

Multilingual Fonts for Arabic Script

Jamil Khan*

Department of Computer Science, University of Peshawar, Pakistan

Abstract

This paper is about developing a multi-lingual font for Pashto, Arabic, Urdu and Persian in such a way that the text for these four languages can be typed in it. Main purpose for developing fonts was to handle and remove many problems faced in earlier fonts. Fonts developed as a part of this work are ligature-based and text for all four languages (Pashto, Arabic, Urdu, and Persian) can be written in a single font. The paper is structured as follows:

Shortcomings of existing fonts based on Arabic script and some text editors are discussed in Section-I. Section-II discusses the proposed solution to all those limitations. In Section-III, is about implementation of suggested solution, and Section-IV discusses features of the suggested solution. Section-V concludes the paper and Section-VI discusses future work discussed for Arabic typography.

Keywords: Component; Formatting; Style; Styling; Insert

Introduction

During this work, some existing fonts based on Arabic script, including Pashto, Urdu, and Persian, are used in different bilingual text editors, and their limitations are pointed out. These fonts are studied thoroughly to get a clear understanding of what things are done and how.

Requirements for this new approach are defined. These requirements stated which changes are necessary to eliminate the problems identified in the initial analysis. Some of the available text editors are discussed in the following lines:

Inpage 2.40

Inpage 2.40 is a page making software for Urdu and related languages, introduced in 1994. It is a multi-lingual word processor, supports Urdu, Arabic, English, Hazaragi, Kashmiri, Pashto, Persian and Sindhi languages. Inpage works almost similar to MS-Word, provides facilities to create a document, to draw tables and to split page setting into columns. Inpage has still some problems to be faced by every user: [1] No support to use .ttf or .otf fonts for Urdu, Pashto, Arabic and Persian Text. It has its own font family (.utf) for Arabic script [2]. Each Inpage font has a limited number of glyphs for all the letters to make a complete text in Arabic script [3]. Only required ligatures (such as lam-alef áÇ and lam-lam-ha ááá (lillah)) are added in Arabic fonts. Inpage does not have proper fonts for Pashto script [4]. Pashto fonts are designed by some font developers in such a way that Urdu letters are replaced by Pashto letters such that Urdu letter is replaced by Pashto letter and this technique is applied for all letters. No support to install new fonts for Arabic script [5]. Although some font makers (designers) replaced the original Inpage fonts with new typefaces but it is not a legal achievement because the old typefaces have been removed due to this action [6]. For example, the Old "Trad Arabic" font has been replaced by another font. Only the type faces of the font has been changed, the name of the font is the same as old font. The samples are given below: (Figure 1).

The first line in Urdu is typed in Inpage's original font named "Trad Arabic" while second line is modified form of the original "Trad Arabic" font and these typefaces are taken from another font "Nafees Web Naskh". Inpage fonts have no proper setting for mark positioning in Arabic text. No ayah signs are added. The limitations for Quranic verses in Inpage are: (a) Improper Mark Positioning (b) Only one Madda mark is added, the other one is missing (c) No ayah sign (d) All the ramooz-o-awqaf of Quranic verses are ignored, which are typed

manually in Inpage or sometimes with the help of other software such as CorelDraw.

Liwal pashto system for windows

Liwal Pashto System-introduced in 2004 for Windows-98 and used widely. This system was first windows based Pashto system, which enabled users to write Pashto text in most of the word processors (e.g. Microsoft Word, Notepad, Wordpad, and CorelDraw). After a couple of years, Liwal introduced another Pashto System for Windows XP, and was appreciated a lot. The system has 22 different fonts for Pashto, Persian and Arabic text, uses windows fonts, so one can easily install new fonts and can use those fonts for their purposes. But it has still some problems: [1] a single font cannot be used for multiple languages to type text (Figure 2).

The underlined letters in Figure 2 are Pashto letters, while the text

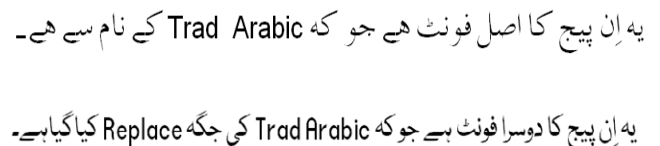


Figure 1: Original Trad Arabic font is replaced by another font.

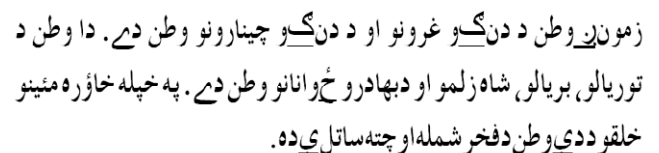


Figure 2: Liwalarabic font with missing pashto letters.

*Corresponding author: Jamil Khan, Department of Computer Science, University of Peshawar, Pakistan, Tel: +92 91 9216701; E-mail: jamilkhan@upesh.edu.pk

Received May 23, 2015; Accepted August 22, 2015; Published September 10, 2015

Citation: Khan J (2015) Multilingual Fonts for Arabic Script. J Inform Tech Softw Eng 5: 154. doi:10.4172/2165-7866.1000154

Copyright: © 2015 Khan J. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

is typed in an Arabic font, so these letters are missing from the font because Pashto supported letters are not included. Liwal fonts are purely for Arabic, Pashto and Persian, while Urdu language is ignored [2]. If a mixed text is required to be typed, then in this situation, what has to be done? The answer is very simple; the system fails in this situation. Islamic symbols are excluded from all the fonts [3]. A separate font for these symbols is used or sometimes the initial letters of each Islamic symbol are written in brackets in front of its corresponding words such as Çááá(İ), áġġ(Ö), ÇÈæÈBÑ(ÑÖ), ÍaiÝä(ÑÍ) and so on, which is not a proper representation of these symbols. Proper Unicode codes should be included for these symbols from ‘Arabic Presentation Forms-A’ such as FDF2, FDF4, FDFA, FDFB, and FDFD in a font, because these symbols are not used only in Arabic text but also in Urdu, Pashto and Persian languages. Limited number of glyphs and no use of ligatures, maximum four possible shapes for each character are included, no support for alternate shapes [3]. This situation can be handled in two ways that is, either ligatures must be used or the ‘calt’ (contextual alternates) feature must be added into the font. More simply, opentype features and opentype tables must be included into the font to behave in different situations with different results only for a single character. Figure 3 is a representation of alternate shapes of a character (Figure 3).

There must be such a system which can cover all the requirements of typing text in a language that should have a good looking typeface, must have all the possible characters in different styles and shapes, and there must be no loss of memory. Each font must have to support text in different languages of Arabic script and support for Quranic verses as well [7]. These problems can be solved by developing Unicode based fonts. Some extra ordinary features must be added to each font in order to overcome the earlier problems as possible [8].

The Proposed Approach

A possible attempt has been taken to develop such fonts that can be used for each of the discussed languages (such as *Pashto, Arabic, Urdu* and *Persian*). Hence these fonts are Unicode based and can be used in Microsoft Word.

Typesetting process of Arabic script is more complex than typesetting using Latin script. Arabic script has independent contextual rules. It is cursive and its characters are joining characters i.e. one or more character(s) may join the other(s) either from one side or from both sides (right and left) to form a complete and a meaningful word. The Arabic alphabet has much of the characters joining from its both sides, right and left except for 6 characters that join only to the preceding ones. Additionally, 4 characters from Pashto (ب، پ، ت، ث)، 2 characters from Urdu (ج، ش)، and 2 characters from both Pashto and Urdu (ژ، ځ) are also those characters that join only to the preceding ones. Arabic script has four maximum shapes for a single character. These shapes are mainly known as: stand alone (isolated), connecting previous (final), connecting following (initial) and connecting both (medial). More specifically, all the characters can be divided into two categories: Dual-joining and Right-joining. Dual-joining characters have 4 different glyphs for their representation: isolated, initial,

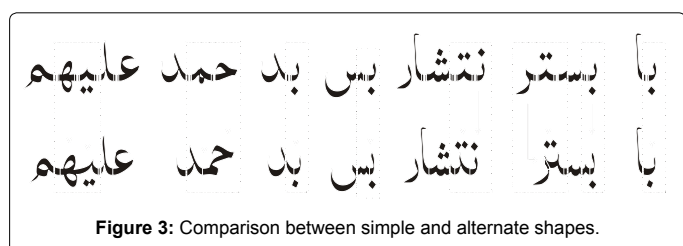


Figure 3: Comparison between simple and alternate shapes.

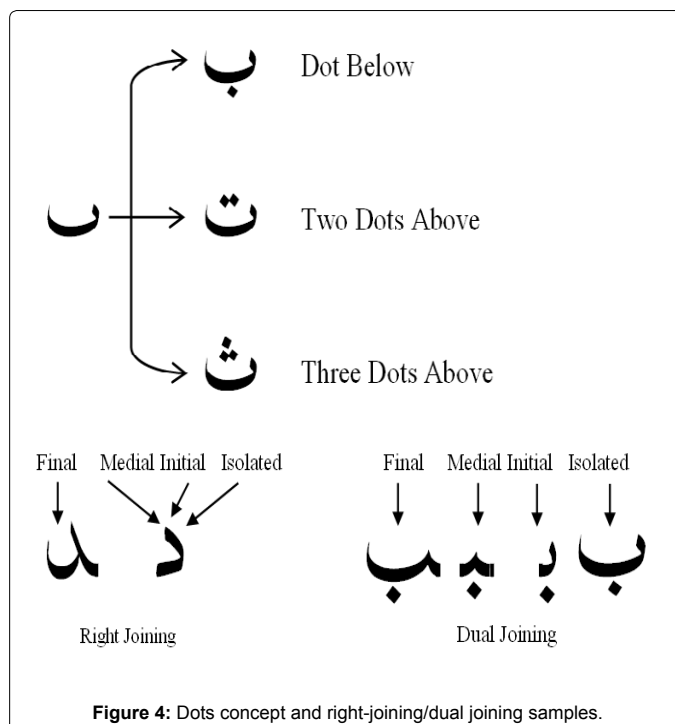


Figure 4: Dots concept and right-joining/dual joining samples.

medial, and final while right-joining characters have 2 glyphs: isolated (initial) and final (medial). Arabic alphabet has 28 character set, where seventeen characters have basic forms and eleven characters are variations by adding dots above or below to the basic characters. This procedure of adding dots is applied the basic characters to make new characters for extended Arabic such as Persian, Pashto, Urdu and Sindhi. Arabic script also uses diacritical marks (aarab) above or below the character. Figure 4 shows two categories of Arabic character set and dot concept as well (Figure 4).

Spacing is another issue in Arabic script as it is different in Latin because the script is cursively joined. Kashida feature is used to handle extra space. Kashida is a horizontal stroke which behaves like a single character and has a separate key on the keyboard. Spacing in Arabic is different from how we know it in Latin because the script is joined. As such, users cannot “space out” a word or a paragraph. Extra space is controlled by the kashida, a horizontal stroke that has a dedicated key on keyboard. This character has its specific placement rules according to the rules of calligraphy [9].

Arabic script is rich in ligatures, and ligatures are of two types: (1) mandatory (required ligatures), which are all times needed and (2) optional (standard) which exist only for aesthetic reasons (legibility or justification). Ligatures enhance font’s capabilities and take the typesetting towards more realistic view. One single word can be represented in different styles using ligatures. Figure 5 shows importance of ligatures in a Arabic script: (Figure 5).

There must be different shapes that a letter can take according to its neighboring letters. Opentype feature ‘calt’ (contextual alternatives) is used to add some calligraphic touch to a digital font. For example; in Figure 6, the initial form of Beh can take three alternate shapes according to its left neighboring letter: (Figure 6).

Kashida is a connection between Arabic letters and it is not a separate character but a stretch of the previous letter; used for various purposes: emphasis, legibility, aesthetic and justification. Kashida

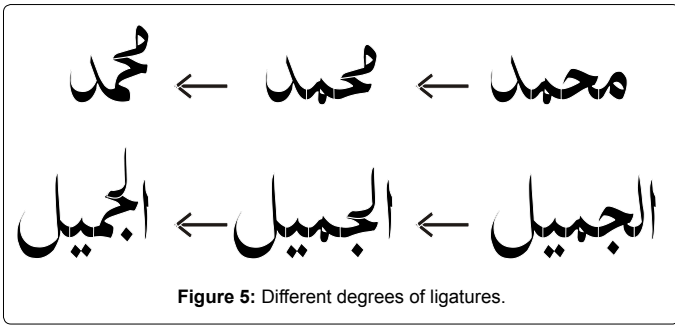


Figure 5: Different degrees of ligatures.

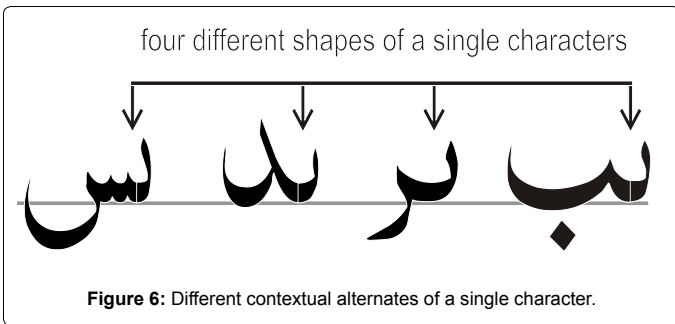


Figure 6: Different contextual alternates of a single character.

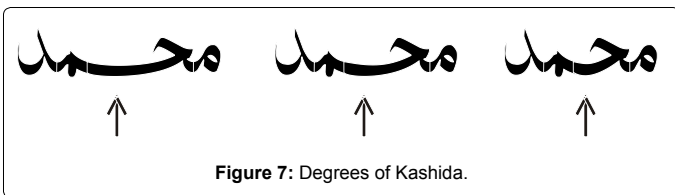


Figure 7: Degrees of Kashida.

comes in various degrees in Figure 7.

The developed font is a multi-font (Arabic, Pashto, Persian and Urdu) such that fully completed text in all aspects can be written in all these four languages with a single font. Figure 8 shows the overall functionalities for each of the language (Figure 8).

Overall functionalities of the multi-font has all features of Arabic scrip and its related languages. The dotted line shows that language other than Arabic uses Arabic letters such that it is an extended form of Arabic script. More specifically, for all the languages opentype features added in the font for its corresponding text [10].

Implementation

Pashto, Urdu and Persian use Arabic letters and some of the letters of Persian alphabets are used by Urdu language, so just these two languages (Pashto and Urdu) are discussed here [11-13].

Pashto alphabet is a modified form of the Persian alphabet, which is basically derived from Arabic alphabet. It has extra letters for Pashto-specific sounds. Pashto alphabet consists of 45 letters, and 4 diacritic marks. Figure 9 shows a sample of Pashto text is given in nasar form (Figure 9).

Urdu is written right-to-left in an extension of the Persian alphabet, which is itself an extension of the Arabic alphabet. Perso-Arabic script has been extended for Urdu with additional letters Š, š. Two letters À and ى have added dimensions in use, in order to make the language suitable for the people of South Asia (mainly Pakistan).

Developing and generating fonts for Arabic script languages is more complex than Latin script. Unicode presents four character code charts for the Arabic Script, but none of the four can be used separately. The four encoding charts are:

1. 'Arabic' contains only the isolated letterforms of the basic and extended Arabic.
2. 'Arabic Supplement' contains the extended Arabic letters needed for African languages that use the Arabic script.
3. 'Arabic Presentation Forms A' contains the extended Arabic letters and all the Arabic ligatures or mergers.
4. 'Arabic Presentation Forms B' contains all the basic Arabic character set in isolated, initial, medial and final forms.

An Arabic font that only needs to support Arabic language; 'Arabic' and 'Arabic Presentation Forms B' are needed. If a font needs

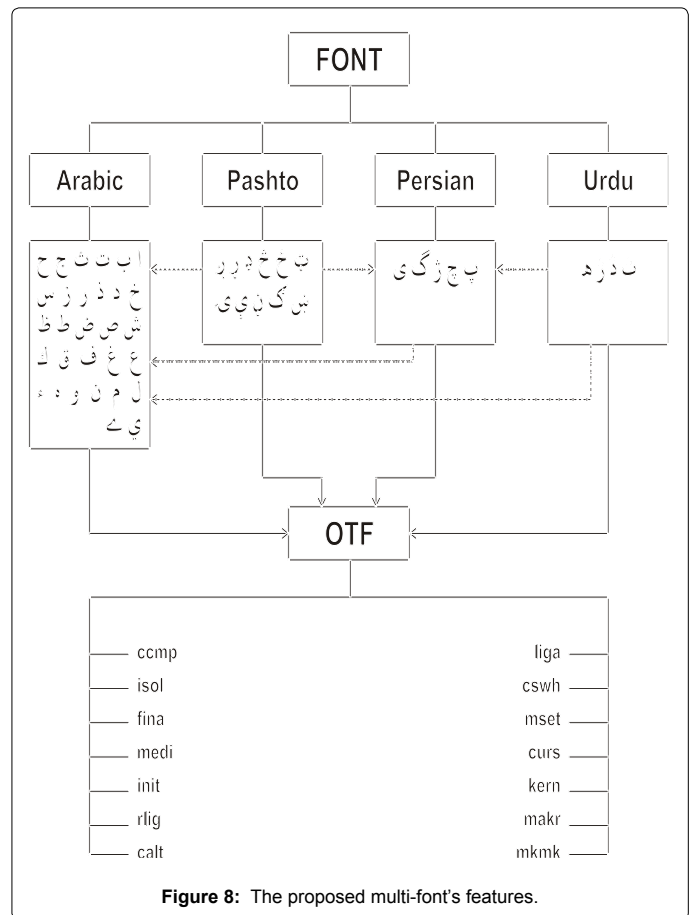


Figure 8: The proposed multi-font's features.

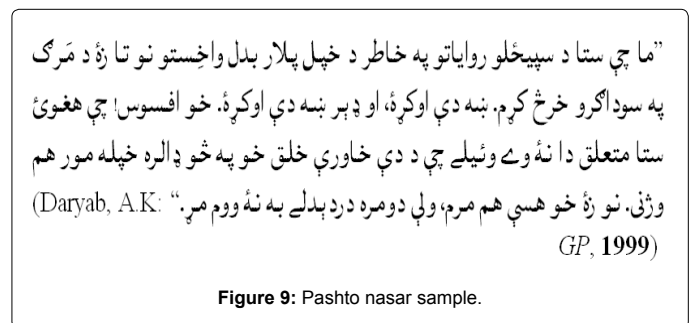


Figure 9: Pashto nasar sample.

to support other Asian languages that use Arabic script (such as Pashto, Persian and Urdu); ‘Arabic’, ‘Arabic Presentation Forms A’ and ‘Arabic Presentation Forms B’ are needed.

Developing OTF

To encode complex script features in fonts, choose character sets, organize font information, and use existing tools to produce Arabic fonts. Registered features of the Arabic script are defined and illustrated, encodings are listed, and templates are included for compiling Arabic layout tables for OpenType fonts.

In addition to being a primer and specification for the creation and support of Arabic fonts, this document is intended to more broadly illustrate the OpenType Layout architecture, feature schemes, and operating system support for shaping and positioning text [14].

Shaping engine: The Uniscribe Arabic shaping engine processes text in stages. The stages are: (a) Analyzing the characters for contextual shape, (b) Shaping (substituting) glyphs with OTLS (OpenType Library Services) and (c) Positioning glyphs with OTLS. Uniscribe next applies features concerned with positioning, calling functions of OTLS (opentype library services) to position glyphs [15,16].

Handling invalid combining marks: Combining marks and signs that appear in text not in conjunction with a valid consonant base are considered invalid. This section describes how to handle invalid combination of combining marks to reduce the number of mistakes during typing text. Uniscribe displays these marks using the fallback rendering mechanism e.g. positioned on a dotted circle.

An Arabic OTL font should contain a glyph for the dotted circle (U+25CC) for the fallback mechanism to work properly. If this glyph is missing from the font, a white box as an invalid sign will be displayed on the missing glyph shape.

If an invalid combination is found, like two fathas (ُ ُ) on the same base character (for example َ), the diacritic that causes the invalid state is placed on a dotted circle (َ) to indicate the invalid combination.

The shaping engine for non-OpenType fonts will cause invalid mark combinations to overstrike. This is the problem that inserting the dotted circle for the invalid base solves. It should also be noted that the dotted circle is not inserted into the application’s backing store. This is a run-time insertion into the glyph array that is returned from the Script Shape function.

The invalid diacritic logic for Arabic is based on the classes listed in next page. There is a check to make sure more than one mark of a class is not placed on the same base. It means that one mark of a class will not be placed over or below the other mark of the same class.

Additionally, DIAC1 and DIAC2 classes should not be applied on the same base character. To handle invalid combining marks, all the entries in table 6.3-1 must not be ignored. The table for handling invalid combining marks is shown in next page [12].

Features

The features listed below have been defined to create the basic forms for the languages that are supported on Arabic systems. Regardless of the model an application chooses for supporting layout of complex scripts, Uniscribe requires a fixed order for executing features within a run of text to consistently obtain the proper basic form. This is achieved by calling features one-by-one in the standard order listed below.

The order of the lookups within each feature is also very important.

Following the OpenType specification, lookup sequence defines in lookup array to control the order of a text processing client uses to apply lookup data to glyph substitution and positioning operations. The order of lookup within the feature tag is critical for desired processing.

Figure 10 shows a brief comparison of existing and modified font. This font has been taken form Liwal Pashto system (Pashto Kror), which is modified during this work along with other fonts. The font was designed for Pashto purpose only. In this work, features of Arabic, Persian and Urdu are also added. Ligatures, character composition/ decomposition, mark positioning, contextual alternates, cursive positioning, kerning and other opentype features are implemented. Circles around characters in Figure 10 show some of those features (Figure 10).

What’s Better?

After studying and using existing fonts and other systems for Arabic script and its related languages (such that Pashto, Persian and Urdu), it seems to be very clear that existing fonts have some critical issues about Arabic typography. Some other problems has been faced related to these languages. Some of the fonts have undefined extensions and while others are limited according to their functionalities. New fonts were developed for these languages but still some features have been ignored. My personal experience in graphics designing and typing mostly in Pashto, Urdu, Persian and Arabic was very beneficent for me to work with existing fonts, to draw its limitations and to suggest another system which solves those problems and proposed a complete font in all aspects. This work is not just about the font development (tricks and tactics) but it is about an idea, the idea which means “all in one”. The idea is that there must be such a system (fonts) which can completely solve issues for Arabic script and its related languages. The idea is not only to type text in Arabic script or in other language uses Arabic script; this is about how to work with a single font in all these four languages. More specifically, the proposed font should have

TEXT TYPED IN EXISTING FONT	TEXT TYPED IN MODIFIED FONT
<p>’ماچې ستا د سپېڅلو روایاتو په خاطر د خپل پلار بدل (اڅستو نو تا زه د مرگ په سوداگرو خرڅ کړم. ښه دې اوکړه، او ډېر ښه دې اوکړه خو افسوس! هغوی ستا متعلق دا نه وے وتیلے چې د دې خاورې خلق خو په څو ډالره مور هم (ژنی) نو زه خو هېڅی هم مرم، (ډلی) دومره دردېدلے به نه ووم هر’</p>	<p>’ماچې ستا د سپېڅلو روایاتو په خاطر د خپل پلار بدل (اڅستو نو تا زه د مرگ په سوداگرو خرڅ کړم. ښه دې اوکړه، او ډېر ښه دې اوکړه خو افسوس! هغوی ستا متعلق دا نه وے وتیلے چې د دې خاورې خلق خو په څو ډالره مور هم (ژنی) نو زه خو هېڅی هم مرم، (ډلی) دومره دردېدلے به نه ووم هر’</p>
<p>’میں نے جب (تمہاری) تابندہ روایات کی خاطر اپنے باپ کا بدلہ لیا تو تم نے مجھے موت کے سوداگرو کے ہاتھوں بیچ دیا، اچھا کیا، (دبت) اچھا کیا مگر افسوس! کہ وہ (تیرے) بارے میں ایسا نہ کہتے کہ اس (مٹی) کے لوگ تو چند ڈالر کی خاطر (اپنی) ماں کو بھی مار دیتے ہیں، تو مجھے تو مرنا ہی تھا، مگر تب یہ موت اتنی تکلیف دہ نہ ہوتی’</p>	<p>’میں نے جب (تمہاری) تابندہ روایات کی خاطر اپنے باپ کا بدلہ لیا تو تم نے مجھے موت کے سوداگرو کے ہاتھوں بیچ دیا، اچھا کیا، (دبت) اچھا کیا مگر افسوس! کہ وہ (تیرے) بارے میں ایسا نہ کہتے کہ اس (مٹی) کے لوگ تو چند ڈالر کی خاطر (اپنی) ماں کو بھی مار دیتے ہیں، تو مجھے تو مرنا ہی تھا، مگر تب یہ موت اتنی تکلیف دہ نہ ہوتی’</p>

Figure 10: Comparison of existing and modified font.

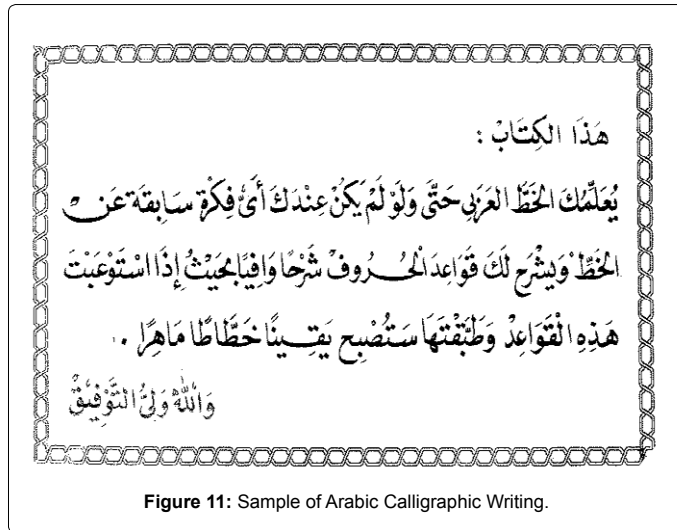


Figure 11: Sample of Arabic Calligraphic Writing.

all opentype features in order to use a computer system in an easy way to type calligraphic text in right to left languages [11].

Possible efforts have been taken to develop such a font that must be appreciated and user must take advantage of it. For each supported script, a set of rules is provided for detailing the default shaping behavior. Each font optionally provides alternate shaping behavior to be used for specific languages within different contexts.

Conclusion

Many aspects of the font making for Pashto, Urdu, Arabic, and Persian script have discussed in detail. Main focus of this work is to remove previous problems and to develop such fonts that are all in one.

A number of problems were faced during this work but at the end all the problems are successfully faced and all the goals are achieved in a proper way. A possible calligraphic touch has been given to all fonts developed during this work.

Some of the problems were mainly in the connections of characters with ligatures. In first glance thinking about all possible ligatures and then coding those ligatures was a tough task. The Ayah symbols and its proper positioning in font setting was another difficult task. Its coding was made for so many times and each time coding has been started and progressed to the goal. Similarly, Ramooz-o-Awqaf setting was also problematic task, but at the end, the goal of the project has been achieved.

Future Work

Stretching is a typographic effect used with Arabic writing systems to elongate characters. Stretching is preferred for inserting extra inter-word spacing to achieve left (end-of-the-line) justification. Figure 11 shows an example of Arabic calligraphy. This stretching is called “kashida” in Arabic, Persian, Pashto and Urdu as well, which means “to stretch”. In Arabic script, there are two kinds of stretching.

1. Stretching of the connection to the last connecting-before letter in a line or words.

2. Stretching of the last stretchable letter in a line or words (Figure 11).

In Figure 11, the pronounced stretching at the (left) end of line 2 is of a letter, the connecting-before nun, while those near the middles of lines 3 and 4 are of connections [1].

Stretching feature is not added to the developed fonts due to limited time but the idea indicates to future researchers that what should be done afterwards in the field of typeface designing.

References

1. Alan P (1997) Production for Graphic Designers. (2ndedn), Prentice Hall Inc.
2. Richard E (1994) Glossary of Typesetting Terms. Chicago guides to writing, University of Chicago Press.
3. Craig J, Scala IK (2006) Designing with Type, the Essential Guide to Typography. (5thedn), Watson-Guption Publications, China.
4. Graham L (2002) Basics of Design: Layout and Typography for Beginners. (2ndedn), New York.
5. Pressman RS (2001) Software Engineering: A Practioner's Approach. (6thedn), McGraw. Hill Publications, USA.
6. Kristin C (2005) Layout Workbook: A Real-World Guide to Building Pages in Graphic Design, Rockport Publishers, Quayside Publishing Group, China.
7. Wesley A (2000) The Unicode Standard Version 3.0, The Unicode Consortium. (1stedn), Addison Wesley Longman, Inc, USA.
8. Levine YM, Kay DC, Wagner R (2004) WordPerfect 12 for dummies For Dummies. Willey Publishing, Inc, Indiana.
9. Poole A (2003) Issues of Saliency and Recognition in the Search for Web Page Bookmarks. Master Thesis, Lancaster University.
10. Alikhel D (1999) Gul Pana Hamesha Pashto Adabi Tolana (Regd) Amandara.
11. http://en.wikipedia.org/wiki/Arabic_alphabet
12. <http://www.microsoft.com/typography/otfntdev/arabicot/features.htm>
13. <http://www.unicode.org/unibook/index.html>
14. <http://www.yourdictionary.com/grammar-rules/phonetics.html>
15. http://whatis.techtarget.com/definition/0,,sid9_gci212200,00.html
16. <http://www.microsoft.com/typography/OpenType%20Dev/arabic/intro.msp>