respective RASM is also maintained. The RASM classes as sequence of character classes

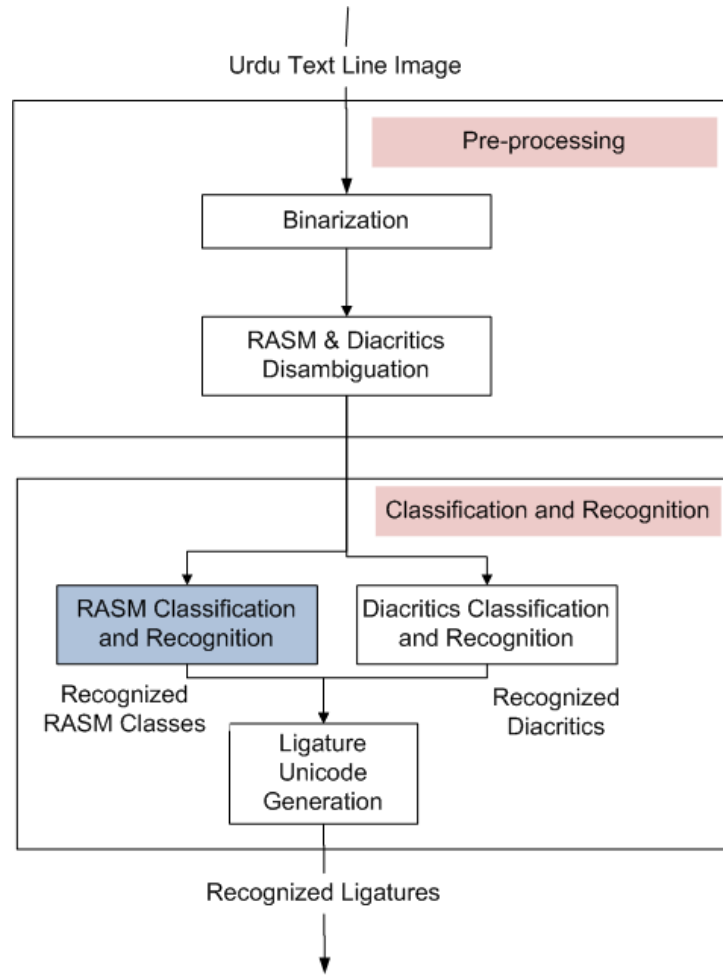


FIGURE 9.1: Process Flow of Integrated Urdu OCR Framework

Unicode are recognized using the presented system. The diacritics are recognized separately using different features based intensity and dimensional information. The recognized diacritics and RASMs are used to recognize the ligature Unicode. The text line images of UPTI dataset are tested using the integrated OCR framework system. The character recognition results of the proposed system along the comparisons with other techniques are given in Table 9.3.

TABLE 9.3: Comparison with State-of-the-art Urdu Recognition Techniques

Systems	Seg. Tech	Features	No. of Lines	Classifier	Acc (%)
Ul-Hassan et al.[98]	Implicit	Pixels	2003	BLSTM	94.85
Naz et al.[74]	Implicit	Statistical	1600	MDLSTM	94.97
Naz et al.[72]	Implicit	Statistical	1600	MDLSTM	96.4
Sabbour and Shafait [85]	Holistic	Contour	—	BLSTM	91
Naz et al.[75]	Implicit	Convolutional	1600	MDLSTM	98.12
Ahmad et al.[5]	Holistic	Pixels	1600	GBLSTM	96.71
Presented	Implicit	Cognitive features	1600	HMMs	98.44

Chapter 10

Conclusion

To develop the recognition systems, usually the dimensional, structural and geometrical features are extracted, termed as visual features, from document images using image processing techniques. These visual features along with labels are classified using state of the art machine learning and deep learning approaches. The conventional methods for the recognition of Urdu document images are classified into the two main categories (1) ligature-based classification and recognition and (2) character-based classification and recognition. Despite recognition of state of the art learning approaches of Urdu document images gives promising results, still practical use of developed systems have drawbacks. Urdu character set having 39 letters of Urdu constitutes around 25,000 commonly used unique ligatures. This ligature set is not closed because addition of new word in Urdu language which can be a transliterated word of foreign language, may cause addition of new ligature. Therefore the ligature based solution for the recognition of Urdu text is not appropriate. Whenever a new ligature would be added in the language, the system would require to be retrained on the additional ligature so that this can be recognized. To handle this issue, the character-based system seems to be an optimal solution. As Urdu character set is close set therefore addition of new ligature would not affect the performance of recognizer as new ligature will be segmented into characters which would be recognized by the recognizer. Both implicit character recognition and explicit character recognition techniques for Nastalique

writing style have some issues. The overlapping of characters and ligatures make the system more complex especially feature computation module. The horizontal sliding window is normally used to extract the features of the respective character. When a character overlaps with other character in a ligature, noisy features are computed for training and recognition of respective character and cause confusions for the machine learning system. In the same way, the recognition of the contextual character shaping is also a challenging task requiring significantly more training data to learn. Due to these challenges, the existing character recognition techniques generate errors by introducing character insertions and deletions in the recognized text.

According to the cognitive model of reading [91] and [84], the cognitive features (i.e. sequence of strokes) are processed from iconic input. These cognitive features are used to recognize the characters by using contextual information so that correct words can be recognized. In the same way, in Arabic writing style specifically in Nastalique, each character is written by following the special rules of calligraphy. The sequence of strokes is used to write the characters. In Nastalique, each character has many contextual character shapes based on the position in the ligature.

The complete perceptual experiment has been conducted on complete character set having all 21 Urdu characters' RASM classes. After detailed analysis of response of participants, the strokes are categorized into three categories (1) primary strokes which represent core shape of the character, do not change its shape in different contexts and play significant role for the recognition of characters, (2) secondary strokes which play less important role for character identification, however, along with the primary strokes improve the confidence for the recognition of the character and (3) connector strokes which give only contextual positioning information of the character. The perceptual experiment study of Urdu Nastalique character set drastically reduces the complexity of recognizing each character having multiple contextual shapes to the limited set of primary strokes.

Based on the findings of the perceptual experiment, a cognitive based computational framework is developed which will be used for the recognition of Arabic

script languages. The cognitive-inspired character recognition framework is presented along with mathematical formulation of each demon. This formulation starts with the recognition of strokes and later, characters are recognized using statistical model of characters and respective sequence of strokes. This technique can be used for character recognition of text of any language belongs to Arabic script. Up till now, no research exists in the literature to recognize the Urdu letters using the cognitive model of reading. This cognitive based character recognition framework is aimed to develop a robust character recognition system to handle complex contextual character shaping and to resolve character insertion and deletion issues.

The cognitive-inspired recognition framework based on cognitive features of Urdu is modeled to recognize the sequence of characters in a ligature. This technique resolves noisy feature computation due to the overlapping of characters in a ligature. In addition, the recognition of multiple contextual shapes is drastically reduced by focusing on recognized strokes of respective character. Later, these recognized strokes are weighted to further improve the character recognition. The recognition results of the system outperform the state of the art Urdu character recognition techniques, giving 98.44% character recognition accuracy. The technique is also tested on 1600 text lines of UPTI dataset. The recognition results show that this technique outperforms the state of the art Urdu character recognition techniques. The presented cognitive-inspired recognition framework improves the Urdu recognition accuracy tested on the printed Nastalique writing style. However, in future the same technique can be used to recognize Urdu handwritten images.

Author Bibliography

- **Akram, Q.** and Hussain, S. "Improving Urdu Recognition using Character-based Artistic Features of Nastalique Calligraphy", in IEEE Access, vol. 7, pp. 8495-8507, 2019.
- **Akram, Q.** and Hussain, S. "Ligature-based Font Size Independent OCR for Noori Nastalique Writing Style", in the Proceedings of 1st International Workshop on Arabic Script Analysis and Recognition (ASAR 2017), LORIA, Nancy, France, 2017.
- **Akram, Q.**, Niazi, A., Adeeba, F., Urooj, S., Hussain, S. and Shams, S. "A Comprehensive Image Dataset of Urdu Nastalique Document Images", in the Proceedings of Conference on Language and Technology 2016 (CLT 16), Lahore, Pakistan.
- Hussain, S., Ali, S. and **Akram, Q.** "Nastalique Segmentation-Based Approach for Urdu OCR", in International Journal on Document Analysis and Recognition (IJDAR), pp. 1-18, 2015.
- **Akram, Q.**, Hussain, S., Adeeba, F., Rehman, S. and Saeed, M. "Framework of Urdu Nastalique Optical Character Recognition System", in the Proceedings of Conference on Language and Technology 2014 (CLT 14), Karachi, Pakistan.
- Adeeba, F., **Akram, Q.**, Khalid, H. and Hussain, S. "CLE Urdu Books N-grams", poster presentation in Conference on Language and Technology 2014 (CLT 14), Karachi, Pakistan.

- **Akram, Q.**, Hussain, S., Niazi, A., Anjum, U., Irfan, F. "Adapting Tesseract for Complex Scripts: An Example for Urdu Nastalique", in the Proceedings of 11th IAPR Workshop on Document Analysis Systems (DAS 14) 2014, Tours, France.
- Naz, M., **Akram, Q.** and Hussain, S. "Binarization and its Evaluation for Urdu Nastalique Document Images", in the Proceedings of The 16th International Multi Topic Conference (INMIC) 2013, Lahore, Pakistan.

References

- [1] Nazir Tatjana A, Ben-Boutayab Nadia, Decoppet Nathalie, Deutsch Avital, and Frost Ram. Reading habits, perceptual learning, and recognition of printed words. *Brain and Language*, 88(3):294 – 311, 2004. ISSN 0093-934X. doi: [https://doi.org/10.1016/S0093-934X\(03\)00168-8](https://doi.org/10.1016/S0093-934X(03)00168-8). URL <http://www.sciencedirect.com/science/article/pii/S0093934X03001688>. Two hemispheres, one reading system: Early cortical representations of print during reading.
- [2] Husni A.Al-Muhtaseba, Sabri A.Mahmoud, and Rami S.Qahwajib. Recognition of off-line printed arabic text using hidden markov models. *Signal Processing*, 88(12):2902 – 2912, 2008. ISSN 0165-1684. doi: <https://doi.org/10.1016/j.sigpro.2008.06.013>. URL <http://www.sciencedirect.com/science/article/pii/S0165168408001928>.
- [3] S. Abirami, V. Essakiammal, and Ramachandran Baskaran. Statistical features based character recognition for offline handwritten tamil document images using hmm. *IJCVR*, 5:422–440, 2015.
- [4] Farah Adeeba, QuratulAin Akram, Hina Khalid, and Sarmad Hussain. Cle urdu books n-gram. In *Conference on Language and Technology(CLT-14)*, pages 81–88, 2014.
- [5] Ibrar Ahmad, Xiaojie Wang, Yuz hao Mao, Guang Liu, Haseeb Ahmad, and Rahat Ullah. Ligature based urdu nastaleeq sentence recognition using gated bidirectional long short term memory. *Cluster Computing*, Jun 2017. ISSN 1573-7543. doi: 10.1007/s10586-017-0990-5. URL <https://doi.org/10.1007/s10586-017-0990-5>.
- [6] Muaz Ahmad. Urdu optical character recognition system.

- [7] QuratulAin Akram, Sarmad Hussain, Farah Adeeba, Shafiqur Rahman, and Mehreen Saeed. Framework of urdu nastalique optical character recognition system. In *Conference on Language and Technology(CLT-14)*, 2014.
- [8] QuratulAin Akram, Sarmad Hussain, Aneeta Niazi, Umair Anjum, and Faheem Irfan. Adapting tesseract for complex scripts: An example for urdu nastalique. In *2014 11th IAPR International Workshop on Document Analysis Systems*, pages 191–195, April 2014. doi: 10.1109/DAS.2014.45.
- [9] QuratulAin Akram, Aneeta Niazi, Farah Adeeba, Saba Urooj, Sarmad Hussain, and Sana Shams. A comprehensive image dataset of urdu nastalique document images. In *Conference on Language and Technology(CLT-14)*, pages 81–88, 2014.
- [10] Jawad Hasan Yasin AlKhateeb, Jianmin Jiang, Jinchang Ren, Fouad Khelifi, and Stan Ipson. Multiclass classification of unconstrained handwritten arabic words using machine learning approaches. 2009.
- [11] Jawad Hasan Yasin AlKhateeb, Jinchang Ren, Jianmin Jiang, and Husni Al-Muhtaseb. Offline handwritten arabic cursive text recognition using hidden markov models and re-ranking. *Pattern Recognition Letters*, 32:1081–1088, 2011.
- [12] Alan Baddeley. The episodic buffer: a new component of working memory? *Trends in cognitive sciences*, 4(11):417–423, 2000.
- [13] Walter B. Barbe and Raymond H. Swassing. *Teaching Through Modality Strengths: Concepts and Practices*, page 117. Zaner Bloser, 1979. ISBN 978-0883091005.
- [14] Michael L. Bernard, Barbara S Chaparro, Melissa M. Mills, and Charles G. Halcomb. Comparing the effects of text size and format on the readability of computer-displayed times new roman and arial text. *Int. J. Hum.-Comput. Stud.*, 59(6):823–835, December 2003. ISSN 1071-5819. doi: 10.1016/S1071-5819(03)00121-6. URL [https://doi.org/10.1016/S1071-5819\(03\)00121-6](https://doi.org/10.1016/S1071-5819(03)00121-6).
- [15] C. Blakemore and F. W. Campbell. On the existence of neurones in the human visual system selectively sensitive to the orientation and size of retinal images. *The Journal of physiology*, 203:237–260, 1969. doi: doi:10.1113/jphysiol.1969.sp008862.

- [16] E. Darcy Burgund, Bradley L. Schlaggar, and Steven E. Petersen. Development of letter-specific processing: The effect of reading ability. *Acta Psychologica*, 122(1):99 – 108, 2006. ISSN 0001-6918. doi: <https://doi.org/10.1016/j.actpsy.2005.11.005>. URL <http://www.sciencedirect.com/science/article/pii/S0001691805001319>.
- [17] Neil R. Carlson, Harold L. Miller Jr., Donald S. Heth, John W. Donahoe, and G. Neil Martin. *Psychology: The Science of Behavior*, page 672. Pearson, 2009. ISBN 978-0205547869.
- [18] A. Cheung, M. Bennamoun, and N.W. Bergmann. An arabic optical character recognition system using recognition-based segmentation. *Pattern Recognition*, 34(2):215 – 233, 2001. ISSN 0031-3203. doi: [https://doi.org/10.1016/S0031-3203\(99\)00227-7](https://doi.org/10.1016/S0031-3203(99)00227-7). URL <http://www.sciencedirect.com/science/article/pii/S0031320399002277>.
- [19] Charles E Connor, Scott L Brincat, and Anitha Pasupathy. Transformation of shape information in the ventral pathway. *Current Opinion in Neurobiology*, 17(2):140 – 147, 2007. ISSN 0959-4388. doi: <https://doi.org/10.1016/j.conb.2007.03.002>. URL <http://www.sciencedirect.com/science/article/pii/S0959438807000347>. Cognitive neuroscience.
- [20] Burgund E. Darcy and Abernathy Alana E. Letter-specific processing in children and adults matched for reading level. *Acta Psychologica*, 129(1):66 – 71, 2008. ISSN 0001-6918. doi: <https://doi.org/10.1016/j.actpsy.2008.04.007>. URL <http://www.sciencedirect.com/science/article/pii/S0001691808000607>.
- [21] Iain Darroch, Joy Goodman-Deane, Stephen Anthony Brewster, and Philip D. Gray. The effect of age and font size on reading text on hand-held computers. In Maria Costabile and Fabio Paternò, editors, *Human-Computer Interaction - INTERACT 2005*, pages 253–266, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. ISBN 978-3-540-31722-7.
- [22] Krishnagopal Dharani. Chapter 3 - memory. In Krishnagopal Dharani, editor, *The Biology of Thought*, pages 53 – 74. Academic Press, San Diego, 2015. ISBN 978-0-12-800900-0. doi: <https://doi.org/10.1016/B978-0-12-800900-0.00003-8>. URL <http://www.sciencedirect.com/science/article/pii/B9780128009000000038>.

- [23] Daniel Fiset, Caroline Blais, Catherine Ethier-Majcher, Martin Arguin, Daniel Bub, and Frdric Gosselin. Features for identification of uppercase and lowercase letters. *Psychological Science*, 19:11611168, 2008.
- [24] L. H. Geyer and C. G. DeWald. Feature lists and confusion matrices. *Perception & Psychophysics*, 14(3):471–482, Oct 1973. ISSN 1532-5962. doi: 10.3758/BF03211185. URL <https://doi.org/10.3758/BF03211185>.
- [25] Denis G.Pelli, Catherine W. Burns, Bart Farell, and Deborah C. Moore. Feature detection and letter identification. *Vision Research*, 46(28):4646 – 4674, 2006. ISSN 0042-6989. doi: <https://doi.org/10.1016/j.visres.2006.04.023>. URL <http://www.sciencedirect.com/science/article/pii/S004269890600232X>.
- [26] Denis G.Pelli, Catherine W. Burns, Bart Farell, and Deborah C. Moore-Page. Feature detection and letter identification. *Vision research*, 46: 46464674, 2006.
- [27] William Grabe. *Reading in a Second Language: Moving from Theory to Practice*. Cambridge Applied Linguistics. Cambridge University Press, 2009. ISBN 9780521729741. URL <https://books.google.com.pk/books?id=prvRHZ7DrIcC>.
- [28] Jonathan Grainger. Cracking the orthographic code: An introduction. *Language and Cognitive Processes*, 23(1):1–35, 2008. ISSN 0169-0965.
- [29] Jonathan Grainger, Arnaud Rey, and Stephane Dufau. Letter perception: from pixels to pandemonium. *Trends in Cognitive Sciences*, 12(10): 381 – 387, 2008. ISSN 1364-6613. doi: <https://doi.org/10.1016/j.tics.2008.06.006>. URL <http://www.sciencedirect.com/science/article/pii/S1364661308001939>.
- [30] Alex Graves. *Supervised Sequence Labelling*, pages 5–13. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. ISBN 978-3-642-24797-2. doi: 10.1007/978-3-642-24797-2_2. URL https://doi.org/10.1007/978-3-642-24797-2_2.
- [31] Alex Graves, Marcus Liwicki, Santiago Fernandez, Roman Bertolami, Horst Bunke, and Jrgen Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis*

- and Machine Intelligence*, 31(5):855–868, May 2009. ISSN 0162-8828. doi: 10.1109/TPAMI.2008.137.
- [32] A. Karl Haberlandt. *Serial Recall*, pages 2265–2266. Springer, New York, 2011.
- [33] Payman Hamed. Famous calligraphers - persian calligraphy- all about persian calligraphy. <http://www.persiancalligraphy.org/Famous-Calligraphers.html>. (Accessed on 19/03/2018).
- [34] Kim Yeu Hong and Goetz Ernest T. *Children’s Use of Orthographic and Contextual Information in Word Recognition and Comprehension*, pages 205–249. Springer Netherlands, Dordrecht, 1995. ISBN 978-94-011-0385-5. doi: 10.1007/978-94-011-0385-5_7. URL https://doi.org/10.1007/978-94-011-0385-5_7.
- [35] Lei Huang, Genxun Wan, and Changping Liu. An improved parallel thinning algorithm. In *Proceedings of the Seventh International Conference on Document Analysis and Recognition - Volume 2, ICDAR ’03*, pages 780–, Washington, DC, USA, 2003. IEEE Computer Society. ISBN 0-7695-1960-1. URL <http://dl.acm.org/citation.cfm?id=938980.939541>.
- [36] Chih-Wei Hue and James R. Erickson. Short-term memory for chinese characters and radicals. *Memory and Cognition*, 16(3):196–205, 1988.
- [37] Sarmad Hussain. www.lict4d.asia/fonts/nafees_nastalique. In *12th AMIC Annual Conference on E-Worlds: Governments, Business and Civil Society*, Asian Media Information Center, 2003.
- [38] Sarmad Hussain. Letter-to-sound conversion for urdu text-to-speech system. In *Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages, Semitic ’04*, pages 74–79, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics. URL <http://dl.acm.org/citation.cfm?id=1621804.1621823>.
- [39] Sarmad Hussain and M. Afzal. Urdu computing standards: Urdu zabta takhti (uzt) 1.01. In *Proceedings. IEEE International Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century.*, pages 223–228, Dec 2001. doi: 10.1109/INMIC.2001.995341.

- [40] Sarmad Hussain, Shafiqur Rahman, Aamir Wali, Atif Gulzar, and Syed Jamilur Rahman. Grammatical analysis of nastalique writing style of urdu. Technical report, Center for Research in Urdu Language Processing, FAST-NU, 2002.
- [41] Sarmad Hussain, Salman Ali, and QuratulAin Akram. Nastalique segmentation-based approach for urdu ocr. *International Journal on Document Analysis and Recognition (IJDAR)*, 18(4):357–374, Dec 2015. ISSN 1433-2825. doi: 10.1007/s10032-015-0250-2. URL <https://doi.org/10.1007/s10032-015-0250-2>.
- [42] Madiha Ijaz and Sarmad Hussain. Corpus based urdu lexicon development. In *Conference on Language Technology (CLT07)*, 2007.
- [43] Tydgat Ilse and Grainger Jonathan. Serial position effects in the identification of letters, digits, and symbols. *Journal of Experimental Psychology: Human Perception and Performance*, 35(2):480–498, 2009. ISSN 1939-1277.
- [44] Biederman Irving. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94(2):115–147, 1987.
- [45] Arthur M. Jacobs, Tatjana A. Nazir, and Otto Heller. Perception of lowercase letters in peripheral vision: A discrimination matrix based on saccade latencies. *Perception & Psychophysics*, 46(1):95–102, Jan 1989. ISSN 1532-5962. doi: 10.3758/BF03208079. URL <https://doi.org/10.3758/BF03208079>.
- [46] Mike Jacobs. The science of word recognition. <https://docs.microsoft.com/en-us/typography/develop/word-recognition>, 10 2017. (Accessed on 8/01/2019).
- [47] Sobia Tariq Javed and Sarmad Hussain. Segmentation based urdu nastalique ocr. In José Ruiz-Shulcloper and Gabriella Sanniti di Baja, editors, *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 41–49, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-41827-3.
- [48] Sobia Tariq Javed and Sarmad Hussain. Segmentation based urdu nastalique ocr. In *18th Iberoamerican Congress on Pattern Recognition (CIARP 2013)*, 2013.

- [49] Sobia Tariq Javed, Sarmad Hussain, Ameera Maqbool, Samia Asloob, Sehrish Jamil, and Huma Mohsin. Segmentation free nastalique urdu ocr. *World Academy of Science, Engineering and Technology*, 46:456–461, 2010.
- [50] Falkenberg Helle K, Rubin Gary S, and Bex Peter J. Acuity, crowding, reading and fixation stability. *Vision Research*, 47(1):126 – 135, 2007. ISSN 0042-6989. doi: <https://doi.org/10.1016/j.visres.2006.09.014>. URL <http://www.sciencedirect.com/science/article/pii/S0042698906004366>.
- [51] Gideon Keren and Stan Baggen. Recognition models of alphanumeric characters. *Perception & Psychophysics*, 29(3):234–246, May 1981. ISSN 1532-5962. doi: 10.3758/BF03207290. URL <https://doi.org/10.3758/BF03207290>.
- [52] Naila Habib Khan, Awais Adnan, and Sadia Basar. Urdu ligature recognition using multi-level agglomerative hierarchical clustering. *Cluster Computing*, May 2017. ISSN 1573-7543. doi: 10.1007/s10586-017-0916-2. URL <https://doi.org/10.1007/s10586-017-0916-2>.
- [53] Mohammad S. Khorsheed. Offline recognition of omnifont arabic text using the hmm toolkit (htk). *Pattern Recognition Letters*, 28:1563–1571, 2007.
- [54] Paul A. Kirschner, John Sweller, and Richard E. Clark. Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching. *Educational Psychologist*, 41(2):75–86, 2006. doi: 10.1207/s15326985ep4102_1. URL https://doi.org/10.1207/s15326985ep4102_1.
- [55] Axel Larsen and Claus Bundesen. A template-matching pandemonium recognizes unconstrained handwritten characters with high accuracy. *Memory and Cognition*, 24(2):136–143, Mar 1996. ISSN 1532-5946. doi: 10.3758/BF03200876. URL <https://doi.org/10.3758/BF03200876>.
- [56] Gurpreet Singh Lehal and Ankur Rana. Recognition of nastalique urdu ligatures. In *Proceedings of the 4th International Workshop on Multilingual OCR, MOCR '13*, pages 7:1–7:5, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2114-3. doi: 10.1145/2505377.2505379. URL <http://doi.acm.org/10.1145/2505377.2505379>.

- [57] Willem J. M. Levelt, Ardi Roelofs, and Antje S. Meyer. A theory of lexical access in speech production. *Behavioral and Brain Research*, pages 1–38, 1999.
- [58] L. Lorigo and V. Govindaraju. Segmentation and pre-recognition of arabic handwriting. In *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*, pages 605–609 Vol. 2, Aug 2005. doi: 10.1109/ICDAR.2005.207.
- [59] Perea Manuel, Comesaña Montserrat, and Soares Ana P. Does the advantage of the upper part of words occur at the lexical level? *Memory and Cognition*, 40(8):1257–1265, Nov 2012. ISSN 1532-5946. doi: 10.3758/s13421-012-0219-z. URL <https://doi.org/10.3758/s13421-012-0219-z>.
- [60] Davis Mark and Lancu Laurenii. Unicode text segmentation. unicode text segmentation 29. Technical report, Unicode Consortium, 12 2016.
- [61] U.-V. Marti and H. Bunke. The iam-database: an english sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition*, 5(1):39–46, Nov 2002. ISSN 1433-2833. doi: 10.1007/s100320200071. URL <https://doi.org/10.1007/s100320200071>.
- [62] Judit Mate and Josep Baqus. Visual similarity at encoding and retrieval in an item recognition task. *The Quarterly Journal of Experimental Psychology*, 62(7):1277–1284, 2009.
- [63] Margaret W. Matlin. *Cognition*, page 640. John Wiley and Sons, New York, 2004. ISBN 978-0471450078.
- [64] James L. McClelland and David E. Rumelhart. An interactive activation model of context effects in letter perception: I. an account of basic findings. *Psychological Review*, 88(5):375–407, 1981. ISSN 1939-1471.
- [65] Diane McGuinness. *Language Development and Learning to Read: The Scientific Study of How Language Development Affects Reading Skill*, page 480. A Bradford Book, 2006. ISBN 978-0262633406.
- [66] Saul McLeod. Stages of memory encoding storage and retrieval. <https://www.simplypsychology.org/memory.html>, 2013. (Accessed on 07/01/2019).

- [67] Ramin Mehran, Hamed Pirsiavash, and Farbod Razzazi. A front-end ocr for omni-font persian/arabic cursive printed documents. *Digital Image Computing: Techniques and Applications (DICTA '05)*, pages 56–56, 2005.
- [68] Carol Bergfeld Mills and Linda J Weldon. Reading text from computer screens. *ACM Computing Surveys*, 19(4):329–357, 1987.
- [69] John V. Monaco and Charles C. Tappert. The partially observable hidden markov model and its application to keystroke dynamics. *Pattern Recognition*, 76:449 – 462, 2018. ISSN 0031-3203. doi: <https://doi.org/10.1016/j.patcog.2017.11.021>. URL <http://www.sciencedirect.com/science/article/pii/S0031320317304752>.
- [70] Saeed Mozaffari, Karim Faez, Farhad Faradji, Majid Ziaratban, and S. Mohammad Golzan. A comprehensive isolated farsi/arabic character database for handwritten ocr research. In *in Proceedings of the 10th international workshop on frontiers in*, pages 385–389, 2006.
- [71] Mamoon Naz, QuratulAin Akram, and Sarmad Hussain. Binarization and its evaluation for urdu nastalique document images. In *INMIC*, pages 213–218, Dec 2013. doi: 10.1109/INMIC.2013.6731352.
- [72] Saeeda Naz, Arif I. Umar, Riaz Ahmad, Saad B. Ahmed, Syed H. Shirazi, Imran Siddiqi, and Muhammad I. Razzak. Offline cursive urdu-nastaliq script recognition using multidimensional recurrent neural networks. *Neurocomputing*, 177:228 – 241, 2016. ISSN 0925-2312. doi: <https://doi.org/10.1016/j.neucom.2015.11.030>. URL <http://www.sciencedirect.com/science/article/pii/S092523121501749X>.
- [73] Saeeda Naz, Arif I. Umar, Riaz Ahmad, Muhammad Imran Razzak, Sheikh Faisal Rashid, and Faisal Shafait. Urdu nastaliq text recognition using implicit segmentation based on multi-dimensional long short term memory neural networks. In *SpringerPlus*, 2016.
- [74] Saeeda Naz, Arif I. Umar, Riaz Ahmad, Saad B. Ahmed, Syed H. Shirazi, and Muhammad I. Razzak. Urdu nastaliq text recognition system based on multi-dimensional recurrent neural network and statistical features. *Neural Computing and Applications*, 28(2):219–231, 2017.
- [75] Saeeda Naz, Arif I. Umar, Riaz Ahmad, Imran Siddiqi, Saad B Ahmed, Muhammad I. Razzak, and Faisal Shafait. Urdu nastaliq recognition

- using convolutionalrecursive deep learning. *Neurocomputing*, 243:80 – 87, 2017. ISSN 0925-2312. doi: <https://doi.org/10.1016/j.neucom.2017.02.081>. URL <http://www.sciencedirect.com/science/article/pii/S0925231217304654>.
- [76] Ulric Neisser. *Cognitive Psychology*. Appleton-Century-Crofts, 1967. URL <https://books.google.com.pk/books?id=0XH-GgAACAAJ>.
- [77] Donald G. Paterson and Miles A. Tinker. Studies of typographical factors influencing speed of reading: X style of typeface. *Journal of Applied Psychology*, 16:605613, 1932.
- [78] Mario Pechwitz, Samia Snoussi Maddouri, Volker Mrgner, Noureddine El-louze, and Hamid Amiri. Ifn/enit - database of handwritten arabic words. In *in Proceedings of CIFED*, 2002.
- [79] Denis G. Pelli, Bart Farell, and Deborah C. Moore. The remarkable inefficiency of word recognition. *Nature*, 423:752756, 2003.
- [80] Irfan Ahmad Qureshi. *Killaq Khursheed*. Calligraphers Association of Pakistan, 2019.
- [81] Ibrahim Raphiq, Eviatar Zohar, and Aharon-Peretz Judith. The characteristics of arabic orthography slow its processing. *Neuropsychology*, 16(3): 322–326, 2002.
- [82] R.C.Atkinson and R.M.Shiffrin. Human memory: A proposed system and its control processes. volume 2 of *Psychology of Learning and Motivation*, pages 89 – 195. Academic Press, 1968. doi: [https://doi.org/10.1016/S0079-7421\(08\)60422-3](https://doi.org/10.1016/S0079-7421(08)60422-3). URL <http://www.sciencedirect.com/science/article/pii/S0079742108604223>.
- [83] Partha Pratim Roy, Ayan Kumar Bhunia, Ayan Das, Prasenjit Dey, and Umapada Pal. Hmm-based indic handwritten word recognition using zone segmentation. *Pattern Recognition*, 60:1057 – 1075, 2016. ISSN 0031-3203. doi: <https://doi.org/10.1016/j.patcog.2016.04.012>. URL <http://www.sciencedirect.com/science/article/pii/S0031320316300450>.
- [84] David E Rumelhart and James L McClelland. An interactive activation model of context effects in letter perception, part i: An account of basic findings. *Psychological Review*, 88(5):375–407, 1981.

- [85] Nazly Sabbour and Faisal Shafait. A segmentation-free approach to arabic and urdu ocr. In *DRR XX*, 2013.
- [86] Reza Safabakhsh and Peyman Adibi. Nastaaligh handwritten word recognition using a continuous-density variable-duration hmm. 2005.
- [87] O. Samanta, U. Bhattacharya, and S.K. Parui. Smoothing of hmm parameters for efficient recognition of online handwriting. *Pattern Recognition*, 47(11):3614 – 3629, 2014. ISSN 0031-3203. doi: <https://doi.org/10.1016/j.patcog.2014.04.019>. URL <http://www.sciencedirect.com/science/article/pii/S0031320314001654>.
- [88] Naveen Sankaran and C. V. Jawahar. Recognition of printed devanagari text using blstm neural network. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pages 322–325, Nov 2012.
- [89] Thomas A. Sanocki. Looking for a structural network: Effects of changing size and style on letter recognition. *Perception*, 20(4):529–541, 1991.
- [90] Frederick M. Schwantes. Stimulus position functions in tachistoscopic identification tasks: Scanning, rehearsal, and order of report. *Perception & Psychophysics*, 23(3):219–226, May 1978. ISSN 1532-5962. doi: 10.3758/BF03204129. URL <https://doi.org/10.3758/BF03204129>.
- [91] O G Selfridge. Pandemonium: a paradigm for learning. In *Symposium on Mechanisation of Thought Processes*(Blake, D.V. and Uttley, A.M., eds), 1959.
- [92] Bikash Shaw, Swapan Kr. Parui, and Malayappan Shridhar. Offline handwritten devanagari word recognition: A segmentation based approach. In *2008 19th International Conference on Pattern Recognition*, pages 1–4, Dec 2008. doi: 10.1109/ICPR.2008.4761556.
- [93] Khalid Mehmood Siddiqui. *Qawaid-e-Nastalique Lahori*. Calligraphers Association of Pakistan, 2019.
- [94] Ray Smith, Daria Antonova, and Dar shyang Lee. Adapting the tesseract open source ocr engine for multilingual ocr. In *in International Workshop on Multilingual OCR*, pages 81–88, 2009.
- [95] Charles Stangor. *Introduction to Psychology*, page 480. Flat World Knowledge, 2006. ISBN 2940032497493.

- [96] Michal Stevens and Jonathan Grainger. Letter visibility and the viewing position effect in visual word recognition. *Perception & Psychophysics*, 65 (1):133–151, Jan 2003. ISSN 1532-5962. doi: 10.3758/BF03194790. URL <https://doi.org/10.3758/BF03194790>.
- [97] Thomas S. Tunis, Jennifer L Boynton, and Harry Hersh. Readability of fonts in the windows environment. In *Conference Companion on Human Factors in Computing Systems*, CHI '95, pages 127–128, New York, NY, USA, 1995. ACM. ISBN 0-89791-755-3. doi: 10.1145/223355.223463. URL <http://doi.acm.org/10.1145/223355.223463>.
- [98] Adnan Ul-Hasan, Saad Bin Ahmed, Sheikh Faisal Rashid, Faisal Shafait, and Thomas M Breuel. Offline printed urdu nastaleeq script recognition with bidirectional lstm networks. In *12th International Conference on Document Analysis and Recognition (ICDAR 2013)*, pages 1061–1065, 2013.
- [99] Saba Urooj, Sarmad Hussain, Farah Adeeba, Farhat Jabeen, and Rahila Parveen. Cle urdu digest corpus. In *Conference on Language and Technology (CLT-14)*, pages 47–53, 2014.
- [100] Aamir Wali and Sarmad Hussain. Context sensitive shape-substitution in nastaliq writing system: Analysis and formulation. In Tarek Sobh, editor, *Innovations and Advanced Techniques in Computer and Information Sciences and Engineering*, pages 53–58, Dordrecht, 2007. Springer Netherlands. ISBN 978-1-4020-6268-1.
- [101] Willie Walker, Paul Lamere, Philip Kwok, Bhiksha Raj, Rita Singh, Evan-dro Gouvea, Peter Wolf, and Joe Wlfel. Sphinx-4: A flexible open source framework for speech recognition. *Sun Microsystems*, 12 2004.
- [102] Inc. Wikimedia Foundation. Reading. https://en.wikipedia.org/wiki/Reading#cite_note-Certeau,_Michel_1984-1, . (Accessed on 06/01/2019).
- [103] Inc. Wikimedia Foundation. Naskh (script). [https://en.wikipedia.org/wiki/Naskh_\(script\)](https://en.wikipedia.org/wiki/Naskh_(script)), . (Accessed on 07/01/2019).
- [104] Inc. Wikimedia Foundation. Urdu alphabet. https://en.wikipedia.org/wiki/Urdu_alphabet, . (Accessed on 06/01/2019).

-
- [105] Inc. Wikimedia Foundation. Writing system. https://en.wikipedia.org/wiki/Writing_system, . (Accessed on 07/01/2019).
- [106] Inc. Wikimedia Foundation. Arabic diacritics. [https://en.wikipedia.org/wiki/Naskh_\(script\)](https://en.wikipedia.org/wiki/Naskh_(script)), 2014. (Accessed on 21/11/2014).
- [107] George Wolford and Samuel Hollingsworth. Retinal location and string position as important variables in visual information processing. *Perception & Psychophysics*, 16(3):437–442, May 1974. ISSN 1532-5962. doi: 10.3758/BF03198569. URL <https://doi.org/10.3758/BF03198569>.
- [108] Michael Zock. If you care to find what you are looking for, make an index : The case of lexical access. *ECTI Transaction on Computer and Information Technology*, 2(2):71–80, 2006.



FIGURE 10.1: Marked strokes of character SWAD in calligrapher's book [93]

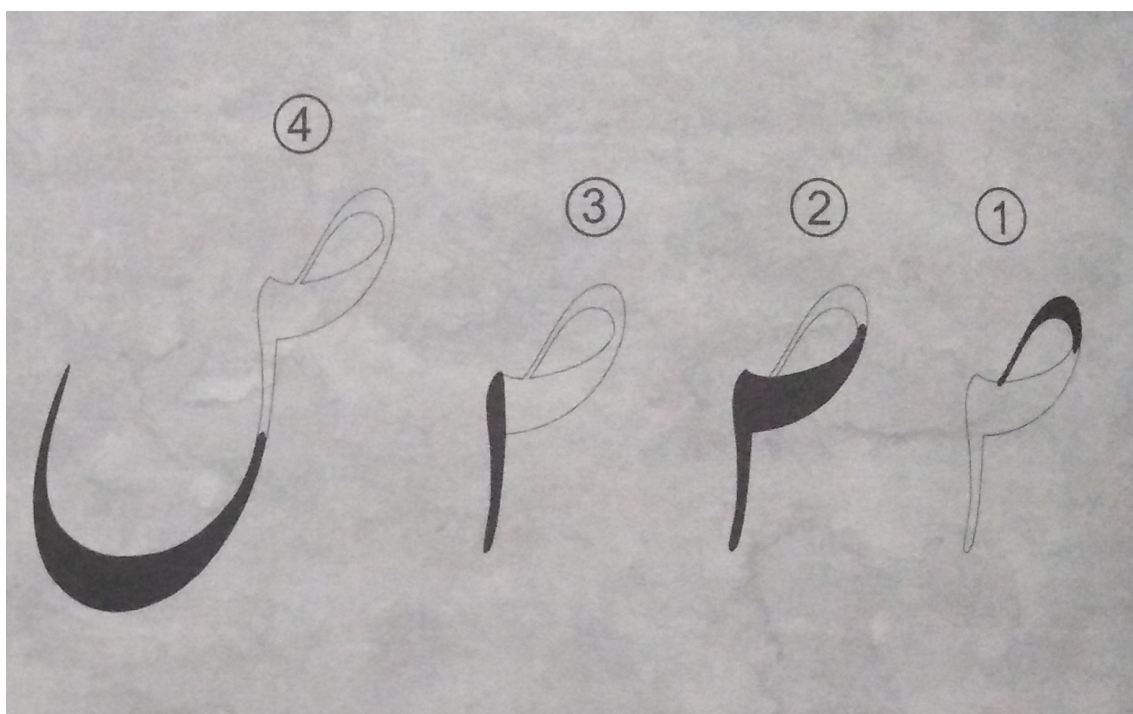
















































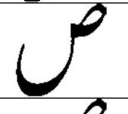


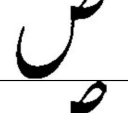
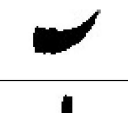
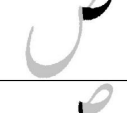



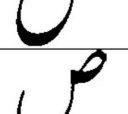
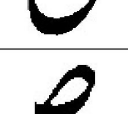

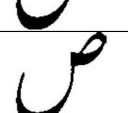


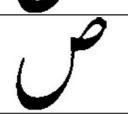
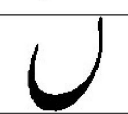

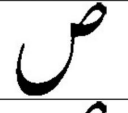








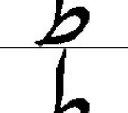


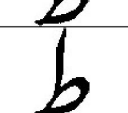
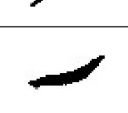

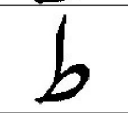
























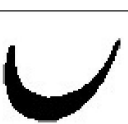








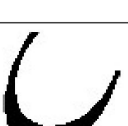









FIGURE 10.2: Marked strokes of character SWAD in calligrapher's book [80]


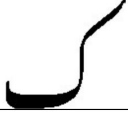
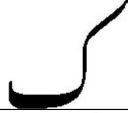




















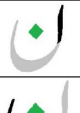











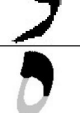









Characters	Stroke Sequence	Stroke Sequence Shape in Ligature	Strokes Sequence	Yes Votes	Acc.
			S ₁	9	100%
			S ₁	5	56%
			S ₂	9	100%
			S ₁ S ₂	9	100%
			S ₁	0	0%
			S ₂	5	56%
			S ₃	0	0%
			S ₄	0	0%
			S ₁ S ₂	9	100%
			S ₂ S ₃	9	100%
			S ₃ S ₄	3	33%
			S ₁ S ₂ S ₃	9	100%
			S ₂ S ₃ S ₄	9	100%
			S ₁ S ₂ S ₃ S ₄	9	100%
			S ₁	8	89%























































			S_2	2	22%
			$S_1 S_2$	9	100%
			S_1	2	22%
			S_2	8	89%
			$S_1 S_2$	9	100%
			S_1	0	0%
			S_2	0	0%
			S_3	0	0%
			S_4	0	0%
			$S_1 S_2$	9	100%
			$S_2 S_3$	2	22%
			$S_3 S_4$	2	22%
			$S_1 S_2 S_3$	9	100%
			$S_2 S_3 S_4$	4	44%













			$S_1 S_2 S_3 S_4$	9	100%
			S_1	1	11%
			S_2	1	11%
			S_3	0	0%
			S_4	0	0%
			$S_1 S_2$	9	100%
			$S_2 S_3$	5	56%
			$S_3 S_4$	0	0%
			$S_1 S_2 S_3$	9	100%
			$S_2 S_3 S_4$	3	33%
			$S_1 S_2 S_3 S_4$	9	100%
			S_1	0	0%
			S_2	1	11%
			S_3	1	11%
			$S_1 S_2$	9	100%

			S_2S_3	1	11%
			$S_1S_2S_3$	9	100%
			S_1	0	0%
			S_2	8	89%
			S_3	2	22%
			S_4	2	22%
			S_1S_2	9	100%
			S_2S_3	9	100%
			S_3S_4	2	22%
			$S_1S_2S_3$	9	100%
			$S_2S_3S_4$	9	100%

			$S_1 S_2 S_3 S_4$	9	100%
			S_1	9	100%
			S_2	4	44%
			$S_1 S_2$	9	100%
			S_1	8	89%
			S_2	3	33%
			$S_1 S_2$	9	100%
			S_1	5	56%
			S_2	1	11%
			S_3	4	44%
			$S_1 S_2$	9	100%
			$S_2 S_3$	7	78%

			$S_1S_2S_3$	9	100%
			S_1	1	11%
			S_2	1	11%
			S_1S_2	9	100%
			S_1	9	100%
			S_2	1	11%
			S_1S_2	9	100%
			S_1	9	100%
			S_2	6	67%
			S_1S_2	9	100%
			S_1	8	89%
			S_2	1	11%
			S_1S_2	9	100%
			S_1	3	33%
			S_2	2	22%

			S_1S_2	9	100%
			S_1	3	33%
			S_2	0	0%
			S_3	2	22%
			S_1S_2	2	22%
			S_2S_3	1	11%
			$S_1S_2S_3$	9	100%
			S_1	1	11%
			S_2	8	78%
			S_3	1	11%
			S_1S_2	9	100%
			S_2S_3	9	100%
			$S_1S_2S_3$	9	100%
			S_1	7	78%
			S_2	6	67%
			S_3	1	11%
			S_1S_2	9	100%
			S_2S_3	7	78%

			$S_1S_2S_3$	9	100%
			S_1	5	56%
			S_2	9	100%
			S_1S_2	9	100%