NAIST-IS-DD0961209

Doctoral Dissertation

Design and Development of Optimized Hygienic Input Systems for Touch Screen Gadgets

Asad Habib

September 17, 2012

Department of Information Processing
Graduate School of Information Science
Nara Institute of Science and Technology

Doctoral Dissertation
Submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Asad Habib

Thesis Committee:

Professor Yuji Matsumoto	(Supervisor)
Professor Ken'ichi Matsumoto	(Co-supervisor)
Associate Professor Masashi Shimbo	(Co-supervisor)
Project Associate Professor Masayuki Asahara	(Co-supervisor, National Institute for Japanese Language and Linguistics)

Design and Development of Optimized Hygienic Input Systems for Touch Screen Gadgets*

Asad Habib

Abstract

This PhD dissertation is multi-dimensional. We tried to find answers to three questions; 1) Are we using input systems having optimum performance? 2) Are these input systems suitable for touch-screen devices? 3) Are they 'hygienic'? In addition, we explored character level NLP (Natural Language Processing) applications. Urdu is used as case study language for experiments and evaluation of our proposed systems.

QWERTY keyboard migrated from typewriters to computers and then to mobile phones. It remained the most common mode of input for a long time. The multi-tap T9 keypads are also in use in mobile phones. However the recent touch screen gadgets boom changed the course of history. Having their own dynamics, they demand designing novel input systems with better usability, more user friendliness and higher performance. The generic QWERTY replica keypads lack these qualities when deployed on small devices like mobile phones, in particular the small touch screen gadgets such as smart phones and PDA (Personal Digital Assistant) etc.

RSI (Repetitive Strain Injuries), CTS (Carpel Tunnel Syndrome), CTD (Cumulative Trauma Disorder), ophthalmic endemics and eyesight weakness etc. are few among many health hazards caused by regularly improper and prolonged use of computers. RPA (Resting Point of Accommodation) and Convergence prospects are important design time considerations. Small screen cluttered with too many buttons and icons etc. puts more strain on eyes due to acute and meager visibility. We developed and proposed "hygienic" touch screen keypads that are free from the aforementioned shortcomings. We used "onion skinning" technique to create large enough buttons that put less strain on the eyes of the user. Character unigram and bigram frequency distributions and properties like shape of individual letters are used to facilitate fast, correct and easy composing. Large buttons and their arrangement on our proposed keypads free the user from bending his/her wrist on an angle that is risky for his/her health in the long run.

Touch screen gadgets come in various sizes, shapes, platforms and other specifications. Keeping in view the intrinsic parameters of touch screen devices and Urdu language, we proposed novel and distinct keypads for each of the small, medium and large touch-screen devices. Each of these has been optimized for accurate, easy, speedy and efficient typing. We carefully designed our proposed keypads such that they offer better visibility, usability, extendibility, aesthetics and user friendliness. We also put users' health into prime focus at design time of proposed keypad.

---

* Doctoral Dissertation, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD0961209, September 17, 2012

Composing Urdu is a thorny task on modern touch-screen devices. Design and development of optimal keypads for Urdu is complicated due to complex orthography and relatively large letter-set. Our proposed keypads are optimized for Urdu but applicable to Arabic, Persian, Punjabi and other Perso-Arabic script languages too. With minor changes in the backend script settings, our proposed keypads are applicable to non-Perso-Arabic script languages with larger letter-sets e.g. Hindi etc.

We carried out evaluation of our proposed keypads in two ways. Automated procedures showed improvement by 52.62% over in-the-market generic keypads for small screen gadgets. Medium size screen keypads showed 57.06% improvement. We conducted users' evaluation for real world performance comparison. Detailed results are presented in the subsequent chapters.

Keywords: touch-screen keypads, smart phones input, Urdu input method editor, hygienic input systems design, Perso-Arabic script input

タッチスクリーン機器に最適化された健康的な入力システムの設計と開発*

アサド　ハビブ

内容梗概

本稿ではタッチスクリーン機器に最適な入力システムについて，以下の3つの問いへの回答を通じて多面的に模索する．1)我々が使用している入力システムは最適なものだろうか．2)それらの入力システムはタッチスクリーン機器に適したものだろうか．3)それらは「健康的」だろうか．加えて，我々は文字レベルの自然言語処理アプリケーションについても検討した．本ケーススタディーにおける提案システムの実験と評価のための言語にはウルドゥー語を用いた．

QWERTYキーボードはタイプライターに由来し，コンピュータや携帯電話などにおける最も一般的な入力装置として現在も残っている．また，マルチタップT9キーパッドも携帯電話における入力装置としてよく使われている．だがこの流れは近年のタッチスクリーンガジェットブームによって変わってきている．そこで求められているのは，より使いやすく，よりユーザフレンドリで，より高速に入力できるよう設計された新たな入力システムである．一般的なQWERTYキーボードを携帯電話などの小さなデバイス，特にタッチスクリーン上に移植したキーパッドはこれらの性質を欠いており，とりわけスマートフォンやPDAといったタッチスクリーンの小さな機器では問題が大きい．

日常的かつ長時間にわたる不適切なコンピュータの使用によって引き起こされる障害は多く，RSI (反復運動過多損傷)，CTS (手根管症候群)，CTD (蓄積外傷性障害)，眼に関する風土病，視力低下などはその一例である．これらを防ぐためには，設計時にRPA (調節安静位)などを考慮することが重要である．小さなスクリーンに大量のボタンやアイコンなどが散乱していると，眼にとって負担が大きい．これに対して，我々が開発した「健康的」なタッチスクリーンキーパッドは，上記の欠点を克服し，高速・正確・容易な入力を実現する．我々が用いた「オニオンスキン」と呼ばれる技術を用いると，ユーザの眼にかける負担が少ない十分な大きさのボタンを配置することができる．文字ユニグラムとバイグラム頻度およ

*奈良先端科学技術大学院大学情報科学研究科情報処理学専攻博士論文，NAIST-IS-DD0961209, 2012 年 9 月 17 日

びそれぞれの文字の形などの性質を用いて，高速で，正確かつ簡単な編集を可能にした．大きなボタンと提案するキーパッドにおけるボタン配置によって，ユーザは長期的に健康的なリスクがあるような角度に手首を曲げなければならないことから解放される．

タッチスクリーン機器は，大きさ，形状，プラットフォームなどが様々である．そこで我々は，個々のタッチスクリーン機器に固有のこれらのパラメータと，ウルドゥー語固有のパラメータを考慮して, 三種類のスクリーンサイズ(大中小)それぞれのための新しいキーパッドを提案した．これらのキーパッドは，正確・容易・高速で，効率のよい入力ができるよう最適化されており，よりよい可視性，使いやすさ，拡張性，美しさ，ユーザフレンドリさを提供するよう慎重に設計されている．設計時には，ユーザの健康にも重点を置いた．

ウルドゥー語は正書法が複雑であり，アルファベットが比較的多いため，タッチスクリーン機器向けの最適なキーパッドの設計と開発には手間がかかる．提案するキーパッドはウルドゥー語に最適化されているが, アラビア語, ペルシャ語, パンジャビ語および, アラビア文字を用いるその他の言語にも適用できる. また, 文字関連の若干の修正によって，アラビア文字を用いない，よりアルファベットの多い言語(ヒンディー語など)にも適用できる．

我々は提案するキーパッドの評価を二つの方法で行った．自動評価では，市販されている小さなスクリーンの機器向けの一般的なキーパッドに対し52.62%の向上を示した．中サイズのスクリーン向けキーパッドでは57.06%の向上を示した．我々はまた，実世界での性能評価のためユーザ評価を行った．

キーワード

タッチスクリーンキーパッド，スマートフォン入力方式，ウルドゥー語入力方式，健康的入力システム設計，アラビア文字入力

Acknowledgements

I offer gratitude from the core of my heart to my advisor and mentor Professor Yuji Matsumoto for his rich in knowledge insights, guidance and never ending support. Sir, thank you very much! I feel highly honored and privileged to be your student and become a part of your lab. Words don’t suffice to convey you my gratitude for everything you have done for me.

Next, I offer my gratefulness to Associate Professor Masayuki Asahara for his kind help and supervision. Sir, thank you very much for pointing out minute and important details to improve my practical projects, research papers and final dissertation report.

I also thank other members of my Doctoral advisory committee; Professor Ken’ichi Matsumoto and Associate Professor Masashi Shimbo for their invaluable feedback and guidance on how to further improve the clarity of my dissertation.

I am deeply indebted to Dr. Masakazu Iwatate, my co-author, my best friend. It’s been a pleasure to work with you. I look forward for our combined research projects in future.

I would also like to thank Assistant Professor Mamoru Komachi and Assistant Professor Kevin Duh for their time, advices and in-depth discussions about improving my work.

Many thanks to Gakuseika staff, all the academic and management staff of NAIST and everyone who made my stay in Japan productive and useful. I also thank MEXT for providing me financial support in bringing this research from concept to reality.

“Kansha” to my Japanese language teachers. I would love to find ways through which I can stay in-touch with this wonderful language in future.

Big thank you to Ms. Fatima for playing active role in completion of this write-up. Your moral and logistic support has been a source of encouragement.

I owe special thanks to my Japanese homestay families in Tomio (Nara) and Kanazawa especially to my mother Mrs. Yoshida. You have been like cool shade in the noontide of life.

Last but not the least, I run out of words when I offer my gratitude to my family in Pakistan. They kept the storms at shore while I cruised on. Your encouragement, support, investment and patience etc.; all are second to none! Thank you very much for believing in me. I dedicate all my achievements to you.

Contents

List of Figures

List of Tables

List of Abbreviations

CL	Computational Linguistics
CPU	Central Processing Unit
CTD	Cumulative Trauma Disorders
CTS	Carpal Tunnel Syndrome
DM	Diacritical Marks
F-Exp	Frequency based layout Expectancy
FA	Finite Automata
FBL	Frequency Based Layout
GUI	Graphical User Interface
HCI	Human Computer Interaction
ICT	Information and Communications Technologies
IEOU	Input English Output Urdu
IME	Input Method Editor/ Environment
IPA	International Phonetic Alphabet
IPA	Intelligent Personal Assistant
IR	Information Retrieval
JaPak IEOU	Japan-Pakistan's Input English Output Urdu
KSPC	Key Strokes Per Character
KSPC-S	Keystrokes per Character on Standard layout
KSPC-F	Keystrokes per Character on Frequency based layout
KSPW	Key Strokes Per Word
MSD	Musculo-Skeletal Disorder
MT	Machine Translation
NASA	National Aeronautics and Space Administration
NER	Named Entity Resolution
NLA	National Language Authority
NLG	Natural Language Generation
NLP	Natural Language Processing
OCR	Optical Character Recognition
OSK	On-Screen Keyboard
PDA	Personal Digital Assistant

PEC	Proposed English Character
POS	Part Of Speech
PRL	Proposed Roman Letter
RPA	Resting Point of Accommodation
RPV	Resting Point of Vergence
RSI	Repetitive Strain Injuries
S-Exp	Standard layout Expectancy
SE	Sentiment Extraction
SMS	Short Message Service

Definitions

Optimize

To maximize efficiency and performance in execution, retrieval and storage etc. of a system.

Hygienic

Hygienic is a generic term that carries several inter-related meanings. In daily life conversation, it means sanitary and healthful. However, in this dissertation report, we used the word ‘hygienic’ in the meanings ‘conductive to good health’. In broader sense, hygienic systems refer to systems that minimize the risk of health hazards caused by regular and prolonged use of computer and its sister devices. Examples of such health hazards are RSI, CTS and eyesight weakness etc.

Input System

Any interface with suitable input method or GUI through which users can write text or enter commands to the underlying device or application software. Hardware keyboards, soft (virtual) keypads, GUIs of web/desktop applications and mobile phone interfaces are typical examples of input systems.

Touch Screen Gadgets

A device that recognizes and senses the ‘touch’ of finger or stylus and sends appropriate signal to a card called controller in order to translate user’s input into CPU readable format. Driver is required so that software can utilize the information from the touch. There are many varieties of touch screen systems varying in size, purpose and the underlying electronics etc. Smart phones, laptops, tablet PCs, cameras, GPS devices etc. all are examples of touch screen gadgets.

Alphabet

Alphabet is any system of signs with which a language is written. In this dissertation, we use it to represent the whole set of letters in a language e.g. in English language, all the letters from A-Z (both in small and upper case) are referred to as alphabet. For clarity and easier understanding, we also refer to alphabet as ‘letter-set’ at some points in this dissertation.

Letter

An element of alphabet is called a letter. Considering an example in the English language, the letter set {A, a, b, F, m} has 5 English letters.

Character

This is a general and broader term that represents all letters in an alphabet, numeric as well as special characters etc.

Script

Script refers to the continuous natural and native way of writing text. Ligatures, words, phrases and sentences are formed based on correct and appropriate shapes of individual letters. Collectively all of these are referred to as ‘script’ in this dissertation.

Chapter 1

1 Introduction

This research is multi-dimensional and covers more than one areas that are summarized in the following. My dissertation emphasizes on finding answers of the following questions.

1. Are we using the input systems that offer optimum performance?
2. Are these input systems suitable for touch-screen devices?
3. Are they ‘hygienic’?

In contrast to theoretical approach, our research work is mainly of practical nature. In addition to the above mentioned questions, we also explored various applications of character level NLP (Natural Language Processing) research. We proposed novel soft-keypads for the input systems of computers, mobile phones and other gadgets. Soft-keypads are also called virtual keypads or touch screen keypads. We used Urdu as the case study language for experiments and evaluation of our proposed systems.

The natural language processing is a vast area of research that has many branches, tasks and applications at different levels. The umbrella of NLP has spread so much that some of its branches became separate fields in their own right e.g. MT (Machine Translation), NLG (Natural Language Generation) and IR (Information Retrieval) etc. There are numerous levels of text and speech at which natural language processing research is carried out e.g. document, discourse, sentence, phrase, word and character level etc. This research is related to the applications of NLP at character level.

1.1 Character Level NLP applications

This research targets the ‘character level’ applications of natural language processing. Some important sub-fields that deal with natural language processing tasks at the ‘characters level’ include Script Generation, Romanization, Transliteration, Transcription and Development of IME (Input Method environment), different types of keypads and their GUI designs etc. These tasks are also illustrated in the following Figure 1.1.

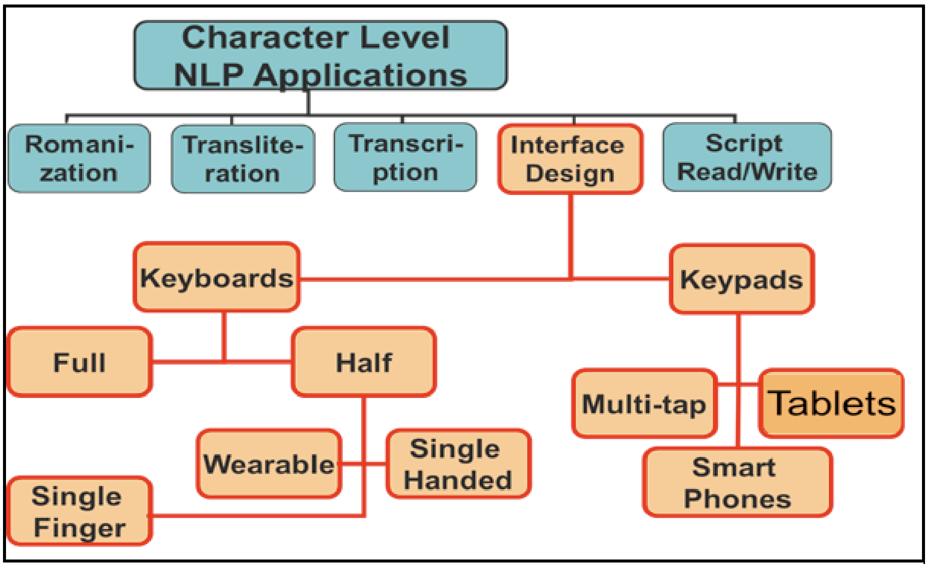


Figure 1.1 Character Level applications of NLP

The highlighted part in Figure 1.1 shows that we focused on interface design and development. We developed various types of keypads for small, medium and large size touch screen devices. The small size touch screen devices refer to smart phone, PDA (Personal Digital Assistant) and other small size handheld gadgets [168]. Large touch screen systems are those devices that have screens equal to or larger than a standard Personal Computer [146]. Acer ICONIA laptop is an example of large touch screen system [10]. Any device that falls between small and large size systems is referred to medium size touch screen device [11]. Typical examples of medium size touch screen systems are tablet PCs and e-readers etc.

1.2 Motivation

The growth of wireless networks, Grid and Cloud computing have enabled users to migrate from PCs with high end hardware specifications to smaller, lighter, low memory, low power and handy touch screen devices. As a result, we are witnessing the evolution and rapid growth of touch screen gadgets. The input systems of modern touch screen gadgets and other sister devices have great room for improvement in their Ergonomics i.e. performance, usability, user friendliness, safety (from user's health perspective) and aesthetics etc. Similarly improvements can be brought in areas such as reduction of error rate, compliance of device and the GUI deployed on it, appropriateness to the language and flexibility in user's choice etc.

The current keypads on touch screen devices perform slower and are prone to errors due to their poor usability. Speech powered IPA (Intelligent Personal Assistant) are also introduced lately but still far from users' satisfaction [104]. One such example is the IPA called "Siri" that has been introduced by Apple Inc. in their iOS 5.1.1 for iPhone4s. It accepts voice input in English language only. The Google Android's voice input system called "Voice Actions App" is another voice input system that enables users to dictate text messages and emails in addition to calling contacts saved on the phone. However these applications seems to be still in their infancy [52]. Nonetheless, in most languages the users don't have any choice other than typing text themselves using the generic QWETRY type keypad for data input [108]. Urdu is no exception. To-date, there is no speech recognition system developed for Urdu, Hindi or any other regional languages. In addition to that, NLP research in Urdu language is in its infancy, in particular there is very thin trace of research at character level applications.

The existence of the above mentioned issues warrants justification of this research and emphasizes the high need of finding optimum solutions to those separate but related problems.

1.3 Research Goals

As mentioned in the preceding sections, the aforementioned existing systems are not at par with the true potential of input systems in terms of usability, performance, safety (from perspective of hygiene), user friendliness, suitability to a certain languages and extendibility to others and satisfaction of user's choice etc. [108][112]. The main goals of this research are to solve these problems. Some other major goals of this research are discussed in much detail in the following sub-sections.

1.3.1 Performance of Existing Systems

QWERTY keyboards migrated from typewriters onto computers [108]. Users accepted them partially because they looked familiar and partly because there was no other choice [37]. Optimality in performance remained a constant and ignored issue with them. Dvorak layout uses less finger motion, increases typing rate, and reduces errors in comparison to its QWERTY counterpart [14][131]. However, over the years, production of the latter increased to an extent where it seemed impossible for the hardware producers to migrate to the DVORAK keyboards.

1.3.2 Problems

We live in the age of touch screen gadgets boom, thanks to smart phones and tablet PCs [58][146]. The change in technology has provided us with an opportunity which, if left un-availed, will turn into a threat that we might continue to use new systems without optimizing their input system [32][35]. The long ignored issue of 'performance' can be solved now [77][99]. This will require us to carefully design and develop keypads that offer optimum performance.

One of the major advantages of hardware keyboard is the existence of home row [62][146]. Small dots or hyphen-like markers under the index fingers of both hands provide tactile feedback to the users. This tactile feedback facilitates users to return fingers to their starting position (home row) after typing other letters on different rows. This allows a user to touch-type. The touch-typing is sometimes referred to as blind-typing because the user doesn't need to look at the keyboard while s/he is typing commands or data [11].

At present, the most common layout of hardware keyboards is QWERTY layout [15][108]. However, this keypad layout is of limited advantage when deployed on touch screen devices [166]. A touch of the screen results in typing the corresponding letter or any other character. Thus there is no concept of home row on a touch screen keypad [10][58]. As a result there is no tactile feedback to guide a user to return his/her fingers to their respective starting location on the keypad [23]. Thus we need to come up with and devise faster, accurate and user friendly input schemes and interfaces for data and text input on touch screen devices.

1.3.3 Hygiene in Design and Development

Hygiene is an important factor in GUIs (Graphical User Interfaces) [6][32][111]. Unfortunately, so far very little attention has been paid towards health hazards caused by use of technology [41][166]. During design and development of new systems, the designers and hardware producers take into account the parameter of hygiene hardly ever. Prolonged and careless use of computers results in causing serious hazards to the health of users [63][99][106]. Examples of some of these health hazards are RSI (Repetitive Strain Injuries), CTS (Carpal Tunnel Syndrome), CTDs (Cumulative Trauma Disorders), Tendinitis and most common complain i.e. of eyesight weakness etc. Similarly there are some other syndromes like Tenosynovitis, De-Quervain's Disease, Cervical Radiculopathy and Ulnar Nerve Entrapment etc.

The Health and Labor departments of New Jersey, USA compiled a report on 'Public Employees Occupational Safety and Health Program' [99]. This report provides a detailed list of health hazards caused by computers and other office gadgets. The report brings forward the need to create health friendly systems and emphasis that hygiene is of prime importance in prolonged and regular use of computers. The above mentioned and other health hazards, originating from the "bad" use of computers and other sister gadgets, and recommendations on their preventions will be discussed in detail in chapter 3.

1.4 Our Contributions

Our principle contributions through this research work are mentioned in the following.

We explored the vast horizons of modern touch screen gadgets, analyzed their performance and pointed out their short comings. In order to solve those shortcomings, we calculated sorted character unigram, bigram and trigram matrices of a sufficiently large growing corpus. This raw corpus has been created using more than 16,000 manually collected Urdu Unicode text files of general genre containing about 16 million words and more than 59 million characters. Frequency distribution of about 12,000 unique words is used for experimentation. Based on unigram character frequencies, the highest frequency characters (Urdu letters) were selected as base letters in the proposed new keypad layouts. Bigram frequencies were used for arrangement of letters on the base layout such that highly probable pairs of letters are placed as close to each other as possible. The bigram characters neighborhood statistics reveal that the non-alphabetic arrangement of Urdu letters alone results in additional 17% improvement in the efficiency of our proposed keypad. Based on the results of the two types of system evaluations we conducted i.e. automated evaluation and user evaluation, we were able to achieve higher accuracy, speedy typing, better usability and more user friendliness than traditional QWERTY replica keypads. More details of our proposed keypads are described in the chapter 4.

Our optimized proposed keypads also comply with the five principles of Ergonomics i.e. performance, safety, comfort, ease of use, aesthetics.

Hygienic designs i.e. we advocated and brought forward the long ignored need to keep users health in prime focus at the time of designing novel touch screen keypads. The computer graphics technique called “Onion Skinning” is used to ensure that the keypad require less screen area for “parking”. Using this technique, we put a new layer on top of the base keypad when a key is touched. This puts the base version of keypad in the background and brings a new layer on top of it showing the current button expanded with it’s edge characters shown as separate buttons. Thus the same screen area is shared between the background layer (base version) and the foreground layer (active button). This way the individual button sizes are bigger than any other in-the-market touch screen keypads. This ensures clear visibility, easier usability, low risk of CTS (Carpel Tunnel Syndrome) and low error rate etc. The clear visibility means that there is no strain on eyes of user while looking at screen for long hours. Easier usability makes it easier to touch-type (sometimes called blind typing). It would be considered hallmark of any system that reduces the risk of CTS to users. Since the buttons on our proposed keypads are big enough, so the user doesn’t need to bend his/her wrist at a discomforting angle. This will certainly reduce the risk of CTS. Finally the low error rate is also achieved due to bigger size of individual buttons. Hence the overall efficiency of time saving and errorless typing is increased.

Our proposed keypads possess another distinct characteristic that their sizes are adjustable to suit touch screen devices having different sizes. This makes it easy for deployment on any type of device. This characteristic becomes more important because the size of touch screen system is not standardized and different vendors produce touch screen gadgets in different sizes.

Our proposed keypad layouts are extendible to Perso-Arabic languages other than Urdu. Although these are optimized for Urdu, but there are other Perso-Arabic script languages that share most of letters in their respective alphabet e.g. Punjabi has only one letter more than Urdu. Persian has 32 letters and Arabic has 28 letters and both of these languages are subsets of Urdu in terms of the letters they use. Since all the letters of both Arabic and Persian languages are included in our proposed keypads, so therefore we need no amendments in the already developed keypads. Similarly, with little additions, languages like Pashto, Kashmiri, Sindhi etc. can also exploit our proposed keypad designs to create new keypads for each of these languages.

Our proposed keypads are not limited to Perso-Arabic script languages. With very little effort and trifling changes in the back-end codes, they can be used by other languages e.g. Hindi, Thai and other languages with large number of letters in their respective alphabets.

We recommend separate keypads for small, medium and large sized touch screen devices. However, the keypads can be deployed on devices other than recommended by us also, just in case a user wishes to do so. Thus these novel keypads provide users freedom of choice so that they may be able to select and use their respective favorite keypads for data input on any device they prefer.

In addition to introducing novel keypads, we also explored some other character level NLP applications e.g. Romanization and transliteration. Romanization is spelling out words in Roman script. Transliteration is phonetic translation across a pair of languages. We found various problems in Urdu Romanization and Transliteration. Different speakers Romanize and Transliterate in different ways while writing the same text.

Another contributing factor is that the existing Romanization is highly polluted by the incorrect but faster SMS lingo because it requires less number of keystrokes to compose text. These factors warrant the need to introducing a new and standard Romanization scheme that is unambiguous, faster in transliteration, unambiguous in reverse transliteration and that suits mobile phone text entry. We proposed Urdu Romanization scheme suitable for text entry using mobile phones and other handheld gadgets.

In our proposed Romanization, there is a one-to-one mapping among Roman and Urdu letters and strings of letters. The case sensitive unambiguous Romanization makes the data entry fast and accurate. Once mastered, our proposed Romanization can help and guide the foreign learners of Urdu language in learning correct pronunciation of Urdu words. More detailed discussion is presented in chapter 5.

Our proposed Romanizing technique makes the process of transliteration as well as the reverse transliteration fast, unambiguous and more users friendly. To date, there is no research work reported on reverse transliteration of Urdu. Reverse transliteration and making appropriate applications for it is challenging in any pair of languages because of its intrinsic lossy nature. Urdu is no exception. Our proposed Romanization technique provides a firm basis for tasks like reverse transliteration etc.

Before this research, there was no choice of keypads for Urdu or other Perso-Arabic script languages except the traditional QWERTY replica keypad on touch screen devices. For non-touch screen systems, multi-tap T9 keypads are also in use. However, they also have great room for improvement. We proposed three novel keypads each of which is most suited for small, medium and large touch screen devices respectively. Our proposed keypads offer optimum performance in data input as well.

Our proposed keypad for small touch screen system is 52.62% faster than in-the-market generic keypads. Medium size screen keypads showed 57.06% improvement. These results, some other results and the way we reached to them will be discussed in detail in the subsequent chapters.

We made another major contribution. According to the National Language Authority of Pakistan, Urdu alphabet contains 58 letters. Making a keypad for languages with large letter-set is challenging. In order to create optimized keypad designs, we reduced the

Urdu letter-set from 58 to 38 letters. However, the user can still type all the letters in Urdu. Section 2.1 in chapter 2 provides a detailed insight on this and other issues related to Urdu scripts.

Our proposed keypads are optimized for Urdu but applicable to Arabic, Persian, Punjabi and other Perso-Arabic script languages too. With minor changes in the backend script settings, our proposed keypads are applicable to non-Perso-Arabic script languages.

1.5 Dissertation Outlines

The remainder of this dissertation is organized as follows.

Chapter 2 is literature review. It is divided into two main sections namely 'Urdu' and 'Keypads'. In 'Urdu' section, both the linguistic and input system related issues are discussed. We describe the history, size, script, letter-set and contextual shape of Urdu letters etc. Other similar regional languages and their sizes are also touched upon. We describe how our proposed designs are applicable and extendible to other languages. In 'keypads' section, we discuss the contemporary keypads, their limitations and the need for designing and developing novel keypads.

Chapter 3 is dedicated to the need and importance of Hygienic Systems. In this chapter we advocate why hygiene is of foremost importance during the design phase of any novel system. Different types of health hazard caused by regular and prolonged usage of computer and sister gadgets are discussed. "Prevention is better than cure". The causes and possible prevention for muscular syndromes and eyesight problems caused by computers and screens of various devices are explained as case studies. Health friendly 'Ergo Keyboards' are also introduced.

Chapter 4 is divided into two sub-sections. **Section 4.1** introduces the generic multi-tap T9 keypads that are used with conventional non-touchscreen mobile phones. It provides knowledge on the basic technique of multi-tapping. The flaws in the existing in-the-market linear ordering based input scheme are pointed out. This technology is fading out. However at present, the non-touch screen multi-tap mobile phones are the leaders in terms of number of users. Therefore a higher performance frequency based keypad layout has been proposed and explained.

Section 4.2 throws light on various types of touch screen devices and groups them into three categories; a) medium size devices e.g. tablet PCs b) large size touch screen devices e.g. Acer ICONIA PC and c) small size touch screen devices e.g. smart phones etc. We propose optimized novel keypads for each of these categories. Results of evaluation are also shown in this chapter. Last one i.e. category C is about small touch screen devices and their intrinsic data input issues and characteristics. We argue that the traditional QWERTY replica keypads don't offer optimized performance in keying-in data and commands on small touch screen systems. We proposed an Ergonomics compliant keypad with better performance and higher safety from users' health point of view. Evaluation of our proposed keypad is done in two ways; automated and user evaluation. Both these evaluations are discussed in greater detail in this chapter.

Chapter 5 is about Urdu Romanization and transliteration. The unambiguous, case-sensitive, one-to-one Roman to Urdu mapping recommended in this chapter needs small CPU and little memory. It is more appropriate for higher layer tasks such as transliteration, particularly reverse transliteration etc. It is also helpful to foreign students in learning Urdu. Our Romanizing application is extendible to other languages as well.

Chapter 6 concludes this dissertation thesis. Final and concluding remarks on our research contributions discussed in the previous chapters have been presented. The future directions have also been presented in the end of this chapter.

Chapter 2

Literature Review

This chapter is divided into the following two main topics.

- **2.1 Urdu**
- **2.2 Keypads**

These sections are explained in much detail in the following.

2.1 Urdu

It belongs to the language family of central Indo-Aryan language. Urdu is written from right to left in Perso-Arabic script [52]. Its grammar is both gender and number sensitive [78][128]. It has many interesting integral linguistic features such as rich morphology, rich phonology, word segmentation and character level NLP applications etc. Some salient features of Urdu language are mentioned here.

2.1.1 Rank of Urdu

It is the national language of Pakistan and an official language of numerous states in India e.g. Uttar Pradesh (India's most populous state). It is the 2nd largest Arabic script language according to the number of speakers. It is the Lingua-franca of Indo-Pak subcontinent and neighboring countries [44]. Due to the large South Asian Diaspora, Urdu is spoken in various parts of the world including the Middle East, Europe and Americas. Ethnologue by Lewis [79] considered Urdu and Hindi as the same language and ranked it the 5th largest language of the world according to the number of speakers. The numbers of Urdu and Hindi speakers are given by Table 2.1.

Table 2.1 Hindi and Urdu Speakers

	Native Speakers	***2nd Language Speakers***	***Total***
Hindi	366,000,000	487,000,000	853,000,000
Urdu	60,290,000	104,000,000	164,290,000
Total	426,290,000	591,000,000	**1,017,000,000**

2.1.2 Urdu Script

The term ‘script’ has several meanings and needs disambiguation. It refers to the continuous natural and native way of writing text. Ligatures, words, phrases and sentences are formed based on correct and appropriate shapes of individual letters. Collectively all of these are referred to as ‘script’ in this dissertation.

Urdu is written from right to left in Perso-Arabic script. Arabic has 28 base letters while Persian has 32 letters. Both Arabic and Persian letter-sets are subsets of Urdu. However, the exact number of Urdu letters is not agreed upon. A number of articles report different numbers of letters in Urdu alphabet [11][86][89]. The NLA (National Language Authority), Pakistan reported the largest letter set that contains 58 letters [48][96][154][3]. It is shown in the following Figure 2.1

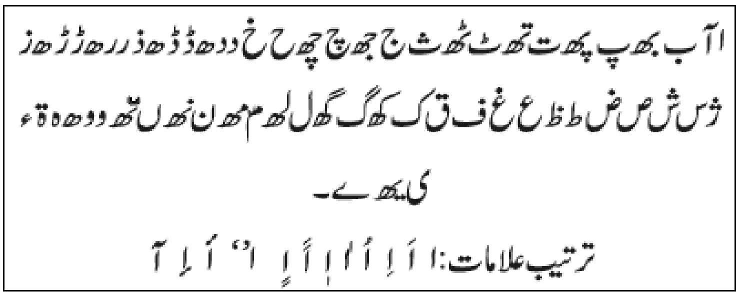


Figure 2.1 The 58 letters-set of Urdu alphabets.

According to Afzal and Hussain Urdu alphabet has 57 letters and 15 diacritical marks [83]. Hussain [52] reported 41 letters in Urdu. Ijaz and Hussain [87] mentioned 56 letters. We reduced the Urdu alphabet to basic 38 letters. This reduction process and the benefits achieved from it are discussed in detail in chapter 5.

2.1.3 Contextual shape changes of Urdu letters

Urdu letters change their shape based on their respective positions inside a word. A letter can have up to four different shapes i.e. base, initial, medial and final shapes [52].

Example:

A letter is in its base shape when it appears alone as a disjoint letter e.g. the letter "ج" pronounced as "jim" with IPA (International Phonetic Alphabet) "[d͡ʒ]". This letter is shown at serial number 8 in Table 2.2. Apart from the base shape of "ج", the rest of its three possible contextual shapes are shown in Figure 2.2.

Figure 2.2 Contextual shape changes of letter "ج"

Initial shape refers to the shape of a letter when it appears in the beginning of a ligature. Medial shape of a letter is written when it is joined by both the preceding and the following letters inside ligature. Final shape appears when a letter marks the end of a word or ligature. Durrani and Hussain [105] discussed this property of Urdu letters in much detail.

2.1.4 Ligatures and Diacritics

Designing optimized Urdu keypads for small screen widgets is a knotty problem [10]. Relatively large letter-set and no agreement over the total number of letters in Urdu alphabet make the problem more complex. In addition to the 58 letters shown in Figure

2.1, Ligatures and Diacritics are also borrowed from Arabic in Urdu. Ligatures are fixed blocks of letters each represented by a single Unicode [71][162]. Diacritics are another set of low frequency characters. They are small macron-like characters normally used to show the correct pronunciation of letters in a word. Both the Ligatures and Diacritics are used mostly in religious texts that have become part of Urdu but they have been originally borrowed from Arabic and Persian. The unigram frequencies of Ligatures and Diacritics are very low. Therefore we allocated them a single button on our proposed keyboard layout. Since these ligatures are comprised of individual letters so our keypads are still able to produce these ligatures.

2.1.5 Input mobility to other languages

Urdu alphabet is a superset of Persian and Arabic alphabets comprising 32 and 28 characters respectively [11][40]. We developed a new input system called JaPak IEOU (Japan-Pakistan's Input English Output Urdu) for speedy and easy Urdu Romanization using mobile phone keypads. It is explain in detail in chapter 5. It is equally applicable to Persian and Arabic languages because their alphabets are subsets of Urdu alphabet. By the same token, it is germane for all other languages with alphabets as subsets of Urdu alphabet. The proposed technique can be applied with trifling add-ons to some other major South Asian Perso-Arabic script languages where the alphabets are supersets of Urdu such as Punjabi, Sidhi and Pashto etc. Besides native Urdu, Arabic and Persian speakers, JaPak IEOU is expected to be useful for over 200 million inhabitants of the dense populated South Asian sub-continent bringing usability and convenience particularly to rural population.

2.1.6 Hamza (ء) and grave accent group

The Hamza (ء) group contains seven letters. This group comprises of extremely low frequency letters [11]. Based on their individual unigram frequencies, top three entries are shown in Table 2.2. Similarly the scarcely written diacritics are tabulated in Table 2.5. These are written hardly-ever in contemporary texts. However they are difficult to be ignored because of their important role in Urdu phonology, particularly the correct pronunciation of Urdu words.

Table 2.2 The Hamza (ء) and grave accent group (sorted by Frequency)

Unicode	Alphabet	PRL	Frequency	Percentage	Example
626	ئ	**’y**	665001	1.11884	کوئی
624	ؤ	**’w**	32614	0.05487	گاؤں،دباؤ
621	ء	**’**	25118	0.04226	شعراء

Diacritical marks are considered important part of Urdu pronunciation system but their use in modern written manuscript is highly scarce [89]. The Arabic loan diacritics ّ (تشدید tašdīd) represents germination. It is pronounced but written hardly ever. Similarly إ (Hamza under Alif), ٍ (do-zer) and ٖ (Khari zer) etc. are also obsolete.

The diacritics serve the same purpose as “ruby” (ルビ) in Japanese transcripts. Table 2.3 illustrates the three Urdu diacritics that don’t appear in modern texts but sometimes helpful when short vowels are required to be pronounced e.g. the letter “i” in “Pakistan”. The first and second columns show the DM (Diacritic Marks) and PRL (Proposed Roman Letter) respectively.

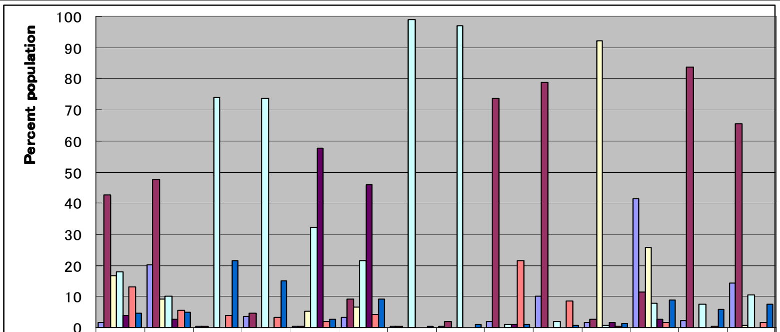
Table 2.3 Diacritics representing short Urdu vowels

DM	PRL	Urdu	Arabic	Unicode
◌َ	a	zbr	fTH	64E
◌ِ	i	zyr	ksrH	650
◌ُ	u	pysx	dmH	64F

2.1.7 Other Regional Languages

According to the population census organization, government of Pakistan, 67.5% of Pakistani population lives in rural areas [49]. They use local Perso-Arabic script languages such as Pashto, Saraiki, Balochi, Shina and Hindko etc. [50]. Pashto is used in Pakistan and Afghanistan whereas the other aforementioned languages are purely

Pakistani regional languages. In addition to the above, there are other regional languages such as Sindhi, Punjabi, Kashmiri etc. that are spoken on both sides of the border between Pakistan and India. For that reason, these languages are written in two different scripts in the two countries i.e. Perso-Arabic and Devanagri script in Pakistan and India respectively. The Pakistani languages are illustrated in Figure 2.3. Our proposed input system, JaPak IEOU, is able to cater the need of majority of rural population in Pakistan and neighboring areas where Perso-Arabic script languages are spoken. Analogously, it is equally exploitable by urban users.



Admin.Units	Pakistan		NWFP		Balochistan		FATA		Punjab		Sindh		Islamabad	
	Rural	Urban	Rural	Urban	Rural	Urban	Rural	Urban	Rural	Urban	Rural	Urban	Rural	Urban
Urdu	1.48	20.22	0.24	3.47	0.21	3.42	0.18	0.18	1.99	10.05	1.62	41.48	2.33	14.18
Punjabi	42.51	47.56	0.24	4.58	0.43	9.16	0.18	1.85	73.63	78.75	2.68	11.52	83.74	65.36
Sindhi	16.46	9.2	0.02	0.11	5.27	6.57	0.01	0	0.15	0.09	92.02	25.79	0.08	0.81
Pushto	18.06	9.94	73.98	73.55	32.16	21.61	99.15	97	0.87	1.81	0.61	7.96	7.62	10.51
Balochi	3.99	2.69	0.01	0.03	57.55	45.84	0.04	0	0.9	0.14	1.5	2.74	0.02	0.08
Saraiki	12.97	5.46	3.99	3.15	1.87	4.16	0	0	21.44	8.38	0.32	1.71	0.3	1.53
Others	4.53	4.93	21.52	15.11	2.51	9.24	0.43	0.96	1.02	0.78	1.25	8.8	5.91	7.53

Figure 2.3 Pakistani Population by Mother Tongue

2.1.8 Summary of Urdu

Syntactically, Urdu is a SVO (Subject Verb Object) language that has very rich linguistic features. Phonetically, Urdu is quite similar to Hindi. Written Urdu and Hindi use different and mutually exclusive scripts. However, in spoken form they appear to be the same language. According to Rai and Alok [113], “One man’s Hindi is another man’s Urdu”. Urdu is written in Perso-Arabic script while Hindi is written in Devanagri script. Many linguists consider Urdu and Hindi as one single language because of their essentially identical linguistic features and contend that they are made as two different languages for socio-political reasons.

Urdu shares syntax, morphology, and the core vocabulary with Hindi. Thus Urdu is the lingua-franca of the dense populated South Asia. It is undergoing gradual growth in countries like United Arab Emirates and Saudi Arabia where mostly informational signage is written in Arabic, English and Urdu. Urdu is also used in United Kingdom and other parts of the world due to large Diaspora. Thus we expect that a large population of users will benefit from our research.

2.2 Keypads

In this section, we discuss various types of contemporary keypads that are commonly used these days. We bring forward their limitations and usability issues etc. We point out the need for designing and developing novel keypads and suggest possible improvements in the existing keypads.

2.2.1 Mechanical keyboard

A mechanical keyboard is the hardware keyboard. It is a device that uses an arrangement of buttons or keys, to act as mechanical or electronic switches. A keyboard layout is any specific mechanical, visual, or functional arrangement of the keys, legends, or key-meaning associations. Mechanical layout refers to the placements of keys on a keyboard. Visual layout is the arrangement of the legends i.e. labels, markings and engravings which appear on individual buttons or keys of a keyboard. Functional layout means the arrangement of the key-meaning associations. It is determined in the software for a keyboard.

Typically the keys on a mechanical keyboard are grouped into sections with different types of keys. e.g. character or letter keys, numeric, navigation, function, editing, modifier, lock and system and GUI keys.

Most of the generic computers come with mechanical keyboards. There are a large number of national variants in keyboards that are used in different countries of the world, optimized for their respective languages e.g. English, Spanish and Japanese etc. The QWERTY type keyboards are the most common type of mechanical keyboard. However there are other variants in use also e.g. QWERTZ, AZERTY, DVORAK and those developed for non-Latin scripts.

Mechanical keyboards became the main input device for computers after punch cards and paper tapes declined in their use. We live in the age of touch screen systems now. The generic keyboards, mainly QWERTY type, have migrated from mechanical to virtual keypads that are used on touch screen systems. The lack of tactile feedback is a major handicap of generic keypad that stops it from achieving the same performance of input as that of the hardware keyboard. Moreover the traditional mechanical keyboard replica on small touch screen systems would require too much of screen space for ‘screen parking’. Thus we are at a turning point in time which require us to think of potent techniques that would particularly suit the touch screen systems.

We borrow the assessment method of virtual keypads from the physical hardware keyboards evaluation technique. This comprise of two major parameters; a) the easiness to learn and b) efficiency [126]. The former parameter takes into account the time needed for a novice to become a veteran with the keyboard whereas the latter parameter refers to the composing speed by a skilled user, a user well familiar with the system under study.

Design constraints are not limited to certain type of platforms, languages, devices and their respective inherent features [11]. There are some additional design issues also that are summarized in the following sub-sections.

At present, more and more data is being generated and uploaded using touch screen smart gadgets [58]. These gadgets come in various shapes and screen sizes such as tablet PCs and mobile phones etc. Recently, there have been zero button touch screen laptop systems in the market e.g., the Acer ICONIA [10]. The current trends and types of new gadgets being introduced in the market suggest the growth of touch screen systems in the days to come.

2.2.2 Virtual Keypads

Virtual keypad is also called soft keyboard [126][132][5]. Unlike the physical hardware keyboard(s), a virtual keypad shows up on the screen. Thus it consumes no physical space in the real world. However, it needs a much precious resource i.e. the screen area and uses some part of the same screen where data is typed i.e. the editor [5]. This gives rise to new concerns such as position, size, and orientation etc. of the virtual keypad w.r.t. the editor. We can make the virtual keypad context sensitive so that it is visible only when the user wants to input or edit text [163]. Theoretically we can show several distinct keypads at the same time, nonetheless a single user is expected to use only one virtual keypad at a single time.

2.2.3 Contemporary and proposed keypads

Apart from the conventional QWERTY and Dvorak keyboards, there are a number of keypads used for text entry e.g. Multi-tap T9, odometer-like, touch-and-flick, Septambic keyer and Twiddler etc. [28][112][114][145]. Various types of input systems are used in touch screen systems. Software solutions such as Dasher have been developed [33][54][118]. Dasher is a computer accessibility tool that enables users to write text without using a keypad [34][41]. The concept is nice because of it novelty but in practical it is very difficult to adapt to because of its slow speed of writing, particularly on small screen devices. QWERTY type keypad layout is the most common input system for most of the languages in the present age [15][108][145]. However, it fails to offer the desired performance and usability on touch screen platforms. On mobile phones, Multi-tap T9 keypads have been tried as well. Some developers rely on improving the back-end solutions such as word prediction, word completion, phrase completion etc. in a bid to speed up the input process and reduce typing effort and time.

We consider that keypad is the most basic and most important tool that needs improvement. An optimized keypad will enhance the overall input system of any device. In the following, we explain the generic in-the-market virtual keypads, OSKs (On-Screen Keyboards) and the multi-tap T9 keypads.

2.2.4 Existing On-Screen keyboard

Microsoft Windows comes with a built-in soft keyboard called the OSK. It supports a number of languages including Urdu. This virtual keypad is basically a replica of the generic and classical QWERTY type hardware keyboard. Both the 'Base' and 'Shift-ON' layouts of Microsoft Windows OSK are shown in the following Figures 2.4 and 2.5. The 'Base' layout means the layout of the keypad when the shift key is not pressed down while keying-in a certain key/character on the OSK.

Figure 2.4 Base layout of Microsoft Windows 7 OSK.

Figure 2.5 Shift-ON layout of Microsoft Windows 7 OSK.

This OSK has migrated to many touch screen platforms including tablet PCs and smart phones. However, in our research, we reached a conclusion that this keypad does not provide optimum performance and ease of use. Detailed results to back up our arguments are presented in Chapter 4.

2.2.5 Multi-tap T9 Keypads

For mobile phones, Multi-tap T9 replica keypads are also in use. A typical T9 type keypad for Urdu is shown in the following Figure 2.5

Figure 2.6 Samsung SGH-C140 Urdu/Arabic T9 keypad.

The working of Urdu Multi-tap keypad is explained in the Table 2.4. The term Multi-tap means tapping (pressing) the same button several times to type a single letter. For example in Table 2.4 if we wish to type the letter ب then we need to tap the key with numeric digit 2 as a label. Tapping the same button VI i.e. 6 times would result in producing the letter ث on the screen. The key with numeric digit 1 as its label is reserved for typing many special characters on the same key. This makes typing cumbersome and time consuming.

Table 2.4 Multi-tap input Table for existing T9 keypads

Numeric Buttons	***Number of taps to type an Urdu letter***						
	I	***II***	***III***	***IV***	***V***	***VI***	***VII***
②	ب	پ	ت	ة	ٹ	ث	
③	ا	آ	ؤ	ۀ	ء		
④	س	ش	ص	ض			
⑤	د	ڈ	ذ	ر	ڑ	ز	ژ
⑥	ج	چ	ح	خ			
⑦	ن	و	ہ،ھ	ی	ے		
⑧	ف	ق	ک	گ	ل	م	ں
⑨	ط	ظ	ع	غ			

Urdu letters are typed using numeric buttons labeled 2 through 9 (encircled digits) on a multi-tap mobile phone Urdu keypad. The numeric button with labels 0 and 1 are not shown in Table 2.4 because they are reserved for typing special characters. The left-most column shows the encircled numerals as row headers that represent the corresponding buttons of a multi-tap mobile phone Urdu keypad. The column headers, marked by Latin numerals, represent the Urdu letters that will be typed when the corresponding button (numeral in row header) is tapped/pressed a specified number of times. For example tapping the number 8 button only once will type the Urdu alphabet "ف". Tapping the same button seven times will result in typing the Urdu alphabet "ں". Both the above mentioned types of keypads are difficult to use and slow on touch screen systems. The multi-tap T9 type Urdu keypads have en suite shortcomings. According to unigram Urdu letters frequencies, the letter "ی" is the 2nd most widely used letter in Urdu. Ideally high frequency letters should be typed with single tap (press) of a button. Table 2.4 shows that typing a single letter "ی" requires four taps of key ⑦. The same flaw applies to some other high frequency letter as well e.g. "ر" on key ⑤ and "ے" on key ⑦ etc.

In the same way, the full sized QWERTY like keyboards are not free from weaknesses. They are not feasible for touch screen devices, in particular devices with small screen where limited screen area needs to be used astutely. This issue becomes more challenging when we design keypads for languages with a large number of alphabets such as Urdu language. The trade-off issues in size and position of keyboard, editor, and individual buttons etc. require great care at the design time. A good design must comply with the five principles of Ergonomics; Performance, Ease, Aesthetics, Comfort and Safety [165]. This goal becomes difficult to achieve if large number of keys (for large number of letters) have to be accommodated on a small limited screen area.

Keeping the above points in view, we propose novel keypads for touch screen devices of different sizes. Careful thought process during the design phase enabled us to make individual buttons large enough to be clearly visible and suitable for easy typing of Urdu text.

Hygiene is also one of the foremost important design factors. We tried to develop the keypads in such a manner that would be health friendly having much visibility and usability coupled with crafty arrangement of keys that is ideal for fast, correct, easy and efficient composing. Our optimization technique for arrangement of alphabets and unique interface for data input is extendable and equally applicable to other natural languages and various sizes of touch screen devices. The next chapter discusses in much detail, the hygiene and its importance in design.

Chapter 3

Hygiene in Design

3.1 Introduction to Hygienic design

Different interfaces suit different devices for users who need to input data in different natural languages [11]. Full keyboard replica designs with base and shift versions e.g., QWERTY and Dvorak etc. cause usability problems as well as visibility problems hence not viable for small touch screen systems [41][32][63]. The handheld touch screen devices offer very little screen area for keypad parking. This means that in QWERTY type keypads, the individual key size to type a letter becomes too small to clearly see and type with fingers. Thus such a keypad is more prone to errors during text entry. Besides, data input using small screen devices bring about health hazards to the user. Eyesight weakness, RSI (Repetitive Strain Injuries) and CTS (Carpal Tunnel Syndrome) etc. are only a few health hazards caused by technology/devices that we use in our daily life [6][63][111][128]. For example, in case of eyesight, the closer objects put greater strain on the muscles converging the eyes retina [8]. Stress on convergence system of eyes is crucial factor for strain [57][99] [106]. Thus we need to keep hygiene in prime focus during design and development of input systems, particularly for small touch screen devices.

Repetitive Strain Injuries, Carpal Tunnel Syndrome, Cumulative Trauma Disorders, ophthalmic endemics and other health hazards are caused by regular and prolonged use of computers and its sister gadgets [32][99][106]. Japan is among countries with the highest percentage of users of computers and mobile devices. However the number of Japanese suffering from weak vision and wearing spectacles or contact lenses is also one of the highest in the world, perhaps due to the over use of screens and computer gadgets. According to Kenichi Tanaka [80], more than 40% of Japanese population suffers from weak vision and they need to wear spectacles or contact lenses.

Lately small touch screen devices and tablet PCs are gaining popularity around the world. Such devices are among the chief driving forces behind miniaturization of technology. Graphical User Interfaces (GUI) for small screen devices are influenced by greedy approach i.e. display is cluttered with too many options on a single small screen [10]. Tiny icons, buttons and menus put excessive strain on users' eyes. Prolong use of such GUIs expose users to potential ophthalmic endemics. By proposing novel hygienic GUIs, we advocate that mammoth attention is required at design time so that use of computers does not procreate hazards to human health.

3.2 Methodology

Systems with hygienic GUIs can be created using hybrid techniques of Medical, Engineering, computer based Natural Language Processing (NLP) and Human Computer Interaction (HCI) etc. We carried out experiments on a large general genre corpus of size 15,594,403 words from real world data. We used unigram character frequencies to choose the base letters and bigram character frequencies to determine positions of individual keys on proposed keypads. We reduced Urdu alphabet-set from 58 to 38 letters in order to reduce the number of buttons required on our keypads. This enabled us to develop clearly visible keypads that put less strain on users' eye. Our proposed new Romanization called JaPak IEOU ensures speedy and correct typing. Onion skinning is also used in creating our keypads that ensures better visibility through creation of larger buttons that also enable blind-touch composing.

We put forth hygiene in prime focus at the design time. Small devices put more strain on eyes due to acute and meager visibility [4][8][12][57]. When a user looks at any close object, his/her eyes do two things; they accommodate and converge. Both of these can contribute to eyestrain [25]. RPA (Resting Point of Accommodation) and Convergence prospects are among important considerations at the design time. These are explained further in the following.

3.2.1 Resting Point of Accommodation

Resting Point of Accommodation (RPA) deals with the point when the lens capsule changes shape to focus on a close object [57]. Human eyes have a default accommodation distance called RPA. This is the distance at which the eyes focus when there is nothing to focus on. The RPA averages 30 inches for younger people and gets farther away with age. In the mid-1980s, it was thought that people would have less eyestrain if the monitor were placed at the distance that coincided with a person's resting point of accommodation. RPA remains a vital design parameter. However more research has shown that the RPA is not the only consideration.

3.2.2 Convergence

Convergence allows the image of the objects to be projected to the same relative place on each retina [8]. Convergence is when the eyes turn inward toward the nose when we view close objects. The closer the objects, the greater the strain on the muscles that converge the eyes. Without accurate convergence, we see double images.

3.2.3 Resting Point of Vergence

The visual system also has a resting point of vergence (RPV). It is similar to the resting point of accommodation, but it's the distance at which the eyes are set to converge when there is no object to converge on. It's also known as dark vergence which means that in total darkness the eyes are set to focus at a particular distance so that if the lights were turned on, an object at that distance would be in clear focus. The RPV averages about 45 inches when looking straight ahead and comes in to about 35 inches with a 30-degree downward gaze angle. Research by Jaschinksi-Kruza [57] and Owens and Wolf-Kelly [104] have shown the stress of convergence contributes more to visual discomfort than the stress of accommodation.

3.3 Literature Review

In order to investigate the effect of the above factors, Jaschinksi-Kruza [57][166] carried out a research study in which they divided subjects into two groups, near and far resting points of accommodation. The first (near) group had RPAs of around 20 inches. The second (far) group's RPAs averaged 40 inches. Both groups worked on computers at viewing distances of 20 inches and 40 inches. As expected, the near group had less eyestrain working at 20 inches than the far group. But both the near and far groups had less eyestrain at the 40-inch distance. Both groups judged the 20-inch monitor distance as "too near," and both groups accepted the 40-inch distance. Although their resting points of accommodation were different, both groups had far resting points of vergence. Jaschinski-Kruza concluded that "the stress on the convergence system may be the crucial factor for visual strain."

When Jaschinski-Kruza measured performance, he found that both groups performed better at the 40-inch distance than they did at the 20-inch distance.

Research by Owens and Wolf-Kelly [104] found that after one hour of near work, the resting points of both accommodation and vergence shifted to a distance closer to the eyes. The size of the shifts depended on the resting points before the near work: Subjects who began the session with far resting points had the greatest inward shifts.

They found that the greater the inward shift in the resting point of accommodation, the greater the reduction in visual acuity, or keenness, when viewing a distant target. Changes in the resting point of accommodation did not correlate with subjective eye fatigue. On the other hand, greater inward shifts in the resting point of vergence were associated with greater eye fatigue, but not with changes in visual acuity.

When a user works at close distances, the visual system adapts by bringing the resting point of vergence closer. That inward movement could be the visual system's reaction to fatigue. While continually viewing objects closer than the resting point of vergence has been found to contribute to discomfort, no studies have shown greater fatigue with viewing distance farther than the resting point of vergence. What does this mean in practical terms? To answer this question, we need to find answers to the following question.

3.3.1 How far is adequate?

It's clear that if a user can't read characters on the screen, the viewing distance is too far. One of the recommendations can be that instead of moving the screen closer, the user should make the characters larger. The guidelines recommending close viewing distances can only encourage the computer industry to maintain relatively small characters. Those in turn, force closer viewing distances and can perpetuate eyestrain [26][56].

The limitation of arm's-length distance from monitor most likely came from recommendations on monitor placement in cockpits. NASA Standard 3000 [99] limits the displays that have associated controls. That is based on reach distance. While that is only for displays located close to their associated controls, the motion of reach distance has been used in other guidelines [165].

3.3.2 How close is too close?

It is difficult to set an exact limit for a minimum viewing distance. If sustained viewing closer than the resting point of vergence contributes to eyestrain, perhaps we should say that eye-screen distance should not be closer than the resting point of vergence [111]. The safe ratios for distance of eyes from a generic monitor’s screen are tabulated in Table 3.1.

Table 3.1 Safe ratio for distance between eyes and screen

Zone	**Distance**	
	Inches	**Cm (~)**
Usual	<= 10	<= 25
Occasional	10 To 25	26 To 65
Rare	>= 26	65+
IDEAL	**18 To 28**	**45 To 70**

The clarification is important that the values mentioned in the Table 3.1 are not universally true for all users. The main reason behind it is that the ‘comfortable and best viewing distance’ varies from user to user. It is almost impossible to put an absolute number on the best distance between eyes and screen. This depends on numerous factors such as font size, screen resolution, zoom level, brightness, sharpness and colors etc. It is recommended that a monitor should be at least 25 inches away from user’s eyes. Nonetheless, closer-viewing distances do not bother some people.

For small screen like mobile phones somewhere between your resting point of vergence and 6 inches in front of your nose you are going to experience discomfort. That distance is a combination of gaze angle, how long you've been working at the computer, your individual visual system's capabilities, and a number of other factors.

3.3.3 How far is too far?

Based on visual fatigue considerations farther viewing distances are better, at least up to the RPV. For example, if the RPV is 35 inches, an eye-to-screen distance of 25 inches is preferred to 20 inches. Thirty-five inches is better than 25 inches. Viewing distances beyond 35 inches (the RPV in this case) should neither increase nor decrease eyestrain.

To allow for greater eye-to-screen distances, we need hygienic input systems, software programs and devices. We need industry standards and clear guidelines that force manufacturers fabricate systems such that they don't force people to use systems other than the distances and postures which they are comfortable.

3.4 Musculo-Skeletal Disorders

Eyes are not the only part of human body affected by overuse of screens. There are many kinds of medical conditions that have ergonomic causes among office workers, including muscle disorders, tendon disorders, and nerve disorders [124]. These are often collectively referred to as MSDs (Musculo-skeletal Disorders) which describes the affected tissue rather than the inferred cause of the disorder. RSI (Repetitive Strain Injuries), CTS (Carpel Tunnel Syndrome), CTD (Cumulative Trauma Disorder) and ophthalmic endemics etc. are also caused by regular and prolonged use of computers and its sister gadgets [99][106]. To make the use of computers and other modern handheld gadgets hygienic, some hardware solutions and rule of thumb recommendations are mentioned in the following.

3.5 Ergo-Keyboards

Ergo means Ergonomics. It is the science of general rules and principles that guides designers to develop systems that are compliant with 5 principles called the 5-golden principles of Ergonomics i.e. safety, comfort, performance/productivity, ease of use and aesthetics [1][8][23][56][116]. Ergonomic Keyboards are hardware keyboards that are compliant with the principles of Ergonomics. The main purpose of Ergo keyboards are to use different muscles during keying. Maintain a hygienic posture during keying is important. The main target postures are deviation (sideways bending at the wrist) or pronation (working with palms facing the floor). Another important consideration is to reduce awkward postures of the arms or hands during keying. A prolong use of keyboard in a too much bending posture would lead to wrist disorder called CTS (Carpal tunnel syndrome). Thus making an awkward angle during typing puts too much strain on wrist and should be avoided.

CTS or median neuropathy it the wrist is a medical condition in which the median nerve is compressed at the wrist, leading to paresthesias. This is caused mainly by improper angle of the wrist for long hours repeatedly and regularly. According to The American Society of Plastic and Reconstructive Surgeons, CTS affects more than 250,000 workers per year in the United States. If delayed, the treatment for such problems is possible only through surgery and the user might need to wear CTS wrist brace for the rest of his/her life. A wrist brace or CTS splint is shown in the following Figure 3.1.

Figure 3.1 A carpal tunnel splint to keep the wrist straight[1].

[1] Image courtesy Wikipedia URL http://en.wikipedia.org/wiki/File:Carpal_tunnel_splint.jpg

Eyes can tire if a user gazes at something for too long. Eyes need to focus at different distances from time to time. It's a good idea to follow and implement the "20/20 reminder"; every twenty minutes, look twenty feet away for twenty seconds [101][164]. Eyes are strained more by close viewing than by distant viewing. The "right" distance for computer monitors and documents depends entirely on how clearly they can be read at a given distance. The general rule is to keep viewed material as far away as possible, provided it can be read easily.

In VDT workstations, the principal factors affecting the ability to see well are glare, amount of light, distance between the eye and the screen object and the readability of the document. Additional factors are the users' own vision quality and luminance between what is being looked at and its immediate environment.

Last but not the least, it is the responsibility of both hardware and software vendors to develop solutions and systems that make the use of computers and modern gadgets hygienic. We took users' health in prime focus during the design and development of our proposed touch screen keypads. At the same time, our keypads facilitate fast, correct and easy composing. This makes the overall task of composing hygienic, much safer and with good aesthetics.

Chapter 4

Proposed Keypads

This chapter is about our proposed keypads. We also discuss the existing keypads produced and marketed by various multinational producers and compare our proposed keypads against them. This chapter is dedicated to our published research work where we explain our research methodologies, experiments and findings. We illustrate various keypads developed by us and present results on their comparison with the existing in-the-market keypads. We also discuss our evaluation techniques are also discussed in this chapter.

We organized this chapter into the following two major sections.

4.1 Multi-tap T-9 keypads

4.2 Single-line, full keypads and 8-Keys keypad

Each of these sections is explained in detail in the following.

4.1 Multi-tap T-9 keypads

Like other countries of the world, the use of cell phones has become prevalent in Pakistan as well. Several cell phone manufacturers have incorporated Urdu language keypads into their cell phone products. This section gives an introduction about the existing multi-tap mobile keypads. We point out problems in the existing keypads and propose an improved counterpart layout of Urdu character set on contemporary cell phone 12-buttons multi-tap T9 type keypad.

The generic (non touch-screen) cell phones use a standard telephone 12-key keypad. The standard numeric telephone keypad contains digits 0-9, * and # symbols. The cell phone keypad also contains characters on keys for entering text into cell phones. Several characters are mapped to the same key because of small number of buttons available on cell phone keypads. The multi-tap method is the simplest text entry method in such situation. In multi-tapping, the user presses each key one or more times to specify the desired input character. Bilingual keyboards provide the ability to enter text in different languages. Urdu-English twelve button keyboard is a bilingual keyboard for cell phones. The smallest alphabet of Urdu contains 38 letters compared to English language, which contains 26 letters. The largest alphabet of Urdu contains 58 letters [2][47] [89]. For multi-tap optimized keypad design, we used Urdu alphabet containing 45 letters [87]. The large number of characters in Urdu language makes text entry very slow. Moreover, out of the 12 keys on mobile phones only 8 are used for entering text. All of the major brand cell phones use a standard mapping of Urdu characters given in Table 4.1. The columns show the characters that will be typed when the corresponding key (numeral in row header) is tapped/pressed a specified number of times mentioned at column headers.

Table 4.1 Existing 12-button Multi-tap input Table for existing T9 keypads

	Order						
Key	I	II	III	IV	V	VI	VII
2	ب	پ	ت	ة	ٹ	ث	
3	ا	آ	ؤ	ۂ	ء	ئ	
4	س	ش	ص	ض			
5	د	ڈ	ز	ر	ڑ	ذ	ژ
6	ج	چ	ح	خ	ہ		
7	ن	و	ھ	ی	ے	ۓ	
8	ف	ق	ک	گ	ل	م	ں
9	ط	ظ	ع	غ			

Figure 4.1 Nokia 3250 Urdu/Arabic T9 keypad

The existing layout for Urdu language is derived from standard Arabic keypad. This is implemented on many handsets produced and marketed by well-known mobile phone set producers such as Nokia 3250 and Samsung SGH-C140 which are shown in Figure 4.1 and Figure 4.2 respectively.

Figure 4.2 Samsung SGH-C140 Arabic/Urdu T9 keypad

The extended keypad layout for Urdu is implemented by handsets such as Samsung SGHC140 (www.samsung.com) (Figure 4.2). This mapping is inefficient in terms of key strokes per character (KSPC) and key strokes per word (KSPW). The layout of characters for Urdu language on cell phone keypad can be improved based on the frequency analysis of Urdu letters.

4.1.1 Frequency Based Character Mapping

Frequency based cell phone keyboard layout has been designed and developed for English language to make typing English text on cell phones easier and faster. For Urdu language the optimized layout presented in Table 4.2 is based on l frequency analysis of 16,638,852 words raw corpus. The frequencies of individual letters in the corpus are shown in Table 4.3. The ordering of characters on each key was decided based on digraph frequencies. Figure 4.3 shows the optimized keypad based on unigram letter frequencies as shown in Table 4.3.

Figure 4.3 Optimized multi-tap frequency based keypad layout

The letters on the keypads in the above Figure 4.3 is different than the existing in-the-market mobile phone handsets. This arrangement is based on the frequencies of individual Urdu letters. This means that the letter that is used the most by Urdu users would be readily available for typing, in most cases on a single tap of the key on which that particular letter has been mapped. The order and arrangement of letters with their corresponding keys has been tabulated in Table 4.2. It is worthwhile a mention here that there is no letter that will need 7 taps of a single key, hence the last column with the row header VII is left blank in our proposed optimized and frequency based 12 button T-9 type keypad.

Table 4.2 Optimized 12-button keyboard layout

Numeric Buttons	***Number of taps to type an Urdu letter***						
	I	***II***	***III***	***IV***	***V***	***VI***	***VII***
②	ا	م	ج	ح	ڈ	ء	
③	ی	ت	ھ،	ز	ژ	ۂ	
④	ک	س	ئ	ٹ	ض	ڑ	
⑤	ر	ل	گ	چ	ظ	ۃ	
⑥	و	ب	ع	خ	غ	ۓ	
⑦	ہ	ں	ف	ص	ذ		
⑧	ے	د	ق	آ	ث		
⑨	ن	پ	ش	ط	ؤ		

The proposed optimized layout shown in Table 4.2 is based on the frequencies of Urdu characters. The layout has been constructed by mapping consecutive characters from rows of Table 4.3 to cells of Table 4.2.

4.1.2 Evaluation

The proposed keypad layout has been evaluated on character-set and words from the lexicon derived from Urdu corpus. Table 4.3 shows the comparison of keystrokes per letter for individual Urdu letters on ‘standard’ layout and frequency based layout. The keystrokes per letter for ‘standard’ layout are shown in column named KSPL-S (Keystrokes per letter on Standard layout). The column with KSPL-F (Keystrokes per letter on Frequency based layout) shows keystrokes per letter on frequency based layout keypad.

Table 4.3 Keystroke per letter comparison

Uni = Unicode

KSPL-MT = Required key strokes per letter using multi-tap design

KSPL-TS = Required key strokes per letter using touch screen design

Uni	Urdu letter	Frequency	%	KSPL-S	KSPL-F	S-Exp	F-Exp
627	ا	6733610	12.24	1	1	12.24	12.24
6cc	ی	5752357	10.45	4	1	41.8	10.45
6a9	ک	3911143	7.11	3	1	21.33	7.11
631	ر	3669392	6.67	4	1	26.68	6.67
648	و	3327481	6.05	2	1	12.1	6.05
6c1	ہ	2994305	5.44	5	1	27.2	5.44
6d2	ے	2857846	5.19	5	1	25.95	5.19
646	ن	2773651	5.04	1	1	5.04	5.04
645	م	2684946	4.88	6	2	29.28	9.76
62a	ت	2117669	3.85	3	2	11.55	7.7
633	س	1987451	3.61	1	2	3.61	7.22
644	ل	1915841	3.48	5	2	17.4	6.96
628	ب	1492997	2.71	1	2	2.71	5.42
6ba	ں	1469466	2.67	7	2	18.69	5.34
62f	د	1431230	2.6	1	2	2.6	5.2
67e	پ	914273	1.66	2	2	3.32	3.32
62c	ج	844670	1.53	1	3	1.53	4.59
6be	ھ	800600	1.45	3	3	4.35	4.35
626	ئ	664594	1.2	6	3	7.2	3.6
6af	گ	643263	1.17	4	3	4.68	3.51
639	ع	636166	1.16	3	3	3.48	3.48
641	ف	546973	0.99	1	3	0.99	2.97
642	ق	544460	0.99	2	3	1.98	2.97

Uni	Urdu letter	Frequency	%	KSPL-S	KSPL-F	S-Exp	F-Exp
634	ش	532262	0.97	2	3	1.94	2.91
62d	ح	501602	0.91	3	4	2.73	3.64
632	ز	454158	0.83	3	4	2.49	3.32
679	ٹ	420666	0.76	5	4	3.8	3.04
686	چ	358159	0.65	2	4	1.3	2.6
62e	خ	352729	0.64	4	4	2.56	2.56
635	ص	327434	0.59	3	4	1.77	2.36
622	آ	259879	0.47	2	4	0.94	1.88
637	ط	220613	0.4	1	4	0.4	1.6
688	ڈ	183081	0.33	2	5	0.66	1.65
691	ڑ	143244	0.26	5	5	1.3	1.3
636	ض	142813	0.26	4	5	1.04	1.3
638	ظ	104163	0.19	2	5	0.38	0.95
63a	غ	100331	0.18	4	5	0.72	0.9
630	ذ	79372	0.14	6	5	0.84	0.7
62b	ث	69641	0.1	6	5	0.6	0.5
624	ؤ	32355	0.06	3	5	0.18	0.3
621	ء	24930	0.045	5	6	0.225	0.27
6c2	ۂ	4390	0.007	4	6	0.028	0.042
698	ژ	2522	0.004	7	6	0.028	0.024
629	ة	2275	0.004	4	6	0.016	0.024
6d3	ۓ	1479	0.002	6	6	0.012	0.012
Total		**55032482**	**100**	**154**	**150**	**309.7**	**166.5**

The last two columns of Table 4.3 show the expected values of keystrokes for ‘standard’ (S-Exp: Standard layout Expectancy) and frequency based (F-Exp: Frequency based layout Expectancy) layouts respectively. The expectancy values for each character have been computed by multiplying percentage by the number of keystrokes required by each layout. Total of the expectancy value of all the characters for the ‘standard’ layout i.e. 309.7 is much larger than 166.5 for frequency based layout. As a result, most frequently occurring characters are typed quickly compared to the least occurring characters. The layout has also been evaluated on 100 most frequent Urdu words. The total number of keystrokes for 100 most occurring words in contemporary existing layout is 917 where as in the proposed layout it is 457 which is an improvement of 50.16%. The number of keystrokes required for a lexicon of 51218 words (excluding the probability of each word) reduced by 36.74% KSPW in frequency based layout KSPW (key strokes per word) which is a significant improvement over the current standard layout KSPW.

4.2 Single-line, full keypad and 8-Keys keypad

In this section, we introduce different input systems for different types of devices based on their sizes as medium screen (Single-line keypad for tablet PCs), large screen (full touch screen PCs) and small screen (8-keys keypad for smart phones). We used separate techniques that suit each of the three types of devices. We designed and developed novel keyboard and keypads for text input on various types of touch-screen devices. All our proposed keypads are optimized for composing Urdu and equally applicable to other Perso-Arabic script languages such as Arabic and Persian etc. However, after small changes our proposed models can be used for non-Arabic languages also particularly those languages that have larger number of letters in their respective alphabets.

Urdu is the 2nd largest Arabic script language according to the number of speakers [79] [43]. However its little presence on the internet does not qualify its rank [11]. Among its major causes is the limited platform support and meager interface designs for composing write-ups in Urdu. Designing optimized Urdu keypads for small screen widgets is a knotty problem since Urdu has a relatively large alphabet set. Various sources and/or authors report different number of letters in Urdu letter set [52][83][87][11]. In addition, Ligatures and Diacritics are also borrowed from Arabic in Urdu but their unigram frequencies are very low. Ligatures are fixed blocks of letters each represented by a Unicode. Therefore we allocated them a single button on our proposed keypad layout. Diacritics form another set of low frequency characters. They are small macrons like characters normally used to show the correct pronunciation of letters in a word. Both the Ligatures and Diacritics are used mostly in religious texts that have been borrowed as it is in Urdu from Arabic and Persian.

In line with the growth of touch screen devices, IMEs (Input Method Editor/Environment) and on-screen/virtual keypads have been hot areas of research lately [145][60][5]. Composing Urdu on generic touch screen gadgets and PDA (Personal Digital Assistant) is a thorny job. Many modern gadgets either lack a good interface for typing Urdu e.g. Apple iPhone 4, or provide sluggish, inconvenient and hard to use keypads. There is no widely used agreed-upon keyboard or IME for Urdu [10].

We live in the age of touch screen gadgets boom. The future trends also show promising growth for them. Currently available input systems developed for standard PCs have room for improvement in efficiency, visibility and usability etc. The English QWERTY type keypads are not suitable for data input of languages with relatively large letter-sets. This concern becomes graver for non-Roman script languages such as Urdu and other Perso-Arabic script languages. Although it is spoken by a large population (shown in Table 2.1), the presence of Urdu is quite limited on the WWW. Among others, one of the reasons is the difficulty in composing Urdu on modern computers particularly the touch screen devices. This problem gets more critical on small screen handheld gadgets.

We developed a novel keypad for Urdu that is compliant with five golden principles of Ergonomics i.e. Performance, Ease, Aesthetics, Comfort and Safety. Our suggested keypad has been optimized for accurate, easy, speedy and efficient typing on small touch-screen handheld gadgets. We carefully designed our proposed keypad so that it offers better visibility, usability, aesthetics and user friendliness. Our optimization technique for arrangement of alphabets and unique interface for data input is extendable and equally applicable to other natural languages with large letter-set, in particular the Perso-Arabic script languages such as Sindhi, Kashmiri, Punjabi, Pashto etc.

At present, more and more data is being generated and uploaded using touch-screen smart gadgets that come in various shapes and screen sizes such as tablet PCs and mobile phones etc. Recently, there have been zero button touch screen laptop systems in the market e.g., the Acer ICONIA. The current trends and types of new gadgets being introduced in the market suggest the growth of touch screen systems in the days to come.

Different interfaces suit different devices for users who need to input data in different natural languages. Full keyboard replica designs with base and shift versions e.g., QWERTY and Dvorak etc. cause usability as well as visibility problems; hence not viable for small touch-screen systems. Besides, small screen devices bring about health hazards to the user. Eyesight weakness, RSI (Repetitive Strain Injuries) and CTS (Carpal Tunnel Syndrome) etc. are only a few health hazards caused by the technology/devices that we use in our daily life.

For example, in case of eyesight, the closer objects put greater strain on the muscles converging the eyes retina [8]. Stress on convergence system of eyes is crucial factor for strain [57][99] [106] Thus we need to keep hygiene in prime focus during design and development of input systems, particularly for small touch-screen devices.

Thus we tried to develop touch-screen keypads that would be health friendly having much visibility and usability coupled with crafty arrangement of keys which is ideal for fast, correct, easy and efficient composing. Our optimization technique for arrangement of alphabets and unique interface for data input is extendable and equally applicable to other natural languages and various sizes of touch screen devices.

4.2.1 Proposed Virtual Keypads

Apart from the conventional QWERTY and Dvorak keyboards, there are a number of keypads used for text entry e.g. tap, odometer-like, touch-and-flick, Septambic keyer and Twiddler etc [28].

Existing on-screen Urdu keyboard is replica of Microsoft Windows QWERTY type keyboard. For Mobile phones, Multi-tap T9 replica keypads are in use. The working of existing Urdu Multi-tap keypad is already explained in the Table 4.1.

Full sized QWERTY like keyboards are not feasible for touch screen devices, in particular devices with small screen where limited screen area needs to be used astutely. This issue becomes more challenging when we design keypads for languages with a large number of alphabets. The trade-off issues in size and position of keyboard, editor, and buttons etc. require great care at design time. A good design must comply with the five principles of Ergonomics; safety, comfort, ease of use, productivity/performance and aesthetics [165].

Keeping the above points in view, we proposed the following keypads for medium, large and small size touch screen devices.

4.2.1.1 Proposed keypad for Tablet PCs (medium size touch screen devices)

Urdu letters can be grouped based on their shapes keeping their alphabetical order can still be preserved. The similar shaped letters have been grouped on a single button in our proposed keypad for Tablet PCs. All the letters are arranged in a single line. Thus we can call this a single line home row keypad. This arrangement eliminated the homing time, the time required by a user to moves his/her finger from home row and then come back to the same position on a hardware keyboard or a soft keypad. Homing time is much longer on touch screen systems as compared to hardware keyboard because the user cannot touch the screen physically. Our proposed keypad for medium size touch screen devices is shown in Figure 4.6.

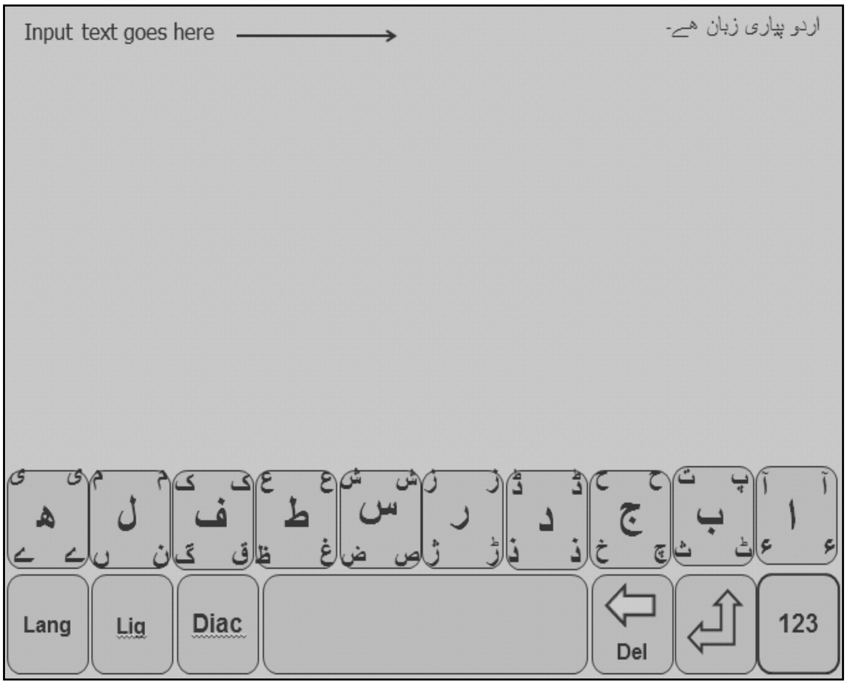


Figure 4.4 Proposed keypad for tablet-PC

There are 10 buttons for typing Urdu that show the corresponding letters in native alphabetical order with some letters shown on the edges of buttons. All the letter typing buttons are shown on a single row called the home row. Unlike hardware keyboards, it is very difficult to return fingers to exactly the same position on a touch-screen keypad. Thus we arranged all the letters on a single row so that the user doesn't need to lift the entire hand in order to type a letter. The user will keep both hands all the time above the single row also called the home row. The user just needs to touch and flick in order to type a certain letter. The little finger of right hand will type the rightmost button on the keypad while the little finger of the left hand will be used to type the leftmost button on the keypad. Letters on the right most edge and left most edge are too close to the end of the keypad and the screen. To type these letters, a user will need to move his/her little finger only 5 pixels on the desired button of the keypad. The four middle buttons will be typed using the index fingers of both hands. The reason for this is that the index fingers are the strongest typing fingers [105].

The lower row includes some familiar special buttons with additional button i.e. Lig and Diac. These terms stand for Ligatures and Diacritics respectively.

4.2.1.2 Proposed keypad for PCs (large size touch screen devices)

Figure 4.7 shows our frequency based full keyboard layout for touch screen systems. The current layout for Urdu keyboard is a replica of Microsoft Windows OSK (On-Screen Keyboard) as shown in Figure 2.4(a) and Figure 2.4(b) in chapter 2 of this dissertation report. This keyboard is not frequency based and has much room for improvement in that some high frequency letters are typed in combination of Shift-key, the last thing a user will need. Similarly, the buttons arrangement is not frequency based. Using Shift version of the keypad, shown in Figure 2.4(b), would require double labor and double amount of time in typing.

We solved these problems and proposed the frequency based full keyboard layout as shown in Figure 4.7. Based on the feedback from the volunteering subjects, the detailed performance examination of this keypad was not done. However the new layout has eliminated the Shift version of Microsoft Windows replica. We also re-arranged the position of keys based on the frequencies of individual letters such that the most frequent letters should be typed by the strongest typing finger i.e. the index finger. Another point in this arrangement is that the keypad is suitable for alternate hand typing which makes the typing much faster and easier. Using automated procedures to compare the existing and our proposed keypad would be a way forward.

There are some additional issues related only to the touch-screen keypads such as the inter-keys distance and the neighborhood of some standard keys might also be required to change. One such example is the neighboring keys of the “Backspace/Delete” and the “Enter/Return” keys. If user tries to touch-type, there are chances of touching a neighboring key by mistake. We suggest that “Backspace/Delete” and the “Enter/Return” keys should be designed and placed away from each other on touch screens. Also there should be some pixels left blank between every two neighboring buttons on the keypad to reduce typing mistakes. Our proposed keypad for large size touch screen devices is shown in Figure 4.5

Figure 4.5 Urdu letters with their corresponding positions/keys on QWERTY keypad

4.2.1.3 Proposed 8-Keys keypad (small size touch screen devices)

This section presents a deep and detailed insight on our proposed keypad for small size touch screen devices such as smart phones and PDAs (Personal Digital Assistant) etc. For the sake of brevity, we also call it 8-Keys keypad because it contains only 8 buttons in its base version. It is worth mentioning that if a user prefers then s/he can use our proposed keypad on tablet PC and full sized zero button touch-screen PC such as Acer ICONIA laptop etc. However, we recommend this keypad for small sized touch-screen devices.

Figure 4.4 shows the base image of our proposed frequency-based keypad for touch screen mobile phones. The individual letters are selected based on their unigram frequencies in 55-million characters corpus. The arrangement of characters is done on the basis of their corresponding bi-gram neighborhood frequencies. The letters in the base version, as shown in Figure 4.4, are not arranged in alphabetical order in Urdu. The base version of keypad shows the most frequently used Urdu letters. This results in much faster and more accurate composing of Urdu text. For the sake of easy understanding, all the remaining Urdu letters will be shown in small font on the corresponding 8-edges of each button. The leftmost button on lower row is reserved for changing the input language, switching to and from the numeric and special characters keypad or typing Ligatures and Diacritics etc.

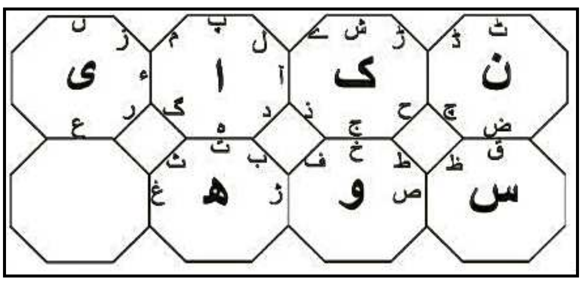
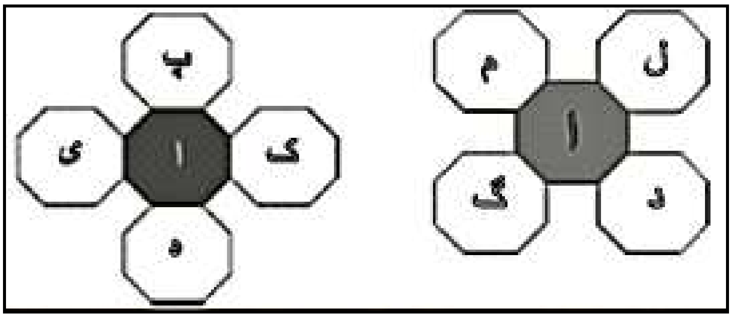


Figure 4.6 Proposed keypad for touch screen mobile phone.

The working of our proposed keypad is explained as when a “button press” event occurs then a single button gets the focus and expands into a smaller sub-keypad with the pressed letter displayed in the center of surrounding letters. Up to 8 neighboring letters of the pressed letter are displayed. These 8 new letters are displayed on a separate layer. The newly displayed 8 letters consist of 4 horizontal neighbors and 4 diagonal neighbors. The user will need to flick his finger in the direction of a particular letter in order to type it. In case of typing a base letter, no flick is required. Only tapping the base letter will do the typing. Beginners will need to look at the screen to select the correct neighboring letter. However experienced users can "touch type" in order to type their desired letter(s). The term "touch type" is sometimes referred to as "blind touch" also. The individual button sizes are big enough for blind touch and/or thumb typing. The size of buttons and their dimensions are flexible and can be adjusted according to the device on which the keypad is required to be deployed. A technique called "Onion Skinning" is used to show the new layer on top of the base layer. The diagonal and horizontal neighbors appear on a new layer on top of the base layer. In practice all the 8 neighboring letters will be visible and available for user to type. The diagonal neighboring letters can be used by a user just like the horizontal neighboring letters and vice versa. The event of a "button press" is illustrated in the following Figure 4.7 where the horizontal and diagonal neighbors are shown separately for better visibility and aesthetics.



Horizontal Neighbors	Diagonal Neighbors

Figure 4.7 Illustration of a button/key press event

For evaluation of our novel proposed keypad, we performed two types of evaluations; a) Automated evaluation procedure b) User evaluation. Our automated experiments on a large Urdu corpus reveal more than 52% improvement over contemporary keypads available in the market. We also carried out real world analysis through user evaluation.

Handheld touch screen widgets come in various sizes. Our proposed keypad is flexible enough to adapt to different screen sizes. Hence it is possible to increase or decrease the width or length or both to fit the screen dimensions of a specific device on which this keypad is required to be deployed. For example for Apple iPhone 4S, the recommended dimensions are as follows.

Table 4.4 Recommended size (in centimeters) of proposed keypad for Apple iPhone 4S

	Width/Height	***Length***
Keypad (base form)	2.50	5.00
Button (base form)	1.25	1.25

The above width, height and length are valid when the iPhone is in portrait mode. Recommended size depends on whether iPhone is in portrait mode or landscape mode. In case, iPhone is in landscape mode then the recommended size should be much longer horizontally. That is also incorporated in our proposed model.

The experimentation and comparison statistics of various keypads have been tabulated in the following.

4.2.2 Experimentation

We carried out experiments on a general genre growing corpus of size 15,594,403 words. Existing touch-screen systems start word prediction as soon as the user types the first letter. For words with length up to two letters, this seems to bring hardly any improvement to the typing speed. On the contrary, it makes the system more complex, memory hungry and larger in size putting more load on CPU. We recommend that word prediction should start after the second letter has been typed by the user. Out of

15,594,403 words, 4,784,234 words are less than or equal to two letters in length. Hence for the experiments of this study, we used a reduced corpus of size 10,810,169 words.
In practice it is faster to type on touch-screen than on multi-tap systems. Research that studies comparing the performance of touch screen and multi-tap systems could not be found. Thus for this study, we assumed “a touch” equal to the “a tap”. This puts bias in favor of the multi-tap systems.

The base form of keypad shows the most frequently used Urdu letters. This arrangement of letters is done on the basis of the bigram letter frequencies. The bigram neighborhood frequencies reveal that this non-alphabetic arrangement of Urdu letters alone gives amicable amount of 17% improvement in composing Urdu text. Results on comparison analysis are shown in the section 4.2.6.

4.2.3 Methodology

The methodology we adopted is mentioned stepwise in the following.

1. Calculate a frequency distribution for the words in an Urdu corpus of 15,594,403 words
2. Calculate a frequency distribution for the alphabets in the words i.e. the Unigram frequency distribution
3. Calculate a frequency distribution for the intra-words neighborhood of alphabets i.e. the characters bigram frequency distribution
4. Based on unigram frequencies, decide which alphabets will be on displayed in the “Base Version” of the keypad
5. Based on bigram frequencies, decide the order of alphabets for display in “Base Version” of the keypad
6. Carefully design the input method keeping in mind certain additional factors such as health issues and Ergonomics
7. Compare the existing and proposed system using suitable statistical models

4.2.4 Model Used

In order to measure the efficiency of our proposed keypad, we used the model presented by Mark D. Dunlop and Finbarr Taylor [92].

T(P) = Th + w (KwTk + r(Tm+Tk))
where
Th = 0.40s → homing time for the user to settle down on keyboard
Tk = 0.28s → time required to press a key
Tm = 1.35s → response time to a word prediction event
Kw = 5.421 (U) → average length of an Urdu word (our modification in the original model)
w = No. of words
r = 1.03 → ranked word list selection time[2]

To date, there is no full-fledged Urdu word prediction IME. In case of English and some other languages, existing touch screen systems start word prediction as soon as the user types the first letter. For words with length up to two letters, this seems to bring hardly any improvement to the typing speed. On the contrary, it makes the system more complex and larger in size putting more overhead on CPU. We recommend that word prediction should start after the second letter has been typed by the user. In the corpus we used, out of 15,594,403 words, 4,784,234 words are less than or equal to two letters in length. Hence for the experiments of this study, we discarded the words having length less than or equal to two character. The main reason to do so is; by the time the system is able to predict the desired word, the user will have already typed two letters or tapped the screen twice. Users' evaluation showed that responding to a word prediction event and then tapping the appropriate option takes longer than typing the next alphabet from the keypad. Reducing the size of corpus gave us the extra advantage of using a smaller corpus of size 10,810,169 words that subsequently resulted in the low CPU overhead and less memory requirement for our proposed input system.

[2] Tm and r are related to word prediction which is a separate area of research, beyond the scope of current work. This makes the last term in the model zero which has no effect on calculating the performance of our touch screen keypads.

4.2.5 Repeating Letters

The bigram character neighborhood matrix of the entire corpus gave us with an additional boost in typing speed in performance. Some Urdu words contain double and repeating letters. Using our proposed keypad the user needs to tap the same button twice in order to type a repeating letter. On the contrary, the same repeating letter can cost up to 12 taps in order to type it twice using a multi-tap T9 type of keypad,

We categorized the words with repeating letters in three different groups. These groups and their respective examples are presented in the following sub-sections.

4.2.5.1 Native Urdu Words

These are purely native single Urdu words. In comparison to our proposed keypad, typing this kind of letters i.e. the repeating letters take much longer on the existing generic multi-tap T9 keypads.

مقرر، ضرر، سبب، ممکن، کوشش، ممالک، مدد

4.2.5.2 Native Urdu Words (Compound)

These are Urdu words that are made up of a root word followed by a suffix. In such a case, the root word ends with a letter whereas the suffix begins with the same Urdu letter. This results in a repeating letter when a user types such a compound word.

خبررساں، دوررس

4.2.5.3 Foreign Words

Sometimes foreign words are written in native Urdu script. Examples of such foreign words are scorer, lecturer and manufacturer etc. These types of words result in repeating letters when written in native Urdu script. Thus they consume less time in typing on our proposed keypad.

سکورر، لیکچرر، مینوفکچرر

4.2.6 Results and Comparison Evaluation

We compared the performance of proposed keypad with its existing counterparts. The evaluation was done by two distinct techniques; a) Automated performance evaluation b) User evaluation.

4.2.6.1 Automated Performance Evaluation

Pressing a button several times to type a single letter/character is called a "tap". A "touch-and-flick" refers to a touch followed by a flick for typing a letter on a touch screen platform. The reduced corpus size and assumption of "touch=tap" put the bias in favor of the existing systems because a tap takes longer than a touch-and-flick. However, we still achieved results that show substantial improvement over the existing systems. The comparison of time required to type the corpus using existing Multi-tap T9 and our proposed keypads is given in the Table 4.5. Thus the proposed keypad is 48.65% faster than its contemporary counterparts.

Table 4.5 Time analysis results chart

Time	***Multi-tap (existing)***	***Touch Screen***
Seconds	263,380,598	135,249,436
Hours	73,161.28	37,569.29
Days	3,048.4	1,565.4
Improvement	**48.65%**	

The second parameter for automated comparison of proposed keypad with existing in-the-market keypads is the number of taps/touches. Our proposed keypad outperformed its counterparts on this measure also. The results are tabulated in Table

4.6. It shows that the proposed keypad achieved 52.62% improvement over the existing multi-tap keypad.

Table 4.6 Comparison of number of taps/touches required to type the corpus

	Multi-tap keypad (existing)	***Touch Screen keypad (Proposed)***
	170,580,560	80,818,830
Improvement	**52.62%**	

A simple everyday life observation reveals that a tap, particularly the multi-tap, takes longer than a touch-and-flick. Table 4.1 and Table 4.2 explained the input using multi-tap existing and proposed multi-tap optimized keypad. For the sake of fair comparison, we developed the multi-tap touch screen replica and used it on touch screen system for evaluation of our proposed keypad for small device. As seen in Table 4.6, typing with the help of Multi-tap T9 keypad is slow and time consuming. There are multiple reasons behind it. Some high frequency Urdu letters require 4 to 5 taps of a button to type them. Similarly some of the buttons need 7 taps to type a single letter. On the contrary, our proposed keypad requires a maximum of 2 taps/touches to type a letter (supposition; tap=touch=flick). Notwithstanding this supposition puts the bias in favor of the existing multi-tap system, we were able to reduce the typing payload by 46.10% w.r.t. composing all the letters in Urdu alphabet. Table 4.7 shows this comparison for both the existing and proposed keypad layouts.

Table 4.7 Comparison of cumulative typing payload to type all letters in Urdu alphabet

	Multi-tap (existing)	***Touch Screen***
Total number of taps	154	83
Improvement	**46.10%**	

4.2.6.2 User Evaluation

The user evaluation was carried out by three native Urdu speakers (all males and volunteers). Their ages ranged from 25 to 32 years. Two users were right-handed and one was left-handed. All of them were well versed with computers and experienced in typing but none of them was a professional typist. However, all of them had the experience of using the Microsoft Windows OSK for Urdu and Multi-tap T9 Urdu mobile keypad. The Acer ICONIA zero button PC running Microsoft Windows 7 was used as a test bed during users' evaluation. Each user was allowed to re-size the width and adjust the width and height of Microsoft's OSK according to the size of his hands and fingers. Our proposed keypad was novel and unseen for all the three participants. Except for a 10-minutes initial briefing, no training sessions were conducted before the volunteers could start using our proposed keypad for typing unseen Urdu text.

We conducted 20 typing sessions. A session means that each user was given unseen text to type on the Microsoft Windows OSK, the multi-tap T9 keypad and on our proposed keypad. The order to use the three keypads and the text to type by each user was all random. The text length was also kept random and the users were always given unseen text to type. This users' evaluation procedure was adopted in order to prevent the bias in favor of a particular keypad or user.

Figure 4.8 shows the real world performance analysis through user evaluation.

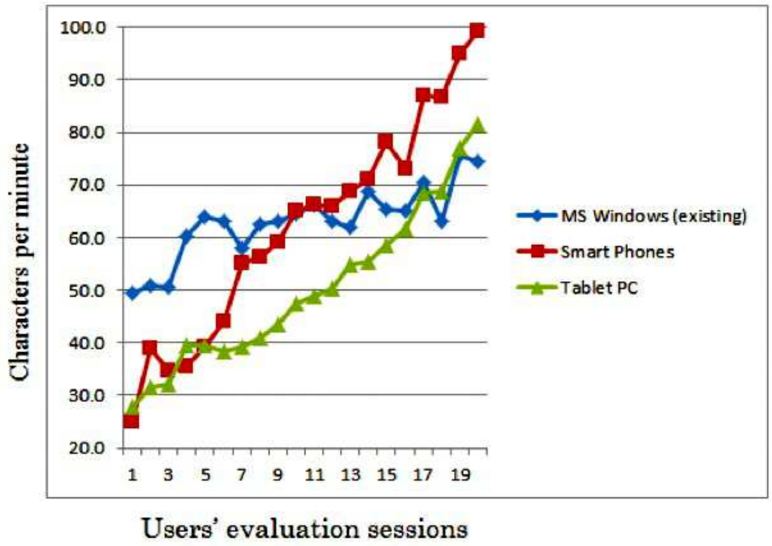


Figure 4.8 User evaluation results chart.

The results have been averaged and illustrated in Figure 4.8. X-axis represents the number of sessions while Y-axis means the typing speed of users in characters per minute. All the values in the chart are averages of all the three users who performed typing in a random order using random order of keypads and random pieces of text. As clear from the chart, the learning curve for our proposed keypad for small devices (smart phones etc.) is short and it's easy to memorize quickly. This shows that our proposed keypad is easy to understand and memorize, hence user friendly.

Evaluation of our proposed keypad for middle size touch screen devices (tablet PCs etc.) was carried out following the same procedure and in the same timing. The results of that evaluation are shown by the green curve in the above Figure 4.8. The blue line represents the Microsoft Windows OSK performance with which the subjects were already familiar. As clear from the green and blue line, our proposed keypad for middle size touch screen devices took only 18 sessions to surpass the performance of generic Microsoft Windows OSK.

Since the users were familiar with Microsoft Windows OSK and since they were able to use both their hands to type Urdu text, therefore the advantage was in favor of Microsoft OSK when we started users' evaluation. Nonetheless, it took our novel keypad for small and medium screen only 9 and 18 user sessions respectively to show better performance than the Microsoft Windows OSK. During evaluation of our proposed keypads, the users' evaluation did not show any significant difference between the working and performance of the diagonal and the horizontal neighboring letters illustrated in Figure 4.5. Similarly, no notable difference was observed between flicking the upper letters and lower letters on buttons of keypad in Figure 4.6.

Chapter 5

Transliteration

Kunrei-shiki, Hepburn and Nihon-shiki are major standard Romanization systems for Japanese language. Apart from minor differences, each system is based on the same technique where a string of characters has one-to-one relationship with kana characters that can be in any of the three scripts i.e. Kanji, Hiragana or Katakana e.g. "e" represents え and "ke" is used for "け" etc. Phonology is the corner stone for Roman to Japanese mapping and pronunciation of individual letters is of foremost importance.
The acronym JaPak IEOU refers to compound Phrase Japan-Pakistan`s Input English Output Urdu. The name has been formed according to the Otto Behaghel's "law of growing members" [65][103] and later contracted according to the de-facto Japanese abridging convention such as the word Wopro representing the compound term <u>Wo</u>rd-<u>Pro</u>cessor. In Urdu, aspirated Japak **(جھپک)** means "blink". This is in-line with the fact that our proposed system will produce output in a blink of an eye.

JaPak IEOU is an input system for Urdu that is based on the Japanese Kana input technique. It is a novel input system proposed as standard that composes Urdu on-the-fly using QWERTY keyboard for English. The Urdu alphabets/strings have one-to-one relation with case-sensitive English alphabets/strings.

5.1 JaPak IEOU Motivation

Urdu is one of the 10 most influential languages in the world yet uncommon on Internet. To date, there are only 10,000+ Urdu articles in famous source of knowledge such as Wikipedia. The main reason is that composing Urdu on computers, mobile phones and other handheld gadgets is a knotty task. There is no widely used keyboard or IME (Input Method Environments) for Urdu. Several famous daily e-papers are still collection of images e.g. The daily Express, The daily Jang; The daily Mashriq etc.

In this dissertation, in addition to introducing novel keypads we also explored some other character level NLP applications e.g. Romanization and transliteration.

Romanization is spelling out words in Roman script. Transliteration is phonetic translation across a pair of languages. اسلام علیکم is a common greeting used in Urdu and other Perso-Arabic script languages and appear quite often in written communication such as letters or e-mails. A research study of over 5,000 e-mails showed that there are 17 different variations of the same greetings when Romanized. This is an alarming figure and warrants the need of introducing a new Romanization scheme that is unambiguous, faster in transliteration, unambiguous in reverse transliteration and that suits mobile phone text entry. We proposed standard Urdu Romanization scheme suitable for text entry using mobile phones and other handheld gadgets.

Lately, non-standard way of user defined polluted Romanized Urdu called SMS-lingo has become a fashion among the youth because it requires less number of keystrokes in order to write a sentence. An example of SMS lingo in English would be; “U R 2 cloz” for “You are too close”. The popularity of such incorrect SMS Lingo has many reasons. One of the major ones is that the user wants to type his/her desired text in the least number of keystrokes. This becomes more important on mobile phone keypads. In our proposed Romanization, there is a one-to-one mapping among Roman and Urdu letters. The case sensitive unambiguous Romanization makes the data entry fast and accurate. Once mastered, our proposed Romanization can help and guide the foreign learners of Urdu language in learning correct pronunciation of Urdu words.

Thus it is high time to standardize Romanization of Urdu and other regional languages shown in Figure 2.5 (Chapter 2). All the above mentioned problems are solved using potent ambiguity free English-to-Urdu character based mapping described in Table 5.1. It enables users to type in English and get the resultant Urdu text in native Perso-Arabic script, hence the name IEOU (Input English Output Urdu).

Table 5.1 Basic 38 Urdu letters and the proposed unambiguous Romanization

Roman Letters	*Urdu Letters*	*S.No.*
a,e	ا	1
~	آ	2
b	ب	3
p	پ	4
T	ت	5
t	ٹ	6
S	ث	7
j	ج	8
C	چ	9
H	ح	10
K	خ	11
D	د	12
d	ڈ	13
Z	ذ	14
r	ر	15
R	ڑ	16
z	ز	17
J	ژ	18
s	س	19
sx	ش	20
Sx	ص	21
Zx	ض	22
Tx	ط	23
zx	ظ	24
3	ع	25
G	غ	26
f	ف	27
q	ق	28
k	ک	29
g	گ	30
l	ل	31
m	م	32
n, N	ن،ں	33
o,v,w	و	34
h	ہ،ھ	35
’	ء	36
y	ی	37
Y	ے	38

Further main motivations behind our proposed input system are mentioned in the following.

Urdu has no distinct upper and lower case letter forms. However the Romanization scheme shown in Table 5.1 is case-sensitive (Roman letters only). This helps in distinguishing the correct Urdu pronunciation. The right most column shows serial numbers and represents the alphabetical order of the corresponding Urdu letters. It is shown on the right side because the table is arranged for reading from right-to-left in order to comply with the native Urdu writing and reading style. In the leftmost and middle columns, each Urdu letter is mentioned along its respective letter for Romanization. Lower-case Roman letters represent the pronunciations exactly similar to their respective pronunciations in English. Upper-case letters represent similar but non-equal English pronunciation for the same letter.

The small case Roman letters correspond to Urdu letters that have exactly the same phoneme as in English. Lower case English letters represent Urdu letters whose phonemes produce the same sound as those of English. ب *(b)*, پ *(p)*, ٹ *(t)*, ج *(j)* etc. are examples of such phonetic pairs. On the contrary, each of the upper case English letter represent Urdu letter that has similar but not the same sound as that of English letters e.g. ت *(T)*, د *(D)*, ث *(S)* etc.

5.2 Reduced letter-set

It is worth mentioning that our reduced letter set can still write all the letters in Urdu language. We reduced letter set mentioned in Figure 2.1 by 22.62% to 38 basic alphabets set [82][153] which is shown in Table 5.1. This reduction was achieved by eliminating compound lettered aspirated consonants [89] mentioned in Table 5.2. We also grouped some very low frequency Urdu letters in a new group of letters called ‘the Hamza (ء) and grave accent group’. It is shown in Table 5.3. All these letters share a property that Hamza is used as a diacritic on each letter, hence the name ‘Hamza group’. Thus all the 58 letters mentioned in Figure 2.1 can be written using our proposed reduced letter set. As a result, there is no need to include the following aspirated consonants in our proposed system as distinct letters. Hence they are dropped out in our reduced letter set. This reduction made our job of developing keypads easier and enabled us to design bigger individual keys on touch screen keypads (already explained in Chapter 4).

Table 5.2 Urdu Aspirated Consonants

بھ [b^{h}]	ڑھ [ɽ^{h}]
پھ [p^{h}]	ڑھ [ɽ^{h}]
تھ [t̪^{h}]	کھ [k^{h}]
ٹھ [ʈ^{h}]	گھ [g^{h}]
جھ [ʤ^{h}]	لھ [l^{h}]
چھ [ʧ^{h}]	مھ [m^{h}]
دھ [d̪^{h}]	نھ [n^{h}]
ڈھ [ɖ^{h}]	

5.3 Benefits of JaPak IEOU

Apart from better keypad designs, we were able to achieve the following benefits from our novel and potent input system i.e. JaPak IEOU.

5.3.1 Mobility

Mobile phones and other hand held devices have changed computing paradigm radically. Our proposed system needs very little memory and processing power. Thus it is more suitable for deployment on mobile phones and handheld gadgets. It is equally good for desktop and web environments of computing. A tiny operating system resident process can ensure availability of Urdu input in all standard application software without installing additional software or hardware.

5.3.2 Online communication

Online communication is mostly restricted to ASCII only environments where not only Urdu alphabet but some Roman letters with diacritics are also unavailable. Even when Urdu, Arabic or Roman characters with diacritics are available, they are often hard to type. The lack of appropriate IMEs makes the problem further severe. This problem is faced by most speakers of Perso-Arabic script languages who use non-Roman alphabets. An ad-hoc solution consists of using Arabic numerals which mirror or resemble the relevant Arabic letters in shape e.g. “3” mirrors Urdu Alphabet “ع” and “7” resembles “ح”. The former has been included in JaPak IEOU design.

5.3.3 Ambiguity free

Its Romanization scheme helps protect contamination of Urdu Romanization. This would be useful in Urdu machine transliteration, particularly reverse transliteration [63]. The same can help foreign students learn and write/compose Urdu faster. It requires no deep knowledge of English hence predominantly suitable for users from rural areas where English medium education and ICT (Information and Communications Technologies) infrastructure are not well developed. Lately, the use of mobile phones and SMS (Short Message Service) has amplified dramatically in urban as well as rural regions of Pakistan and neighboring countries.

5.3.4 Projection of utility

JaPak IEOU is proposed to function using generic QWERTY keyboard and also work with any Urdu enabled IME. The typing speed for Urdu is estimated nearly equal to English because Urdu letters map to single upper or lower case letters in English. The Urdu character set along with our Proposed unambiguous Roman characters are described in Table 5.1. A major advantage of this one-to-one mapping would avoid various hard-to-type circumflexes or macrons e.g. ā, Ā, î, ñ and š etc. These are found mostly in old written Romanized texts.

5.3.5 Handheld gadgets support

Due to lack of Urdu typing support, mobile phone users develop their own user defined SMS-lingo for Romanizing Urdu and other regional languages. Our proposed system can be a strong candidate for replacing that non-standard Romanization. Mobile clients can benefit from it by the same token as desktop users without compromising much on typing speed. Our Romanization method is case sensitive. At times, it may be slower to type capital letters on mobile phones. Nonetheless higher layer HCI (Human Computer Interaction) compliant interface design improvements would make it fast.

5.3.6 Foreign learners

The phonetically unambiguous English-to-Urdu mapping makes it considerably quick and trouble-free for Non-native learners to utilize Romanization. JaPak IEOU can make learning process of Urdu easier and faster both for children and foreign students.

Table 5.3 The Hamza (ء) and grave accent group (sorted by Frequency)

Unicode	**Alphabet**	**PEC**	**Frequency**	**Percentage**	**Example**
626	ئ	**’y**	665001	1.11884	کوئی
624	ؤ	**’w**	32614	0.05487	گاؤں،دباؤ
621	ء	**’**	25118	0.04226	شعراء
670	◌ٰ	**’A**	22738	0.03826	تعالیٰ،لہٰذا،زکوٰۃ
672	ٲ	**’a**	15896	0.02674	فوراً،مثلاً
6c2	ۂ	**’H**	4416	0.00743	نظریۂ*
6d3	ۓ	**’Y**	1492	0.00251	رائے

Diacritical marks are considered important part of Urdu pronunciation system but their use in modern written manuscript is highly scarce. The Arabic loan diacritics ّ (تشدید *tašdīd)* represents germination. It is pronounced but written hardly ever. Similarly ٳ *(Hamza under Alif),* ◌ٍ *(do-zer)* and ◌ٖ (Khari zer) etc. are also obsolete.

The diacritics serve the same purpose as “ruby” (ルビ) in Japanese transcripts. Table 5.4 illustrates the three Urdu diacritics that don’t appear in modern texts but sometimes helpful when short vowels are required to be pronounced e.g. the letter “i” in “Pakistan”. The first and second columns show the DM (Diacritic Marks) and PEC (Proposed English Characters) respectively.

Table 5.4 Diacritics representing short Urdu vowels

DM	PEC	Urdu	Arabic	Unicode
◌َ	a	zbr	fTH	64E
◌ِ	i	zyr	ksrH	650
◌ُ	u	pysx	dmH	64F

5.4 Frequency distribution

A sufficiently large growing raw corpus has been created using more than 16,259 manually collected Urdu Unicode text files of general genre containing 1,09,64,150 words and 5,94,36,661 characters. Frequency distribution of 12,137 unique words is used for experimentation. There were almost 675 words and 3,660 characters per file. The highest rank dozen of words are tabulated in Table 5.4. For typing speed assessment, Urdu words and their corresponding required KSPW (Key Strokes Per Word) are shown. All the words show exactly the same number of Urdu and English letters. However on some occasions, the short vowels become necessary to be written e.g. "i" in "Pakistan". This reduces typing speed to 94% in comparison to typing speed for English. Nonetheless, Urdu words can be Romanized using precisely equal number of letters for the pair of languages. It will make Urdu typing speed 100% equal to that of English. This is possible by skipping the extra "i" or "u" that represents the diacritical marks representing short vowels and shown in Table 5.4. In fact, they are not written in native Urdu text anyway. However, this practice is recommended after a user becomes well familiar with this input system. Examples of such character elimination process is also present in the Japanese system i.e. the Japanese alphabet "つ" can be written either typing "tsu" or "tu". The later shows that character "**s**" has been skipped. Writing "つ" in this manner is a regular practice for experienced Japanese users.

Table 5.5 A dozen highest frequency Urdu words

Unique words	**Frequency**	**Frequency (%)**	**JaPak IEOU Romanization**	**No. of Characters**	
				Urdu	**English**
کے	628367	3.73	kY	2	2
میں	571471	3.39	myN	3	3
کی	534508	3.17	ky	2	2
ہے	451820	2.68	HY	2	2
اور	390082	2.31	awr	3	3
سے	342591	2.03	sy	2	2

Unique words	Frequency	Frequency (%)	JaPak IEOU Romanization	No. of Characters	
				Urdu	English
کا	319255	1.90	ka	2	2
کو	278639	1.65	ko	2	2
اس	245028	1.45	es	2	2
نے	224643	1.33	ny	2	2
ہیں	219845	1.31	HyN	3	3
کہ	209872	1.25	kH	2	2

5.5 Experimentation and Error Analysis

Our bi-directional Urdu-to-English Romanizer program crashed at locations where it encountered inaccuracies or inconsistency in Urdu text. Typically such crashes occurred on words with frequency lower than 93. We continued the automated Romanization process down to word frequency limit equal to 58 after which the program encountered an error at least every alternate word. Thus we manually checked further low down the word frequency distribution.

3,59,543 out of 1,68,42,403 words were found erroneous. It means that Urdu typists make a mistake every 46.84 words. This high rate of mistakes shows the difficulty of typing Urdu text. Several errors that came to surface during our experiment are described below.

5.5.1 Alien letters

The automatic Romanization halts as soon as it encounters an alphabet other than Urdu e.g. ڜ، ڞ، ڑ and ݕ etc. These are not legal Urdu alphabets despite their close resemblance with Urdu alphabets. JaPak IEOU has been optimized for Urdu but it can be easily extended to other Perso-Arabic script languages having larger alphabet-sets.

5.5.2 Obsolete Diacritics

Some diacritics have already gone archaic e.g. ٳ *(Hamza under Alif),* ٍ *(do-zer)* and ٖ (Khari zer) etc. There are others that are pronounced but not written such as some of those as mentioned in Table 4.15.

5.5.3 Incorrect Spellings

Our program found an interesting problem. The search for two Urdu alphabets ئ and ۓ both displayed the same word "لئے". However frequency of on both instances of "لئے" was different i.e. 60,508 and 231 respectively. Thus our program detected two distinct words. This is because of incorrect spellings by the user. Due to orthographic style used in computers this problem is not visible to human eye.

5.6 Comparison with GT (Google Transliteration)

An authentic document on transliteration states; "phonetic translation across pairs of languages is called transliteration". According to this fact it can be concluded that Google Transliteration (http://www.google.com/transliterate/urdu) does not perform transliteration. On the contrary, it is merely a Romanization tool based on ambiguous mapping. Apple, mango, rose, karaoke, abandon, USA, Mexico etc; all these words produce incorrect corresponding Urdu words.

The GT Urdu-to-Roman character mapping has been created by trial and error method and enlisted in Appendix 1. It uses many-to-many relationship between pair of languages i.e. a single Urdu alphabet can be represented by multiple (upto 10) corresponding Roman strings. The same holds vice-versa. It seems beneficial from the point of view of users. But it produces poor results, e.g., the common word مائرہ (name for females) could not be produced on GT using all of these strings; myra, maira, maera, maaera, mayra, maaira, mera, mira, myara. Surprisingly all these produced the word میرا (mine). Similarly, results were not encouraging for another common word such as شعر (poetry verse or a couplet). GT also failed to produce common words ضائع، ذریعہ، باعث، اعظم، تعاون، شعراء etc. It has been concluded that GT performs poorly on alphabets ۂ، ع، ۃ، اً ، ء، ٰ.

JaPak IEOU has only one, straightforward and easy-to-guess possible Roman notationfor every Urdu word, e.g., مائرہ and میرا map to English strings “ma’yrH” and “myra” respectively. Similarly there is always one-to-one relationship i.e. Urdu letters map to single upper or lower case letters in English and vice-versa. JaPak IEOU system is proposed to operate on-the-go on character-by-character basis so that Roman alphabet is converted into Urdu letter as soon as it is typed by the user. The advantage of this method is that users can see the output words while they are being typed. Any errors encountered during Romanization can be corrected as soon as they occur. It will make the typing precision and speed considerably adequate. This is converse to GT which produces a word after the space bar has been pressed by the user. The cost can be too high for words like those discussed earlier.

5.7 Algorithm

The following algorithm explains operation of our proposed input system. It generates Urdu letters and forms words on-the-fly as the user types individual characters. The character mapping is done according to Appendix 1, presented at the end of this dissertation.

1 Get next English character
2 IF current character member of ambiguous letter set
3 True: PUSH current character on stack
3.1 Goto step 1
3.1.1 IF current character equals “**x**”
3.1.1.1 True: POP two characters from stack
3.1.1.2 APPEND two characters // form a pair of Roman letters
3.1.2.3 Print corresponding Urdu letter for pair of input characters
3.1.1.4 False: Goto step 3.2
3.2 False: Print corresponding Urdu letter for singular Roman character
4 Goto step 1
5 Exit

There are two nested checks in the above algorithm. First the input letter is checked if it belongs to any of the five pair-represented letters [{zx ,Tx ,Zx ,Sx ,sx} or {ض، ط، ظ، ة

ص، ش، }] or not. The Urdu output letter is delayed until the next letter has been received. After this the inner check is performed for checking one of the two possible pair types. If next character is lower case "x" then an appropriate Urdu letter out of the six (ش، ص، ض، ط، ظ، ۃ) is displayed. Else Table 5.1 provides appropriate output single letter.

5.8 Methodology

The approach employed for English-to-Urdu cross language mapping of alphabets is phonetic. This makes Romanization more natural, easy to predict and fast to learn. Lower case English letters represent Urdu letters whose phonemes produce the same sound as those of English. ب *(b)*, پ *(p)*, ٹ *(t)*, ج *(j)* etc. are examples of such phonetic pairs. On the contrary, upper case English letters represent Urdu letters that have similar but not the same sound as those of English letters e.g. ت *(T)*, د *(D)*, ث *(S)* etc.

English-to-Urdu and vice versa mapping has been described in Table 5.1, Table 5.3 and Table 5.4. The letter-pairs representing aspirated consonants in Table 5.2 have been removed as the proposed system is capable to produce those without assigning distinct Romanized counterparts. The goal is to avoid phonetic ambiguity so that unambiguous phonetic English-to-Urdu mapping can be achieved.

Both the Arabic letters *yaa* and *haa* are split into two in Urdu. *yaa* has been split into ی and ے whereas *haa* has been split into ہ and ھ . The *yaa* variant ی *(choṭī ye)* is used at the end of words for sound /i/. Similarly *haa* variants ھ *(choṭī he)* is used to indicate the aspirated consonant as shown in Table 5.2. Some letters which represent distinct consonants in Arabic are conflated in Persian. The same has spilled over to Urdu also e.g. the ***cerebral consonants*** /ḍ/ ڈ (ḍāl) and /ṭ/ ٹ (ṭe) are added. This was accomplished by placing a superscript ط (*to'e*) above the corresponding ***alveolar*** /t̪/ ت and /d̪/ د.

The process of Romanization becomes challenging at times e.g. the three letters س، ص ، ث produce the sound /s/ and are candidates for English letter "s" as clear from

Appendix 1. The closest to English phoneme is letter “س” hence assigned lowercase “s” in Table 5.1. “ث” is the next closest so mapped to uppercase “S”. Our goal is to keep mapping clean from phonetic ambiguity. This goal can be achieved by making a pair of English letters to represent “ص”. The candidates are “s or S” followed by any of the two English letters “v” and “x” because both these letters have not been used in our Romanization scheme. The letter “x” is the best choice since neither it has a counterpart Urdu alphabet nor a phoneme. Thus “ص” is represented by the pair “Sx” where “x” is considered silent. Other four letter-pairs have been created in similar fashion to resolve ambiguity.

Words in Urdu retain Arabic and Persian spellings but with many irregularities. The alphabet “ع” (the glottal stop character) is an Urdu letter loaned from Arabic. It mirrors the number “3”.

Urdu Alphabet و (vā'o) has been represented by three English letters **w**, **v** and **o** in Table 1. The letters **w** and **v** can be used exactly alternatively. This has been allowed to give users more liberty in Romanization without any risk of ambiguity. This alphabet sounds /v/ thus represented by **w** or **v** when it appears in the start or middle of a word otherwise it is pronounced as **/o/** and accordingly represented by **o**. Thus the word وہ can be written in either ways; **wh** and **vh.** Similarly, the word اور can be romanized equally in two ways **awr** and **avr**.

5.9 User Evaluation

Figure 5.1 shows the user evaluation of our proposed Romanized input method JaPak IEOU compared with the existing Urdu soft-keyboard i.e. the Microsoft Windows OSK. The latter was already included in the graph in Figure 4.8 in the previous chapter where we compared it with our proposed small and medium size keypads.

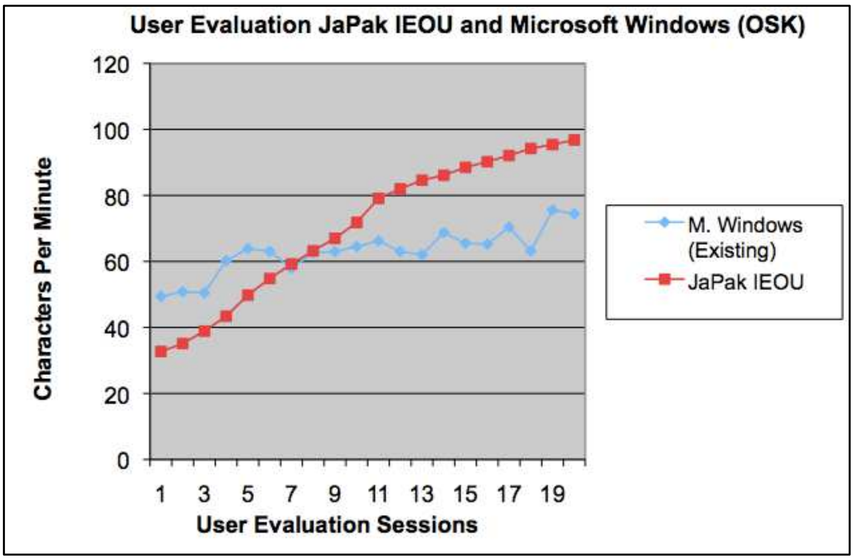


Figure 5.1 User evaluation results chart of Japak IEOU

Three native Urdu speakers volunteered for our user evaluation tests. The touch screen Acer ICONIA PC running Microsoft Windows 7 was used as the touch screen hardware platform for conducting user evaluation tests. Most of the conditions for user evaluation were kept the same as already described in the previous chapter. However, this time the subjects were asked to test our proposed JaPak IEOU system using QWERTY keyboard on touch screen system. The average values on comparison and the difference between the two systems is presented in Figure 5.1.

X-axis represents the number of sessions while Y-axis shows the typing speed of users in characters per minute. All the values in the chart are averages of all the three users who volunteered for our user evaluation. The chart shows that our proposed system performs better than the existing mode of Urdu input. This result is inline with our argument that our proposed JaPak IEOU system is easy to memorize, faster in performance and more user friendly.

Chapter 6

Conclusion

This chapter summarizes and concludes this dissertation report.

Different people prefer different input systems on different types of devices. Keeping that in view and keeping users health in prime focus, we proposed multiple types of keypads and input systems each of which has been tested and compared with the existing generic counterparts. Our proposed solutions show a great deal of improvement of the existing in-the-market systems.

The frequency based Urdu characters layout on 12-button phone keypad reduces the keystrokes per word significantly as compared to the traditional and existing keypad layout. The probabilistic analysis of 51218 words from the Urdu corpus shows that the proposed frequency based layout reduces keystrokes by 46% compared to the standard keyboard layout. Keeping in view the large number of letters in Urdu alphabet compared to the number of keys available on the mobile phone, memorizing the layout is worthwhile and practical.

For different types of touch screen devices, we proposed different types of keyboard and keypads. The comparison analysis showed promising results. In addition to great amount of improvement over existing keypads, our proposed designs are flexible because the size and dimensions of keypads, buttons, and editors can be adjusted according to the device on which the keypad is deployed. Similarly our keypads offer greater usability because Urdu letters include all the letters of Arabic and Persian. Hence these layouts are equally usable by the Arabic and Persian users. The keypads are optimized for Urdu though. With minor additions, our input systems are extendible to other Perso-Arabic languages as well.

For small handheld touch screen devices, we proposed a novel keypad. The comparison analysis were performed on two distinct tracks; the automated procedures and by detailed user evaluation. Both the evaluation method showed promising results. In addition to a significant amount of improvement over existing keypads, our proposed keypad design is flexible because the size and dimensions of keypads, buttons, and editors can be adjusted according to the device on which the keypad is intended and required to be deployed. Similarly our keypad offers greater usability because Urdu letters include all the letters of Arabic and Persian alphabet. Hence our keypad is equally usable by the Arabic and Persian users. Nonetheless, the keypad is optimized for Urdu. With minor additions, our input system is extendible to other Perso-Arabic script languages.

We also proposed a novel unambiguous Urdu Romanization technique. The exiting Romanization is ambiguous hence difficult for users to adopt. It contains special characters like macrons that are almost impossible to write on devices with small memory low power CPUs. We presented a discrete approach for typing Urdu on any device having ASCII support with any English keyboard. The lack of clear and distinct rules makes the task of developing an automated Romanizer more difficult and complex. This low overhead, light weight, fast and precise input mechanism is expected to be well received. It has the potential to open new avenues of research and development in Urdu computation.

The system has been optimized for Urdu but it is likewise practical for users of Arabic and Persian because the alphabet sets of both these languages are subsets of Urdu character set. It is also easily extendable to other regional languages such as Punjabi, Pashto and Sindhi etc. As a result, it will be more constructive for users from majority rural regions.

The Way Forward

Krestensson [105] said, “If we don’t search, we will never know if we are currently at a local or global optimum”. Hardware keyboard designs available today have room for improvements. Using our proposed systems together with HCI compliant designs can be an interesting future direction. On screen input methods and console systems can also be exploited in the days to come. Most popular modern gadgets like iPhone, iPad and other modern touch-screen systems are still waiting for optimized input interface for

Urdu language. If the vendors desire, our proposed systems are ready for deployment on above mentioned and other commercial platforms.

We intend to extend our keypads to include other Perso-Arabic script languages such as Punjabi, Pashto, Dari and Potohari and Kashmiri etc. More thorough testing of our keypad by a larger score of human subjects is welcome. Another possibility to exploit our work can be in the design of a single hand operated keypad (separate designs for each of the left and right hand), single finger operated and two fingers operated keypad designs suitable for numerous touch screen devices.

Appendix 1: Urdu alphabets with proposed Romanization, corresponding NLA Phonemic representation and comparison with Google Transliteration

Letter	Letter Name	PEC	Phonemic representation (in IPA)	GT Romanization list alphabet/string
آ	*Alif mad*	~	long vowel	a, aa
ا	*Alif*	a/e	/ɪ/,/ʊ/,/ə/,/ɑ/ depends on diacritic	a, aa, e, i, o, oo, u, ao
ب	*Be*	B	/b/	b, ba, bi, be, bu
پ	*Pe*	P	/p/	p, f, pa, pi, pe, pu, fa, fi, fe, fo, fu
ت	*Te*	T	/t̪/ (alveolar)	t, tt, ta, te, ti, tu
ٹ	*ṭe*	T	/ʈ/ (cerebral)	t, tt, te, ti, tu
ث	*Se*	S	/s/	s, se
ج	*Jīm*	J	/dʒ/	j, ja, je, ji, ju
چ	*Ce*	C	/tʃ/	c, ch, cha, che, chi, chu
ح	*Baṛī he*	H	/h/	h, ha, hi, hu
خ	*Khe*	K	/x/	k, kh, kha, khe, khi, khu
د	*Dāl*	D	/d̪/ (alveolar)	d, da, de, di, du
ڈ	*ḍāl*	d	/ɖ/ (cerebral)	d, da, di, du
ذ	*Zāl*	Z	/z/	z, za, ze, zi, zu
ر	*Re*	r	/r/	r, ra, re, ri, ru
ڑ	*ṛe*	R	/ɽ/ (cerebral flap)	R
ز	*Ze*	z	/z/	z, zz, za, ze, zi, zu
ژ	*Zhe*	J	/ʒ/	j, z, je
س	*Sīn*	s	/s/	s, sa, se, si, su
ش	*shīn*	sx	/ʃ/	s, x, sh, sha, she, shi, shu
ص	*su'ād*	Sx	/s/	s, sa, se, si, su
ض	*zu'ād*	Zx	/z/	j, z, ja, je, za
ط	*To'e*	Tx	/t/ (alveolar)	t, ti
ظ	*zo'e*	zx	/z/	z, ze, zu

ع	*'ain*	3	/ɑ/ after a consonant; otherwise /ʔ/, /ə/	a, e, i, o, u, shu
غ	*ghain*	G	/ɣ/	gh, gha, ghe, dhi, ghu
ف	*Fe*	f	/f/	p, f, pa, pi, pe, pu, fa, fi,
ق	*Qāf*	q	/q/	q, qa, qe, qi, qu
ک	*Kāf*	k	/k/	c, k, ca, ce, ci, cu, ka, ke, ki, ku
گ	*Gāf*	g	/ɡ/	g, ga, ge, gi, gu
ل	*Lām*	l	/l/	l, la, le, li, lu
م	*Mīm*	m	/m/	m, ma, me, mi, mu
ن	*Nūn*	n	/n/	n, na, ne, ni, nu
ں	*Nūnghunna*	N	/n/ nasal vowel	N
و	*vā'o*	w/v/o	/v/, /u/, /ʊ/, /o/	o, oo, u, uu, v, w, au
ه ،ہ	*choṭī he*	h	/h/; at end of word /ɑ/	a, aa, h, ha, he, hi, hu
ی	*choṭī ye*	y	/i/, /e/, /ɛ/	e, i, ii, w, y, ai
ے	*ye*	Y	/ɛ ː /	e, ee, y, ai
ؤ	*hamza vā'o*	'w	/ow/	o, u, v, w
ئ	*hamzachoṭīye*	'y	/ɛi/	i, ee, ii, iy
ۓ	*hamzabaṛīye*	'Y	/ɛe ː /	Ay
ۃ	*Gol ṭe*	't	/t/	Not found
ۂ	*hamza he*	'h	/ɛ/ used in compound	Not found
ء	*hamza*	'	/ə/ allograph	Not found
اً	*dozabar*	'a	/ən/, schwa followed by	Not found
◌ٰ	*kharazabar*	'A	long vowel	Not found

References

[1] Ahmed Sabbir Arif and Wolfgang Stuerzlinger. *Analysis of text entry performance metrics.* In Proceedings of the IEEE Symposium on Human Factors and Ergonomics, 2009, pages 100–105.

[2] *Akhbar-e-Urdu* (Special Supplement on Urdu Software; Jan-Feb. 2002), National Language Authority, Islamabad.

[3] Andreas Stolcke. *Entropy-based pruning of back-off language models.* In Proceedings of DARPA Broadcast News Transcription and Understanding Workshop,1998, page 270274.

[4] Andrew Sears, Doreen Revis, Janet Swatski, Rob Crittenden, and Ben Shneiderman. *In vestigating touchscreen typing: The effect of keyboard size on typing speed.* Behaviour and Information Technology, 1993, 12(1):17–22.

[5] Andrew Sears, Julie A. Jacko, Josey Chu, and Francisco Moro. *The role of visual search in the design of effective soft keyboards.* Behaviour and Information Technology, 2001, 20(3):159–166.

[6] Andriy Pavlovych, Wolfgang Stuerzlinger. *Model for non-Expert Text Entry Speed on 12-Button Phone Keypads.* In Proceedings of CHI 2004, Vienna, Austria. 2004.

[7] Animesh, N., Ravikiran Rao, B., Pawandeep, S Sudeep, S. And Ratna, S. *Named Entity Recognition for Indian Languages.* In proceedings of IJCNLP 2008 workshop on NER for South and South-East Asian languages.

[8] Ankrum, D.R. *Viewing Distance at Computer Workstations.* Workplace Ergonomics, September/October, 1996, 10-13.

[9] Asad Habib, Masakazu Iwatate, Masayuki Asahara and Yuji Matsumoto. *Keypad for large letter-set languages and small touch-screen devices (case study: Urdu).* International Journal of Computer Science Issues, Vol. 9, Issue 3, No. 3, May 2012. ISSN: 1694-0814.

[10] Asad Habib, Masakazu Iwatate, Masayuki Asahara and Yuji Matsumoto. *Different Input Systems for Different Devices: Optimized Touch-Screen Keypad Designs for Urdu Scripts*. In proceedings of Workshop on Text Input Methods WTIM2011, IJCNLP, 2011, Chiang Mai, Thailand.

[11] Asad Habib, Masayuki Asahara, Yuji Matsumoto and Kohei Ozaki. *JaPak IEOU: Japan-Pakistan`s Input English Output Urdu (A Case Sensitive Proposed Standard Input System for Perso-Arabic Script clients).* In proceedings of ICIET 2010, Karachi, Pakistan.

[12] Atencio, R. *Eyestrain: the number one complaint of computer users.* Computers in Libraries, 1996, 16(8): 40-44.

[13] August Dvorak, Nellie Merrick, William Dealey, and Gertrude Ford. *Typewriting Behaviour.* American Book Company, New York, USA, 1936.

[14] Baker, Nick. *Why do we all use Qwerty keyboards?* BBC Corporation. 11 August 2010. Retrieved on July 15, 2012.

[15] Beesley, Kenneth R. and Karttunen, Lauri. *Finite State Morphology*. 2003, CSLI Publication, USA.

[16] Bhattcharyya, Pushpak. *Hindi Urdu Machine Transliteration using Finite-state Transducers*. In proceedings of COLING 2008, Manchester, UK. 2008.

[17] Bjorn Hartmann, Meredith Ringel Morris, Hrvoje Benko, and Andrew D. Wilson. *Augmenting interactive tables with mice & keyboards.* In Proceedings of the ACM Symposium on User Interface Software and Technology, PP. 149–152, 2009.

[18] Bonnie Glover Stalls and Kevin Knight. *Translating Names and Technical Terms in Arabic Text.* In Proceedings of 40th Annual Meeting on Association for Computational Linguistics. ACL 2002.

[19] Carsten Schwesig, Ivan Poupyrev and Eijiro Mori. *Gummi: a bendable computer.* In proceedings of the SIGCHI conference on Human factors in computing systems. CHI '04, ACM New York, USA. 2004. PP. 263-270.

[20] Chinnakotla, M. K., Damani, O. P., and Satoskar, A. *Transliteration for resource-scarce languages.* 2010.

[21] Chopde, A. *Printing Transliterated Indian Language Documents- ITRANS* V 5.30. (1991-2001).

[22] Chris Harrison and Scott E. Hudson. *Providing dynamically changeable physical buttons on a visual display.* In Proceedings of the ACM CHI Conference on Human Factors in Computing Systems, PP. 299–308, 2009.

[23] Colin P. Masica, *The In do-Aryan languages.* Cambridge Language Surveys Cambridge: Cambridge University Press, 1993, Cambridge.

[24] Collin Drury and Errol Hoffman. *A model for movement time on data-entry keyboards.* Ergonomics, 35(2):129–147, 1992.

[25] Collins, C., O'Meara, D., and Scott, A.B. *Muscle strain during unrestrained human eye movements*. Journal of Physiology, London, 245, 351 - 369. 1975.

[26] D. S. Katre. *Position Paper On Cross-cultural Usability Issues of Bilingual (Hindi & English) Mobile Phones.* In proc. Indo-Danish HCI Research Symposium, Hosted by Department of Design, Indian Institute of Technology, Guwahati, India. 2006.

[27] Dan Venolia, Forrest Neiberg. *T-Cube: a fast, self-disclosing pen-based alphabet*. In proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence, Boston, Massachusetts, United States, 1994, PP.265-270.

[28] Daniel J. Wigdor. *Chording and Tilting For Rapid, Unambiguous Text Entry to Mobile Phones.* Master's thesis, 2004, University of Toronto, Canada.

[29] Daniel Vogel and Patrick Baudisch. *Shift: a technique for operating pen-based interfaces using touch.* In Proceedings of the ACM CHI Conference on Human Factors in Computing Systems, PP. 657–666, 2007.

[30] Daniel Wigdor, Gerald Penn, Kathy Ryall, Alan Esenther, and Chia Shen. *Living with a tabletop: Analysis and observations of long term office use of a multitouch table.* In Proceedings of the IEEE Workshop on Tabletop Displays, PP. 60–67, 2007.

[31] Dario D. Salvucci , John R. Anderson. *Intelligent gaze-added interfaces*. In proceedings of the SIGCHI conference on Human factors in computing systems, The Hague, Netherlands, 2000, PP. 273-280.

[32] David Goldberg and Cate Richardson. *Touch-typing with a stylus.* In Proceedings of the ACM INTERCHI Conference on Human Factors in Computing Systems, PP. 80–87, 1993.

[33] David J. C. MacKay. *Information theory, inference, and learning algorithms.* Cambridge University Press, ISBN 978-0-521-64298-9, 6 October 2003, PP. 119.

[34] David J. Ward, Alan F. Blackwell and David J. C. MacKay. *Dasher - a data entry interface using continuous gestures and language models*. In proceedings of the 13th annual ACM symposium on User interface software and technology, ACM New York, NY, USA, 2000, PP. 129 - 137.

[35] David J. Ward, Alan F. Blackwell & David J. C. MacKay. *Dasher: A Gesture-Driven Data Entry Interface for Mobile Computing*. Human–Computer Interaction, 2002, Volume 17, Issue 2-3, Special Issue: Text Entry for Mobile Computing. PP. 199-228.

[36] Donald A. Norman and Diane Fisher. *Why alphabetic keyboards are not easy to use: Keyboard layout doesn't much matter.* Human Factors, 24(5):509–519, 1928.

[37] Edgar Matias. *Half-QWERTY: Typing With One Hand Using Your Two-handed Skills.* In Proceedings of Interactive Experience CHI`94, Boston,Massachusetts USA. 1994.

Edgar Matias, I.Scott MacKenzie, William Buxton. *Half-QWERTY: A One-handed keyboard Facilitating Skill Transfer From QWERTY.* INTERCHI`93, 1993, PP.88 - 94.

[38] Ehara, Yo. and Kumiko Tanaka-Ishii. *Multilingual text entry using automatic language detection.* In proceedings of IJCNLP 2008.

[39] Encyclopaedia Britannica online. *Arabic Alphabet*. Retrieved on July 15, 2012. http://www.britannica.com/eb/article-9008156/Arabic-alphabet

[40] Fisher, R.F. *The force of contraction of the human ciliary muscle during accommodation*. Journal of Physiology, London, 1977, Vol 27, 51 – 74.

[41] G. Reitmayr and D. Schmalstieg. OpenTracker. *An open software architecture for reconfigurable tracking based on XML.* In Proceedings of IEEE Virtual Reality 2001, Yokohama, Japan, March 13–17 2001, PP. 285–286.

[42] Garrett, Matthew, Ward, David, Murray, Iain, Cowans, Phil, Mackay, David. *Implementation of Dasher, an information efficient input mechanism*. UKUUG Linux, Edinburgh, Scotland, 2003.

[43] George Weber. *The World's 10 most influential Languages. American Association of Teachers of French*

(ATTF), National Bulletin, 1999, vol. 24, 3:22-28. http://www.andaman.org/BOOK/reprints/weber/rep-weber.htm (Retrieved on July 15, 2012)

[44] Gerhard Reitmayr, Chris Chiu, Alexander Kusternig, Michael Kusternig, Hannes Witzmann, iOrb - U*nifying Command and 3D Input for Mobile Augmented Reality.* Vienna University of Technology. (2005).

[45] Google Transliteration Labs, Retrieved on July 15, 2012. http://www.google.com/transliterate/urdu.

[46] Government of Pakistan, National Language Authority (Cabinet Division), Islamabad, Pakistan; Retrieved on July 15, 2012. http://www.nla.gov.pk

[47] Government of Pakistan, National Language Authority (Cabinet Division), Jan 26, 2004, Islamabad, Pakistan, Retrieved on July 15, 2012. http://www.nla.gov.pk/beta/images/alphabetsfull.gif

[48] Government of Pakistan, Population Census Organization, Islamabad Pakistan, Retrieved on July 15, 2012. http://www.statpak.gov.pk/depts/pco/statistics/area_pop/area_pop.html

[49] Government of Pakistan, Population Census Organization, Islamabad, Pakistan; Retrieved on July 15, 2012. http://www.statpak.gov.pk/depts/pco/statistics/other_tables/pop_by_mother_tongue.pdf

[50] Grimes, Barbara F. Pakistan. *Ethnologue: Languages of the World,* 14th Edition Dallas, Texas; Summer Institute of Linguistics, 2000,pp: 588- 598.

[51] Haizhou Li, A Kumaran, Min Zhang, and Vladimir Pervouchine. *Report of NEWS 2009 machine transliteration shared task.* Named Entities Workshop --- Shared Task on Transliteration. In Proc. Named Entities Workshop at ACL-IJCNLP 2009.

[52] Hussain, Sarmad. *Letter to Sound Rules for Urdu Text to Speech System.* In proceedings of Workshop on Computational Aproaches to Arabic Script-based Languages, COLING 2004, Geneva, Switzerland.

[53] J. R. Wolpaw, N. Birbaumer, W. J. Heetderks, D. J. McFarland, P. H. Peckham, G. Schalk, et al. *Brain-computer interface technology: a review of the first international meeting*. IEEE Transactions on Rehabilitation Engineering, 2000, Vol. 8, PP. 164-73.

[54] J. Viterbi. *Error bounds for convolutional codes and an asymptotically optimum decoding algorithm.* IEEE Transactions on Information Theory, 1967, IT-13:260-269.

[55] J. Ward, A. F. Blackwell, and D. J. C. MacKay. *Dasher - A Data Entry Interface Using Continuous Gestures and Language Models*. In proceedings of UIST 2000: The 13th Annual ACM Symposium on User Interface Software and Technology, 2000.

[56] Jacob O. Wobbrock, Duen Horng Chau and Brad A. Myers. *An Alternative to Push, Press, and Tap-tap-tap: Gesturing on an Isometric Joystick for Mobile Phone Text Entry.* In Proceedings of Mobile Interaction Techniques 1, CHI 2007, San Jose, CA,USA. 2007.

[57] Jaschinski-Kruza, W. *Visual strain during VDU work: the effect of viewing distance and dark focus.* Ergonomics 31, 1988, pp: 1449 – 1465.

[58] Jawaid, Bushra and Tafseer Ahmed. *Hindi to Urdu Conversion: Beyond Simple Transliteration.* In proceedings of Conference on Language & Technology, Lahore, Pakistan, 2009.

[59] Jefferson Y. Han. *Low-cost multi-touch sensing through frustrated total internal reflection.* In Proceedings of the ACM Symposium on User Interface Software and Technology, 2005,PP. 115–118.

[60] Jennifer Mankoff and Gregory D. Abowd. *Cirrin: A word-level unistroke keyboard for pen input.* In Proceedings of the ACM Symposium on User Interface Software and Technology, 1998, PP. 213–214.

[61] Johannes Hirche, Peter Bomark, Mikael Bauer, and Pawel Solyga. *Adaptive interface for text input on large-scale interactive surfaces.* In Proceedings of the IEEE Workshop on Tabletop Displays, 2008,PP. 153–156.

[62] Junichi Aoe. *An efficient digital search algorithm by using a double-array structure.* IEEE Trans. Softw. Eng., 1989, 15:1066–1077.

[63] K. Knight and J. Graehl. *Machine Transliteration.* Computational Linguistics, Volume 24 Issue 4, MIT Press Cambridge, MA, USA, 1998,pp: 599-612.

[64] K.Zia. *Standard Code Table for Urdu*. 4th Symposium on Multilingual Information Processing (MLIT-4), Yangon, Myanmar.

[65] Karl-Heinz Best. *Otto Behaghel (1854-1936).* In: Glottometrics 14. 2007, 80-86.

[66] Kathy Ryall, Clifton Forlines, Chia Shen, Meredith Ringel Morris, and Katherine Everitt. *Experiences with and observations of direct-touch tabletops.* In Proceedings of the IEEE Workshop on Tabletop Displays, 2006, PP. 89–96.

[67] Kellogg, Rev. S. H. *A Grammar of Hindi Language, Delhi, Oriental Book reprints*,1872.

[68] Ken Perlin. *Quikwriting: Continuous stylus-based text entry.* In Proceedings of the ACM Symposium on User Interface Software and Technology,1998, PP. 215–216.

[69] Kenichi Tanaka. *Japan - the strange country*. Final thesis project, California college of Arts (CCA), San Fransisco, USA. 2008.

[70] Kenneth Church, Ted Hart, and Jianfeng Gao. *Compressing Trigram Language Models With Golomb Coding.* In proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, PP. 199–207, Prague, Association for Computational Linguistics, June 2007.

[71] Kevin Curran, Derek Woods, Barry O Riordan. *Investigating text input methods for mobile phones.* Telematics and Informatics, vol.23, 2006, PP.1 - 21.

[72] Khan, Mehboob Alam. *اُردو کا صوتی نظام (Sound System in Urdu).* National Language Authority,1997 .

[73] Knight, K. and Stall, B. G. *Transliterating Names and Technical Terms in Arabic Text.* In proceedings of COLING/ACL Workshop on Computational Approaches to Semitic Languages. 1998.

[74] K¨olsch, M. Turk, and T. H¨ollerer. *Vision-based interfaces for mobility*. International Conference on Mobile and Ubiquitous Systems (MobiQuitous), 2004.

[75] Kubler, F. Nijboer, J. Mellinger, T. M. Vaughan, H. Pawelzik, G. Schalk. *Patients with ALS can use sensorimotor rhythms to operate a brain-computer interface*. Neurology, 2005, Vol. 64, PP. 1775-1777.

[76] Leah Findlater, Jacob O. Wobbrock, and Daniel Wigdor. *Typing on flat glass: examining ten-finger expert typing patterns on touch surfaces.* In Proceedings of the ACM Conference on Human Factors in Computing Systems, 2011, PP. 2453–2462.

[77] Leanna Gingras, Michelle Escobar, Eileen Quintero, Tanuj Shah. *Comparative Analysis of E-readers: The Sony Reader* PRS-505. School of Information University of Michigan, USA, 2008.

[78] Leonard J. West. *The Standard and Dvorak Keyboards Revisited: Direct Measures of Speed.* Technical report, Santa Fe Institute, New Mexico, USA, 1998.

[79] Lewis M. Paul (ed.). *Ethnologue: Languages of the World*. Sixteenth edition, 2009, Dallas, TX, USA. SIL International. Online version: http://www.ethnologue.org/ethno_docs/distribution.asp?by=size (Retrieved on July 15, 2012).

[80] Liebowitz, Stan J.; Stephen E. Margolis. *The Fable of the Keys.* Journal of Law & Economics, April 1990, 33 (1): 1–25. DOI:10.1086/467198.

[81] Lourdes Torres, *Code-mixing as a narrative strategy in a bilingual community.* World Englishes, Volume 11, Issue 2-3, PP. 183–193, DOI: 10.1111/j.1467-971X.1992.tb00063.x, July 1992.

[82] M. A. K. Azad, R. Sharmeen, S. Ahmad and R. Mahmud. *Frequency and Flexibility Based Cell Phone Keypad Layout.* TENCON 2005 IEEE Region 10, Nov. 2005, PP. 1-5.

[83] M. Afzal, S. Hussain. *Urdu Computing Standards: Development of Urdu Zabta Takhti (UZT 1.01).* In proceedings of IEEE International Multi-topic Conference, Pakistan, 2001 , pp: 216-222.

[84] M. Amir K., M.A. Khan. A.Habib and M.N. Ali. *Corps based mapping of Urdu characters for cell phones.* CLT 09, Lahore,2009

[85] M. Humayoun, H. Hammarström, and A. Ranta. *Urdu Morphology, Orthography and Lexicon Extraction.* In Proceedings of 2nd Workshop on Computational Approaches to Arabic Script-based Languages (CAASL), LSA, 2007.

[86] M. Ijaz and S. Hussain. *Corpus Based Urdu Lexicon Development*. In proceedings of the conference on Language & Technology (CLT07), Bara Gali Summer Campus, University of Peshawar, August 7-11, 2007, PP. 85-94.

[87] M.Ijaz, S.Hussain. *Corpus Based Urdu Lexicon Development*. In proceedings of CLT07 Pakistan, 2007.

[88] M. Tregoubov and N. Birbaumer. *On the building of binary spelling interfaces for augmentative communication*. IEEE Trans. Biomed. Eng., 2005, vol. 52, PP. 300.

[89] Malik, L. Besacier, C. Boitet and P. Bhattacharyya. *A hybrid Model for Urdu Hindi Tranliteration.* In proceedings of Association for Computational Linguistics, International Joint Conference on Natural Language Processing (ACL-IJCNLP), 2009, Suntec, Singapore.

[90] Malte Weiss, Julie Wagner, Yvonne Jansen, Roger Jennings, Ramsin Khoshabeh, James D. Hollan, and Jan O. Borchers. *SLAP widgets: bridging the gap between virtual and physical controls on tabletops*. In Proceedings of the ACM CHI Conference on Human Factors in Computing Systems, 2009,PP. 481–490.

[91] Maosong Sun Yabin Zheng, Chen Li. *Chime: An efficient error-tolerant Chinese pinyin input method*. In Proceedings of IJCAI,2011.

[92] Mark D. Dunlop and Finbarr Taylor. *Tactile Feedback for Predictive Text Entry.* In proceedings of Conference on Human Factors in Computing Systems, 2009, Boston, MA, USA.

[93] Miika Silfverberg , I. Scott MacKenzie , Panu Korhonen, *Predicting text entry speed on mobile phones, Proceedings of the SIGCHI conference on Human factors in computing systems*. The Hague, Netherlands, 2000, PP. 9-16.

[94] Mitesh M. Khapra, Pushpak Bhattacharyya. *Improving transliteration accuracy using word-origin detection and lexicon lookup*. In proceedings of NEWS'09, The 2009 Named Entities Workshop: Shared Task on Transliteration. PP. 84-87.

[95] Montaut, A. *A Linguistic Grammar of Hindi Studies.* In Indo-European Linguistics Series, Munchen, Lincom Europe, 2004

[96] Mostofa Kamal Nasir, Md Rashadul Hasan Rakib, Md. Sazzad Hossain, *and* Md. Belayet Ali. *T9 Method Implementation and Bangla Text Input Technique for Short Message Service on Mobile Devices*. International Journal of Computer and Information Technology, vol. 02, Issue 02. 2012.

[97] Muysken, P. *Bilingual Speech-A typology of code-mixing.* Cambridge University Press, Cambridge, 2000.

[98] N. Durrani and S. Hussain. *Urdu Word Segmentation.* In proceedings of 11th Annual Conference of North American Chapter of the Association for Computational Linguistics, Human Language Technologies (NAACL-HLT), Los Angeles, California, USA,2010.

[99] NASA, NASA-STD-3000. *Man Systems Integration Standards.* Volume I - Standards and Volume II, Revision B, National Aeronautics and Space Administration, Houston, USA, 1995.

[100] Nathaniel Rochester, Frank C. Bequaert, and Elmer M. Sharp. *The chord keyboard.* Computer, December 1978, 11(12):57–63.

[101] Noyes, Jan. *The QWERTY keyboard: a review.* International Journal of Man-Machine Studies. Volume 18, Issue 3, March 1983, PP. 265–281. DOI:10.1016/S0020-7373(83)80010-8

[102] Nuray Aykin, Pia Honold Quaet-Faslem and Allen E. Milewski. *Cultural Ergonomics.* Handbook of Human Factors and Ergonomics, Third Edition, 2006, PP. 177–190.

[103] Otto Behaghel. *Beziehungen zwischen Umfang und Reihenfolge von Satzgliedern. In: Indogermanische Forschungen.* 25, 1909, 110-142.

[104] Owens, D.A., Wolfe-Kelly, K. *Near Work, Visual Fatigue, and Variations of Oculomotor Tonus*. Investigative Ophthalmology and Visual Science. 28, 1987,743 - 749.

[105] P. O. Krestensson. *Five Challenges for Intelligent Text Entry Methods.* In proceedings of Association for the Advancement of Artificial Intelligence, Winter, 2009.

[106] Paola, V. and Sanjeev, K. *Transliteration of Proper Names in Cross-language Application.* In proceedings of 26th International ACM SIGIR Conference on Research and Development in Information Retrieval, Toronto, Canada, 2003.

[107] Paul H. Dietz and Darren Leigh. *DiamondTouch: A multi-user touch technology.* In Proceedings of the ACM Symposium on User Interface Software and Technology, 2001,PP. 219–226.

[108] Paul Holleis, Friderike Otto, Heinrich Hufmann, Albrecht Schmidt. *Keystroke-Level Model for Advanced Mobile Phone Interaction.* In Proceedings of CHI 2007, San Jose, CA, USA, 2007.

[109] Per-Ola Kristensson and Shumin Zhai. *SHARK2: a large vocabulary shorthand writing system for penbased computers.* In Proceedings of the ACM Symposium on User Interface Software and Technology, 2004,PP. 43–52.

[110] Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. *Class-based n-gram models of natural language.* Computational Linguistics, 1992,18:467–479.

[111] Petra Isenberg and M. Sheelagh T. Carpendale. *Interactive tree comparison for co-located collaborative information visualization.* IEEE Transactions on Visualization and Computer Graphics, 2007,13(6):1232–1239.

[112] R. William Soukoreff and I. Scott MacKenzie. *Metrics for text entry research: an evaluation of MSD and KSPC, and a new unified error metric.* In Proceedings of the ACM CHI Conference on Human Factors in Computing Systems, 2003,PP. 113–120.

[113] R. William Soukoreff and I. Scott MacKenzie. *Theoretical upper and lower bounds on typing speed using a stylus and a soft keyboard.* Behaviour and Information Technology, 1995, 14(6):370–379.

[114] Rai and Alok. *Hindi Nationalism.* Orient Longman Private Limited, New Delhi, 2000.

[115] Richard Barrett, Michael Berry, Tony Chan, James, Demmel, June Donato, Jack Dongarra, Victor Eijkhout, Roldan Pozo, Charles Romine, Henk van der, Vorst, and Long Restart. *Templates for the solution of linear systems: Building blocks for iterative methods.* 1993.

[116] Sowmya V. B., Monojit Choudhury, Kalika Bali, Tirthankar Dasgupta, and Anupam Basu. *Resource creation for training and testing of transliteration systems for Indic languages*. In proceedings of Language Resources and Evaluation Conference (LREC), 2010.

[117] Robert W. Proctor and Kim-Phuong L. Vu. *Selection and Control of Action.* Handbook of Human Factors and Ergonomics, Third Edition, Wiley Online Library. 2006, PP. 89–110.

[118] Romaine, Suzanne and Braj Kachru. *Code mixing and code-switching.* In T. McArthur (ed.), The Oxford Companion to the English Language.Oxford University Press. PP. 228-2291992.

[119] S. A. Wills and D. J. C. MacKay. *Dasher - An efficient writing system for brain-computer interfaces?* IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2006, vol. 14, PP. 244-246.

[120] S. A. Hussain, Asma Sajjad, Fareeha Anwar. *Online Urdu Character Recognition System.* In proceedngs of Conference on Machine Vision Applications, MVA 2007, Tokyo Japan.

[121] S. Hussain. *Letter-to-Sound Conversion for Urdu Text-to-Speech System.* Center for Research in Urdu Language Processing (CRULP), Lahore, Pakistan, 2004.

[122] S. Hussain, Sana Gui, Afifah Waseem. *Urdu Encoding and Collation Sequence for Localization.* Center for Research in Urdu Language Processing National University of Computer and Emerging Sciences.

[123] S. MacKenzie and S. R. William. *Text Entry for Mobile Computing: Models and Methods, Theory and Practice.* Human Computer Interaction Volume 17, Lawrence Erlbaum Associates Inc, 2002, PP. 147–198

[124] S Ober, *Relative Efficiencies of the Standard and Dvorak Simplified Keyboards.* Education Resources Information Center, Delta Pi Epsilon Journal, 1993, v35 n1 p1-13.

[125] Sandeva, G., Hayashi, Y., Itoh,Y., and Kishino, F. *SriShell Primo: A Predictive Sinhala Text Input System.* In proceedings of IJCNLP 2008 workshop on NLP for less privileged languages.

[126] Sara A. Bly. *A use of drawing surfaces in different collaborative settings.* In Proceedings of the ACM Conference on Computer-Supported Cooperative Work, 1988, PP. 250–256.

[127] Saul Greenberg and David Marwood. *Real time groupware as a distributed system: Concurrency control and its effect on the interface.* In Proceedings of the ACM Conference on Computer-Supported Cooperative Work, 1994, PP. 207–217.

[128] Scott A. Golder and Bernardo A. Huberman. *Usage patterns of collaborative tagging systems*. Journal of Information Science, April 2006, 32(2):198–208.

[129] Scott MacKenzie and R. William Soukoreff. *Phrase sets for evaluating text entry techniques.* In Extended Abstracts of the ACM CHI Conference on Human Factors in Computing Systems, 2003 ,PP. 754–755.

[130] Scott MacKenzie and R.William Soukoreff. *Phrase Sets for Evaluating Text Entry Techniques.* In Proceedings of CHI 2003, Ft. Lauderdale, Florida, USA. 2003.

[131] Scott MacKenzie and R. William Soukoreff. *Text entry for mobile computing: Models and methods, theory and practice.* Human-Computer Interaction, 2002, 17(2/3):147–198.

[132] Scott MacKenzie and Shawn X. Zhang. *The design and evaluation of a high-performance soft keyboard.* In Proceedings of the ACM CHI Conference on Human Factors in Computing Systems, 1999, pp: 25–31.

[133] Scott MacKenzie and Shawn X. Zhang. *An empirical investigation on the novice experience with soft keyboards.* Behaviour and Information Technology, 2001, 20(6):411–418.

[134] Scott MacKenzie and Shawn X. Zhang. *The immediate usability of Graffiti.* In Proceedings of Graphics Interface, 1997,PP. 129–137.

[135] Scott McCarley, J. *Cross language name matching.* In proceedings of ACM-SIGIR2009 Shared Task on Transliteration (NEWS 2009), ACL.

[136] Scott MacKenzie, Shawn X. Zhang, and R. William Soukoreff. *Text entry using soft keyboards.* Behaviour and Information Technology, 1999, 18(4): 235–244.

[137] Shadi Ganjavi, Panayiotis G.Georgiou, Shrikanth Narayanan. *A Transcription Scheme for Languages Employing the Arabic Script Motivated by Speech Processing Application.* In Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages. 20th International Conference on Computational Linguistics, COLING 2004.

[138] Shaun K. Kane. *Context-Enhanced Interaction Techniques for More Accessible Mobile Phones.* SigAccess Newsletter, Issue 93, 2009. PP.39 - 43.

[139] Shinsuke Mori, Daisuke Takuma, and Gakuto Kurata. *Phoneme-to-text transcription system with an infinite vocabulary.* In Proceedings of ACL, ACL- 44, 2006, PP. 729–736.

[140] Shumin Zhai and Per-Ola Kristensson. *Shorthand writing on stylus keyboard.* In Proceedings of the ACM CHI Conference on Human Factors in Computing Systems, PP. 97–104, 2003.

[141] Shumin Zhai, Alison E. Sue, and Johnny Accot. *Movement model, hits distribution and learning in virtual keyboarding.* In Proceedings of ACM CHI Conference on Human Factors in Computing Systems, 2002, PP. 17–24.

[142] Shumin Zhai, Michael A. Hunter, and Barton A. Smith. *The Metropolis keyboard - an exploration of quantitative techniques for virtual keyboard design.* In Proceedings of the ACM Symposium on User Interface Software and Technology, 2000,PP. 119–128.

[143] Shumin Zhai, Per Ola Kristensson, and Barton A. Smith. *In search of effective text input interfaces for off the desktop computing.* Interacting with Computers, 2005, 17(3):229–250.

[144] Strong, E.P. *A Comparative Experiment in Simplified Keyboard Retraining and Standard Keyboard Supplementary Training.* Washington, D.C.: General Services Administration, 1956.

[145] Sungahn Ko, KyungTae Kim, Tejas Kulkarni, Niklas Elmqvist. *Applying Mobile Device Soft Keyboards to Collaborative Multitouch Tabletop Displays: Design and Evaluation.* In proceedings of ACM International Conference on Interactive Tabletops and Surfaces (ITS), Kobe, Japan, 2011.

[146] T. H. Shorrock , D. J. C. MacKay and C. J. Ball J. Winkler , N. D. Lawrence and M. Niranjan. *Deterministic and statistical methods in machine learning*. In proceedings of the Sheffield Machine Learning

Workshop, 2005 : Springer, vol. 3635, PP.88.

[147] T. Rahman. *Language Policy and Localization in Pakistan: Proposal for a Paradigmatic Shift.* Crossing the Digital Divide, Khatmandu, Nepal, 2004.

[148] Taku Kudo, Toshiyuki Hanaoka, Jun Mukai, Yusuke Tabata, and Hiroyuki Komatsu (Google Japan Inc). *Efficient dictionary and language model compression for input method editors.* WTIM, IJCNLP 2011, Chiang Mai, Thailand.

[149] The daily Express Lahore, http://express.com.pk/epaper/ Retrieved on July 15, 2012.

[150] The daily Jang, Jang Group of Newspapers, Pakistan, http://www.jang.com.pk/jang/apr2010-daily/09-04-2010/paper.html,

[151] The daily Mashriq, http://www.dailymashriq.com.pk/mashriq.html Retrieved on July 15, 2012.

[152] *The Oxford Companion to the English Language.* Oxford University Press. PP. 228-229

[153] *The Urdu Qaeda for Primary School Children*, Feroz Sons publishers, Lahore, Pakistan.

[154] Timothy C. Bell , John G. Cleary , Ian H. Witten. *Text Compression*. Prentice-Hall, Inc. NJ, 1990.

[155] Toni Schmidt, Werner A. K¨onig, and Harald Reiterer. *Text input on multitouch tabletop displays.* In Poster Proceedings of the IEEE VGTC/Eurographics Symposium on Visualization, 2009.

[156] Toshiyuki Masui. *An Efficient Text Input Method for Pen-based Computers.* In Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI`98), ACM press, 1998.

[157] Toshiyuki Masui. *Integrating Pen Operations for Composition by Example.* In Proceedings of ACM UIST'98, 1998.

[158] Toshiyuki Masui. *POBox: An Efficient Text Input Method for Handheld and Ubiquitous Computers.* Springer, Handheld and Ubiquitour Computing, 1999, Volume 1707/1999, PP.289-300.

[159] U.Afzal, N.I.,Rao, A.M.Sheri. *Adaptive Transliteration Based on Cross-Script Trie Generation: A case of Roman-Urdu.* CLT-09 Lahore, Pakistan.

[160] Umair Z Ahmed Kalika Bali Monojit Choudhury Sowmya VB (Microsoft Research Labs India, Bangalore). *Challenges in Designing Input Method Editors for Indian Languages: The Role of Word-Origin and Context.* In Proceedings of the Workshop on Advances in Text Input Methods (WTIM 2011), Chiang Mai, Thailand, November 2011.

[161] Unicode. 1991-2001. Unicode Standard version. Online version: http://unicode.org/charts/PDF/U0600.pdf

[162] Uta Hinrichs, Holly Schmidt, Tobias Isenberg, Mark Hancock, and Sheelagh Carpendale. *BubbleType: Enabling text entry within a walk-up tabletop installation.* Technical report, University of Calgary, 2008.

[163] Uta Hinrichs, Mark S. Hancock, M. Sheelagh T. Carpendale, and Christopher Collins. *Examination of text-entry methods for Tabletop displays.* In Proceedings of the IEEE Workshop on Tabletop Displays, 2007, PP. 105–112.

[164] W. Sellers and E. Donchin. *A P300-based brain-computer interface: Initial tests by ALS patients.* Clinical Neurophysiology, 2006, vol. 117, PP. 538-548.

[165] Waldemar Karwowski. *The Discipline of Ergonomics and Human Factors. Chapter-1, Handbook of Human Factors and Ergonomics,* Third Edition, 2006, PP. 1-31.

[166] Wolfgang Jaschtnski-Kruza. *On the preferred viewing distances to screen and document at VDU workplaces*. Ergonomics, Volume 33, Issue 8 August 1990 , PP. 1055 - 1063

[167] Yeh, Chung-chih, WU, Chiao-ying, Tu-Cheng, TW. *Handheld device and text input method.* Hon Hai Precision Industry Company Limited, USA. 2012.

[168] Yuhong Yang. *Information Theory, Inference, and Learning Algorithms.* Journal of the American Statistical Association, 2005, Volume 100, Issue 472, PP. 1461-1462.

[169] Yvonne Rogers and Sian E. Lindley. *Collaborating around vertical and horizontal large interactive displays: Which way is best?* Interacting with Computers, 2004, 16(6):1133–1152.

[170] Zheng Chen and Kai-Fu Lee. *A new statistical approach to Chinese pinyin input*. In Proceedings of ACL 2000 Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, Stroudsburg, PA, USA, 2000, PP. 241-247.

Publication List

ASAD HABIB
Student No. 0961209
Computational Linguistics Laboratory

Journal Papers

Asad Habib, Masakazu Iwatate, Masayuki Asahara and Yuji Matsumoto. *Keypad for large letter-set languages and small touch-screen devices (case study: Urdu).* International Journal of Computer Science Issues, Vol. 9, Issue 3, No. 3, May 2012. ISSN: 1694-0814. URL http://www.ijcsi.org/papers/IJCSI-9-3-3-47-58.pdf, Dissertation chapter 4.

Refereed International Conferences and Workshops

Asad Habib, Masakazu Iwatate, Masayuki Asahara and Yuji Matsumoto. *Different input systems for different devices: Optimized touch-screen keypad designs for Urdu scripts.* In proceedings of workshop on text input methods WTIM2011, IJCNLP, 2011, Thailand. URL http://acl.eldoc.ub.rug.nl/mirror/W/W11/W11-35.pdf#page=40, Dissertation chapter 4.

Asad Habib, Masayuki Asahara, Yuji Matsumoto and Kohei Ozaki. *JaPak IEOU: Japan-Pakistan's input English output Urdu (a case sensitive proposed standard input system for Perso-Arabic script clients).* In proceedings of ICIET 2010, Karachi, Pakistan.URLhttp://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5625679&abstractAccess=no&userType=inst, Dissertation chapter 5.

M.Amir K. M.A. Khan. Asad Habib and M. N. Ali. *Corpus Based Mapping of Urdu Characters for Cell Phones*. 2009, CLT09 Lahore, Pakistan. URL http://crulp.org/clt09/download/Papers/Paper18.pdf, Dissertation chapter 4.

Other Publications

Two domestic workshop papers.