Development of an Efficient Computing Multilingualism Model for Diacritical Marks in Arabic and Hindi

Abu Sarwar Zamani *sarwar_zamani@yahoo.com*
Lecturer/College of Science
Shaqra University
Riyadh, Kingdom of Saudi Arabia

Dr. Nassir Al Arifi *nalarifi@ksu.edu.sa*
Professor, Dean/College of Science
Shaqra University
Riyadh, Kingdom of Saudi Arabia

Md. Mobin Akhtar *jmi.mobin@gmail.com*
Lecturer/College of Science
Shaqra University
Riyadh, Kingdom of Saudi Arabia

Abstract

Language competence is a cognitive property of the individual speaker. There is a wide gap between commonly voiced representations of language, person, and place and actual practices of language use, identity assertion, and spatial occupation. It is noted that the one can focus on resolving related outstanding standardization issues in support of localization and multilingual requirements. This paper investigates the Diacritical marks and various typographic rules in Hindi and Arabic which are complex in multilingual documents. A computing Multilingualism model is developed which proposed a solution to the problem of position of Diacritical Marks in Multilingual documents. The developed model is found to be an efficient tool for solving the problem of positioning diacritical marks for multilingual fonts in True Type as well as Open Type format.

Keywords: Multilingualism, Diacritical Marks, Computing, Typographic.

1. INTRODUCTION

Language is a defining feature of human civilization and many languages and scripts have come into existence that is used by people around the world. Language plays unique role in capturing the breadth of human diversity. We are constantly amazed by the variety of human thought, culture, society and literature expressed in many thousands of languages around the world.

Literature is one of the vital components for the development of the society. With more than half of the worlds literature being published in languages other than English. Massive volumes of text in many languages are becoming available online. The documents may be created initially in digital form or could be converted from other media.

On the other hand rapid diffusion over the international computer networks of the world wide distributed document bases, the question of multilingual access and multilingual information retrieval is becoming increasingly important.

In a multilingual digital document, the principles of designing are risky by the likely conflict rules and mechanism that control each of the writing. Diacritics are an example.

Diacritical marks is a small mark added to a letter that changes its pronunciation such as (أ) hamza indicates upper of Alif, (إ) hamza indicates lower of Alif. Diacritics are often placed above the letter but they can be placed below, in or through, before or after or around a glyph.

Diacritical marks have common roles between the different languages of the world like:-

- Define playback
- Amend the phonetic value of a letter.
- Avoid ambiguity between two homographs.
- Etc.

However, Hindi is an Indo-European language spoken mainly in north, central, and western Indian. Hindi also refers to standardized register of Hindustani that was made one of the official languages of India. Hindi is written in Devnagri and has been partially purged of its Persian and Arabic vocabulary, which was replaced by words from Sanskrit.

This study focuses on to appropriate a resolution to the problem of positioning of diacritics:

We have taken some steps:

- We compare problem design of diacritical marks in the Hindi script with the design of diacritical for Arabic script.
- We spend the last part to problem of positioning diacritical marks.
- We identified strategies to solve this problem and examine their ability in the Hindi case.

2. GENERAL INFORMATION

A. History About Diacritic Sign

The first diacritics appeared in Ancient Greece and Rome, evolved and spread in subsequent European languages. While they were created to help in the pronunciation of letters and words.
Arabic is in the Semitic language group, which seems to have originated somewhere near modern Syria, Hebrew and to have spread from there through Lebanon, Israel, and Jordan down to the Arabian Peninsula. It is also cursive and written from right to left. The majority believes it has developed down writing Nebatean. Others believe it comes from Al-Musnad also known as Al Hamiri (writing of the former yemini). A small group believes that writing is a pure divine production. Until the time of Mohammed, in the 600's AD, Arabic was mainly spoken and not written. Still, there are some written records from the Arabian Peninsula from before the 600's AD. These are called Sabataean. But they are only short inscriptions in stone, not really literature. After the Islamic conquests of the late 600's AD, people soon began to speak Arabic all over the Islamic Empire, from Afghanistan to Spain, and people speak Arabic in even more places today (though not in Spain). By 1000 AD, people spoke Arabic in India. Many people began to write in Arabic. Among the first things to be written was the Quran, because the Quran played a key role in the development of Arabic script. But soon many scientific texts and medical books and math books were written in Arabic, and also stories like the Arabian Nights or the story of Aladdin. There were many Arab historians, geographers, philosophers, and poets. However, the most common solution is to add diacritical marks on the letters, often imitating the spellings of other languages [2].

Hindi is the third most widely-spoken language in the world (after English and Mandarin): an estimated 500-600 million people speak the language. A direct descendant of Sanskrit through Prakrit and Apabhramsha, Hindi belongs to the Indo-Aryan group of languages, a subset of the Indo-European family. It has been influenced and enriched by Persian, Turkish, Farsi, Arabic, Portuguese, and English. Hindi inherited its writing system from Sanskrit. Hindi can be traced back to as early as the seventh or eighth century. The dialect that has been chosen as the official language is Khariboli in the Devnagari script. Other dialects of Hindi are Brajbhasa, Bundeli, Awadhi, Marwari, Maithili and Bhojpuri.The general appearance of the Devanagari script is that of letters 'hanging from a line'. This 'line', also found in many other South Asian scripts, is actually a part of most of the letters and is drawn as the writing proceeds. The script has no capital letters.It was in the 10th century that authentic Hindi poetry took its form and since then it has been constantly modified. History of Hindi literature as a whole can be divided into four stages:

- **Adikal** (the Early Period),
- **Bhaktikal** (the Devotional Period),

- **Ritikal** (the Scholastic Period) and
- **Adhunikkal** (the Modern Period).

Adikal - The Early Period: Adikal starts from the middle of the 10th century to the beginning of the 14th century.

Bhakti Kal or the Devotional Period: Bhakti Kal or the Devotional Period stretched between the 14th and the 17th century. During this age Islamic customs were heaped upon the common people, and the Hindus were quite dejected at the effect on their culture.

Ritikal or The Scholastic Period: The poets of Ritikal or the Scholastic period can be classified into two groups on the basis of their subject: Ritibaddha (those wedded to rhetorics) and Ritimukta (free from rhetorical conventions).

Modern Hindi Literature: Modern Hindi literature has been divided into four phases; the age of Bharatendu or the Renaissance (1868-1893), Dwivedi Yug (1893-1918), Chhayavada Yug (1918-1937) and the Contemporary Period (1937 onwards).
Bharatendu Harishchandra (1849-1882) brought in a modern outlook in Hindi literature and is thus called the 'Father of Modern Hindi Literature'. Mahavir Prasad Dwivedi later took up this vision. Dwivedi was a reformist by nature and he brought in a refined style of writing in Hindi poetry, which later acquired a deeper moral tone.

Classification in Hindi
There are some kinds of Hindi Diacritic Marks which are below:

- Diacritics Above

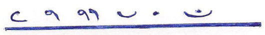

FIGURE 1: Diacritics above in Hindi.

- Diacritics Below

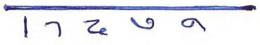

FIGURE 2: Diacritics below in Hindi.

- Diacritics Through

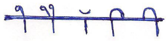

FIGURE 3: Diacritics aesthetics in Hindi.

- Esthetics Diacritics

□□□□□ □□□□□□ , जय हिंद

FIGURE4: Diacritics aesthetics in Hindi.

- Explanatory Diacritics

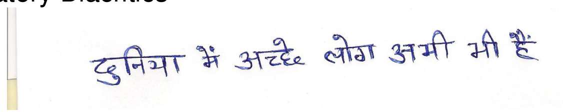
दुनिया में अच्छे लोग अभी भी हैं

FIGURE5: Diacritics Expalantory in Hindi.

C. Classification in Arabic

Arabic Diacritics can be classified into three categories [1] :

- Language's diacritics: composed on:
  - Diacritics above
    it's a mark placed above a letter, as Fatha, Damma or Sukun.

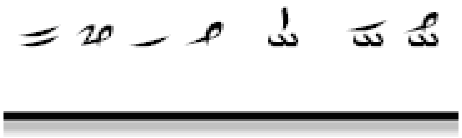

FIGURE 6: Arabic diacritics above.

  - Diacritics below

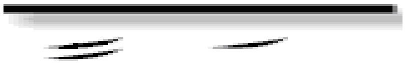

FIGURE7: Arabic diacritics below.

  - Diacritics Through

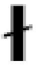

FIGURE:8: Jarrat wasl through Alef.

  - Esthetics Diacritics

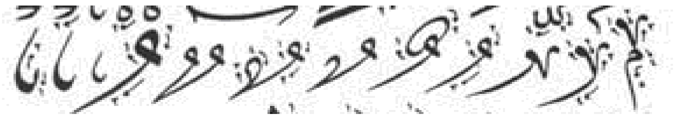

FIGURE9: Kasra and Kasrattan.

  - Explanatory Diacritics

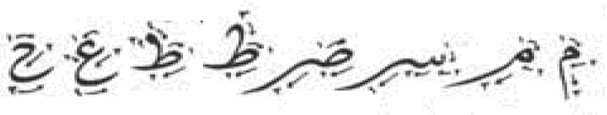

FIGURE10: Explanatory diacritics [10].

3. DIACRITICAL MARKS WITH UNICODE

Unicode provides a unique number for every character, no matter what the platform, no matter what the program, no matter what the language. The Unicode character set has the capacity to support over one million characters, and is being developed with an aim to have a single character set that supports all characters from all scripts, as well as many symbols, that are in common use around the world today or in the past. The Unicode character encoding treats alphabetic characters, ideographic characters and symbols in an equivalent manner, with the result that they can coexist in any order with equal ease. Unicode assigns to each of its character a unique numeric value and name.

There were four key original design goals for Unicode:

(i) To create a universal standard that covered all writing systems.

(ii) To use an efficient encoding that avoided mechanisms such as code page switching, shift- sequences and special states.

(iii) To use a uniform encoding width in which each character was encoded as a 16-bit value.
(iv) To create an unambiguous encoding in which any given 16-bit value always represented the same character regardless of where it occurred in the data.

However, Unicode provides other information crucial to ensure that the encoded text will be readable: the case of coded characters, their properties and their directionality letter. Unicode also defines semantic information and includes correspondence tables of breakage or conversions between Unicode and directories of other important character sets.

Bi-directional text is text containing text in both text directionalities, both right-to-left (RTL) and left-to-right (LTR). The bidirectional algorithm takes place in six steps:

- Determine the default direction of the paragraph;
- Process the Unicode characters that explicitly mark direction;
- Process numbers and the surrounding characters;
- Process neutral characters (spaces, quotation marks, etc.);
- Make use of the inherent directionality of characters;
- Reverse substrings as necessary.

4. DESIGINING, POSITIONING & MULTILINGULISM

In the time of lead type, the creation and design of letters was solely dependent on type foundries. These foundries, such as Monotype and Linotype, had all the copyrights of these fonts. Lots of concept underlies the field of design, as the balance, the rhythm, etc. The principles of design face in the case of mixture of different directions postings to change the rules of writing. It is in a somewhat similar situation when a multitude of styles in a monolingual Arabic text where the change of style indicates a title or section begins [1].

A. Language Variety Space

There are many varieties of language, text, discourse. These varieties are conditioned mainly by the functions of language in actual use. Positions in variety space define a language variety with specific forms, and specific conventions of meaning and use. Consequently, understanding and intelligent design of text and discourse, from conversation, through letters to hypertext, requires models of language functions as touchstones of quality. In Arabic, heights [1] and forms of letters vary depending on the context:

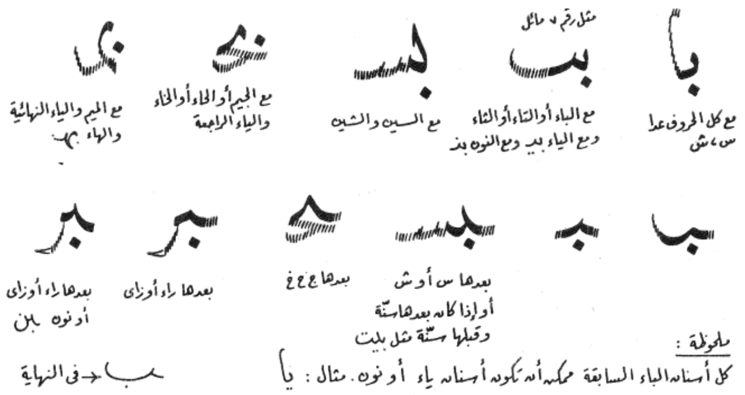

FIGURE 11: Arabic letter Beh

The spatial properties vary between Hindi and Arabic scripts. Arabic scripts start from right to left, which vary slightly depending on whether they are connected to another letter before or after them. The definition of “bold” depends, in Arabic, of style. The reduction in the density of letters is by layering or by reducing the body. Diacritics in the Thulut style, unlike the Naskh, by a Qalam, pen, different from that used for the body of letters base. The harmonization of multilingual document is therefore influenced by the multitude of scripts or styles in the same language [5].

B. **Justification of the Hindi Text**

The justification of the Hindi text makes itself while varying the space between the words and the characters, so that the line of text filled the inter-margin space. The value of the spacing varies between a minimal value and another maximal when the optimal value doesn't permit the justification of the text. The hyphenation permits to cut the word that arrives at the end of line in order to have a better visual within a text. The general appearance of the Devanagari (Hindi) script is that of letters 'hanging from a line'. This 'line', also found in many other South Asian scripts, is actually a part of most of the letters and is drawn as the writing proceeds. The script has no capital letters. Amongst its interesting features is a three-tier level of honorifics, allowing great subtlety in adjusting the level of communication to suit 'formal', 'familiar' and 'intimate' conversational contexts. Thus, the polite communicating of gratitude, etc, is an intrinsic part of the language itself and does not rely solely on separate words for 'please' and 'thank you'.

Problems related to the justification and literature of the text, especially a justification as well as literature of the kind made by processing software word processing, without correction by a human operator are potentially many. Basically there are two types of literature are used in Hindi Sculpture.

(1) Swars (□□□□ / Vowels)

DevaNagari vowels are not scattered in the 'Varnamala' (DevaNagari alphabet) but are arranged at the beginning of the alphabet.

अ आ इ ई उ ऊ ए ऐ ओ औ ऍ ऑ ऋ

These vowels were arranged according to a scheme [7]. This scheme is not completely scientific (phonetic), but definitely helpful in memorizing & reciting these vowels.

(2) Vyanjans (□□□□□□ / Consonants)

These are very logically arranged in following groups.

(i) **Sparsh:** Sparsh means touch. While speaking, along with vibrations in vocal cord and passage of air from mouth and nose, tongue and lips move. Particularly for pronouncing consonants the movements of tongue and lips are important [7]. In DevaNagari, most of the consonants are arranged logically; depending upon the position of tongue (what it touches) and movements of lips. Like…

क		म
ka	to	ma

(ii) **Antashth:** This is the middle set ‘Antahsth’ in Sanskrit means ‘middle’ or ‘inner’. Theses are…

य	र	ल	व
ya	ra	la	va

(iii) **Ushm:** 'UShm' means hot! Isn't it amazing to know that terminologies developed separately? resulted in related terms- ‘Friction’ and ‘Hot’(heat)!!! These are…

श	ष	स	ह
sha	sha‡	sa	ha

C. **Justification of the Arabic Text**

In the Arabic writing, that is cursive, a word can be dilated by the kashida - specific to the Arabic writing - to cover much space [1] [8] and can be pressed by the use of the ligatures [1] [8]. It has other mechanisms of management of the Arabic line: graphic fillers (as the three points), reduction of the size of the characters, elongation of the letters, superposition of the letters, writing in the margin, etc. [1] [8]. These mechanisms influence on the measurements and the positioning of the Arabic diacritical marks [2].

5. DIACRITICS DESIGN

There are two problems in the design of Hindi Diacritics.

- They must concord with the Glyph.
- Do not cause problems with other basic glyphs;

In the Arabic case, there are aesthetic diacritics whose position depends on other diacritical marks. The interactive diacritics relationship with the mechanisms of justification requires resizing and repositioning diacritical word influenced by the effects of justification.

A. Asymmetry Problem

The balance is the stability resulting from the review of an image and a comparison with our ideas of the physical structure (such as mass, gravity, or the edges of a page). That is the arrangement of objects in a design specified according to their weight in the visual picture composition. The balance generally exists in two forms: symmetrical and asymmetrical.

The symmetrical balance occurs when the weight of a graphic composition is evenly distributed around a central axis vertical or horizontal. The symmetrical balance is also known as formal balance. The asymmetrical balance occurs when the weight of the graphic composition is not spread evenly around a central axis. The asymmetrical balance is also known as informal balance. The size of a Hindi diacritic and weight must be balanced with the glyph base with which it is used [9]. The horizontal alignment of diacritical glyph with the foundation should be such that there is balance the two views. For diacritic center symmetry with glyphs basic symmetrical, simply align the center of the bounding box of diacritic with the basic glyph [9]. If either one is asymmetrical other measures must be used. Follow, we present the main issues of design diacritics as they have been cited in [9].

1) Case of symmetrical basic Glyph

A glyph is a graphical symbol that represents a model component, such as an individual molecule. In some cases an attribute of the glyph is a function of the model component that it represents.
One solution is to align the optical center of the letter with the mathematical center of space. The optical center is estimated by the center of the contour.

2) Case of Asymmetrical base Glyph

In this case, the diacritic exchange up connection following the basic glyph. The optical alignment is not always used and other solutions are offered by new technologies such as OpenType and Graphite.

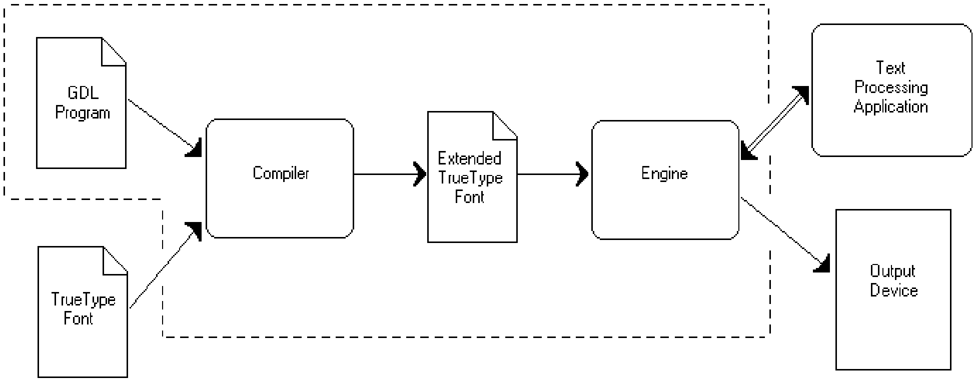

FIGURE 12: Graphite System Architecture.

Description: This system can be used to create “smart fonts” capable of displaying writing systems with various complex behaviors, such as:

- A rule-based programming language Graphite Description Language (GDL) that can be used to describe the behavior of a writing system
- A compiler for that language
- A rendering engine that can serve as the back end of a text processing application.

Graphite renders TrueType fonts that have been extended by means of compiling a GDL program [2].

B. Multiple Diacritics

Diacritics could cause multiple problems with the baseline or with other glyphs. Different techniques are used to solving this problem including: draw a glyph gathering all the diacritics multiple, etc.

C. Particular Issues to Arabic

Arabic diacritics role is to fill the void, white space, in the word that there are specific diacritical marks, for aesthetics. There are three mechanisms for creating void in the Arabic word: kashida, extension glyphs and the interconnection between glyphs. In each case, the void is filled in two steps:

- The first, by resizing the Fatha in proportionality with the white;
- The second, by placing the aesthetics’ and explanatory diacritics.

Diacritical marks lead, according to the language’s function, to repeat the characteristics common to many of the glyphs.
The concept of symmetry in Arabic design is related to the line writing where the extensions are to balance the masses of other glyphs.

Arabic diacritics have a relationship with the mechanisms of justification. The diacritical marks are cosmetic compared to other signs respecting fill the void and not obscure the gray.

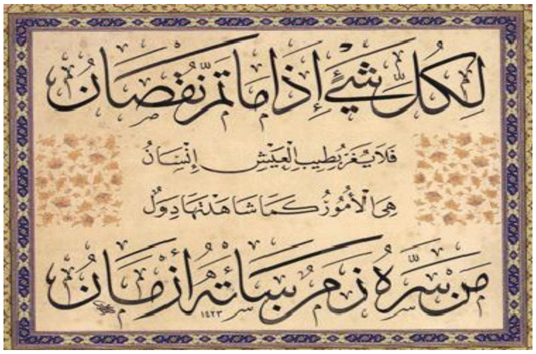

FIGURE 13: Arabic Diacritics Role.

6. NEW TECHNOLOGIES & DIACRITICAL POSITIONING

We are studying the three font's formats: TrueType, OpenType and Graphite.

1) True Type: TrueType fonts offer the highest possible quality on computer screens and printers, and include a range of features which make them easy to use. TrueType is an outline font standard originally developed by Apple Computer in the late 1980s as a competitor to Adobe's Type 1 fonts used in PostScript. The primary strength of TrueType was originally that it offered font developers a high degree of control over precisely how their fonts are displayed, right down to particular pixels, at various font heights [4].
2) Open Type: The Open Type font format is an extension of the TrueType font format, adding support for PostScript font data.
   Open Type fonts and the operating system services which support Open Type fonts provide users with a simple way to install and use fonts, whether the fonts contain TrueType outlines or CFF (PostScript) outlines [12].
   The Open Type font format addresses the following goals:

- broader multi-platform support
- better support for international character sets
- better protection for font data
- smaller file sizes to make font distribution more efficient
- Imported internet and PDF (Portable Document Format) publishing.

GPOS table manages the positioning of glyphs. We can put any diacritic on any glyph basic threw it [4]. Each diacritic has a base. Diacritics are divided into several classes according to their behavior. Each basic glyph as attachment points that diacritic class.

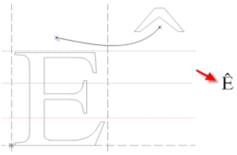

FIGURE 14:Diacritics Position.

A. Attachment and Cluster in Graphite

The positioning of glyphs is done by two simple operations: moving and kerning, a simple tool: the points of attachment. If two glyphs "A" and "B" are attached, one-by-example "B" is attached to "A" and "A" is said base of "B". Another glyph "C" in turn can be attached to either "A" or "B", etc. [12].

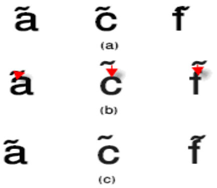

FIGURE 15: Diacritics attachment points.

The FIGURE 15 demonstrates the usefulness of attachment points. As shown in FIGURE 15 (a), a record of diacritics with a "not smart fonts" seems correct when they are attached to a tiny symmetrical centered as "a", but if not symmetric the diacritic is not centered correctly and comes into collision with the upper half of the glyph, or both. For Graphite font, stain is different: FIGURE 15 (b) shows the commitment indicated by small dots and arrows, and FIGURE 15 (c) shows the results with the correct record. The mechanism of base resolves the multiple diacritics problem, when the first diacritic is attached to a glyph base; it in turn is the basis of the following diacritic.

The basic glyph and diacritic form a cluster. Graphite includes the ability to calculate metrics cluster or sub-cluster glyph individual for use in operations positioning [12].

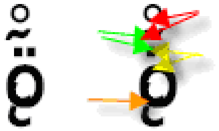

FIGURE 16: Multiple diacritics attachment points.

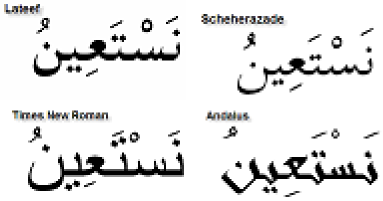

FIGURE 17: Examples of Arabic fonts.

B. Diacritics Positioning System

To place one or more diacritical marks relative to the base glyph, this system use a diacritic’s bounding box and the base glyph's bounding box, in association with diacritic place data stored in the system[11]. The position data enables the diacritic positioning system to call associated functions that place multiple diacritics above and/or below a single base character without interfering with one another, e.g. to stack the diacritics. In addition, the information about the diacritic characters can be employed to prevent interference between a diacritic and the base character in special circumstances [11].

a. The Algorithm of Diacritic Position system

```
Start

Step – 1:        Glyph Received
Step – 2:     If  Diacritical Then Step – 3 Else Step - 7
Step – 3:     If Special base Then Step – 4 Else Step - 5
Step – 4:     Retrieve diacritical mark and GOTO Step - 6
Step – 5:     Retrieve base char mark orientation
Step – 6:     Call H function
Step – 7:     Call V function
Step – 8:     Draw glyph
Step – 9:     Continue………..
End
```

b. Description

When the system receives the information that the mark is to be placed over the base character, he looks up the orientation for this mark in the table that is stored in memory. This table [11] lists each diacritic by its name or their Unicode value. Based on this information in this step, the system calls a pair of functions H and V for properly positioning mark.

c. Commentary

Graphite and OpenType font formats have the advanced features to treat Arabic script. For this reason, we limit this study to the system for positioning diacritical mark in TrueType font format. In the Arabic script, the position and dimension of diacritical mark Fatha and Fathattan are related to form of base glyph and followed base glyph. So, to extend a system which operates under the same architecture as the diacritics positioning system three things to take into account:

- The functions H and V must have the ability to calculate the horizontal and vertical position of diacritic glyph relative to the base glyph and followed base glyph.

- The system must be able to substitute the diacritical mark if an extension takes place.

7. CONCLUSION

Most of the fonts used to write Arabic do not have a deep tables and technologies of different formats, but we believe that the resolution of problems of diacritical in the multilingual digital document affects a layout engines. These problems have link with the problems of design of Arabic basic letters as the superposition of letters, the reduction of body and ligatures.

8. REFFERENCES

[1] Vlad Atansiu, "Le phénomène calligraphique à l'époque du sultanat mamluk", PhD Thesis, Paris, 2003.

[2] http://a1.esa-angers.educagri.fr/informa/, February 2009.

[3] Mohamed Hssini, Azzeddine Lazrek and Mohamed Jamal Benatia, "*Diacritical signs in Arabic e-document*", CSPA'08, The 4th International Conference on Computer Science Practice in Arabic, Doha, Qatar, April 1-4, 2008 (in Arabic).

[4] R. Nicole, "Graphite Application Programmer's Guide", http://www.sil.org/.

[5] Mohamed Hssini and Azzeddine Lazrek," Design and Computer Multilingualism: case of Diacritical Marks, Department of Computer Science, Faculty of Sciences, University Cadi Ayyad - Marrakech, Morocco.

[6] Yannis Haralambus, "Fontes et codage", O'Reilly, Paris, 2004.

[7] J. C. Wells, "Orthographic diacritics and multilingual computing", Language problems & language planning ISSN, 2000, vol. 24, n° 3, pp. 249-272.

[8] Mohamed Jamal Eddine Benatia, Mohamed Elyaakoubi, Azzeddine Lazrek, "*Arabic text justification"*, TUGboat, Volume 27, Number 2, pp. 137-146, 2006.

[9] J. Victor Gaultney, "Problems of diacritic design for Latin script text faces", http://www.sil.org/, December 2008.

[10] H. Albaghdadi, "Korassat alkhat", Dar Alqalam, Beirut, 1980.

[11] Chapman, Christopher J., "*D*iacritic positioning system for digital typography*"*, http://www.freepatentsonline.com/WO2008018977.html, January 2009.

[12] http://www.typographie.org/, January 2009.