Pattern Recognition 47 (2014) 1229–1248

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

The optical character recognition of Urdu-like cursive scripts

Saeeda Naz [a], Khizar Hayat [a,d], Muhammad Imran Razzak [b], Muhammad Waqas Anwar [a], Sajjad A. Madani [a], Samee U. Khan [c,*]

[a] COMSATS Institute of Information Technology, Abbottabad, Pakistan
[b] King Saud bin Abdulaziz University for Health Sciences, Riyadh, Saudi Arabia
[c] North Dakota State University, Fargo, ND 58108-6050, USA
[d] University of Nizwa, Sultanate of Oman

ARTICLE INFO

Article history:
Received 1 June 2013
Received in revised form
29 August 2013
Accepted 30 September 2013
Available online 11 October 2013

Keywords:
Optical character recognition
Ligature
Character

ABSTRACT

We survey the optical character recognition (OCR) literature with reference to the Urdu-like cursive scripts. In particular, the Urdu, Pushto, and Sindhi languages are discussed, with the emphasis being on the *Nasta'liq* and *Naskh* scripts. Before detaining the OCR works, the peculiarities of the Urdu-like scripts are outlined, which are followed by the presentation of the available text image databases. For the sake of clarity, the various attempts are grouped into three parts, namely: (a) printed, (b) handwritten, and (c) online character recognition. Within each part, the works are analyzed par rapport a typical OCR pipeline with an emphasis on the preprocessing, segmentation, feature extraction, classification, and recognition.

1. Introduction

Urdu is the national language of Pakistan [1] having a population of around 180 millions. Moreover, there are an estimated 70 millions native Urdu speakers in India. Furthermore, the importance of Urdu further increases due to the lion share of India and Pakistan in the expatriate population of Middle East, USA, and Europe. In the spoken form, Urdu and Hindi (together called Hindustani, the third main language of the world) are almost identical. However, in the written form, both languages are diametrically opposites. Whilst the former follows Arabic-like script, the latter is in the *Devanagari* script. The popularity of Urdu is mainly due to its rich classical (prose and poetic) literature. In the recent years, there has been an unending demand for Urdu-based Optical Character Recognition (OCR), not only to facilitate the native speakers to readily use it for their mobile or tablet requirements, but also for the digitization of a large amount of legacy documents, such as holy books, magazines, newspapers, poetry books, and handwritten documents.

Although a computation intensive field, OCR has witnessed a significant improvement over the years. This is mainly due to the tremendous advances in the computational intelligence algorithms. The objective of character recognition is to imitate the human reading ability, with the human accuracy but at far higher speed. The target performance is at least five characters per second with a 99.9% recognition rate [2]. The OCR is a basic tool for various applications, such as document automation, cheque verification, data entry applications, advance scanning, reading machine for visually handicapped, and a large variety of many other banking and business applications. The OCR is an active area of research and its importance is well established par rapport the disciplines of digital image processing, pattern recognition, artificial intelligence, database systems, natural language processing, human-machine interaction, and communications. These applications can perform well, if the characters from text images are classified and recognized accurately.

Most of the commercial OCR applications are concerned with the machine printed Latin scripts having well-separated characters. Moreover, the OCR systems for printed Japanese and Chinese languages are also quite mature. The cursive script is also a very popular script and many languages, such as Arabic, Persian, Urdu, and Pushto are based on it. There are many font styles of cursive script, such as Nasta'liq, Kofi, Thuluth, Diwani, Riq'a, and Naskh to name a few. Among the aforementioned, Naskh and Nasta'liq are the most important to mention wherein the former is preferred for Arabic, Persian, and Pushto languages and the latter is adopted for Urdu typesetting (the comparative complexities of Naskh and Nasta'liq will be discussed in Section 2.2). Some commercial OCRs are available for printed Arabic characters but they have many technical problems, especially in the segmentation stage where the results are not enviable. For all practical purposes, the Urdu script is the superset of its Arabic and Persian counterparts.

* Corresponding author. Tel.: +1 701 231 7615; fax: +1 701 231 8644.
E-mail addresses: saeedanaz@ciit.net.pk (S. Naz), khizarhayat@ciit.net.pk, khizar.hayat@unizwa.edu.om (K. Hayat), imranrazak@hotmail.com (M. Imran Razzak), waqas@ciit.net.pk (M. Waqas Anwar), madani@ciit.net.pk (S.A. Madani), samee.khan@ndsu.edu (S.U. Khan).

0031-3203/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2013.09.037

	Arabic		Urdu		Pushto		Sindhi
1	ا [ʔ]	1	ا [a]	1	ا [a]	1	ا [a]
2	ب [b]	2	ب [b]	2	ب [b]	2	ب [b]
						3	ٻ [ɓ]
						4	ٻھ [ɓʰ]
		3	پ [p]	3	پ [p]	5	پ [p] (it is after ٹ)
3	ت [t]	4	ت [t]	4	ت [t]	6	ت [t]
						7	ٽ [ʈʰ]
		5	ٹ [ʈ]	5	ټ [ʈ]	8	ٹ [ʈ]
						9	ٺ [ʈʰ]
4	ث [θ]	6	ث [s]	6	ث [s]	10	ث [s]
5	ج [ʤ]	7	ج [ʤ]	7	ج [ʤ]	11	ج [ʤ]
				8	څ [dz]		
						12	ڇ [ʤʰ]
						13	ڃ [ɲ]
						14	ڄ [ɲʰ]
		8	چ [ʧ]	9	چ [ʧ]	15	چ [ʧ]
				10	څ [ts]		
						16	ڇ [ʧʰ]
6	ح [ħ]	9	ح [h]	11	ح [h]	17	ح [h]
7	خ [x]	10	خ [x]	12	خ [x]	18	خ [x]
8	د [d]	11	د [d]	13	د [d]	19	د [d]
						20	ڈ [ɗʰ]

	Arabic		Urdu		Pushto		Sindhi
						21	ڈ [ɗ]
		12	ڈ [ɖ]	14	ډ [ɖ]	22	ڈ [ɖ]
						23	ڊ [ɖʰ]
9	ذ [ð]	13	ذ [z]	15	ذ [z]	24	ذ [z]
10	ر [r]	14	ر [r]	16	ر [r]	25	ر [r]
		15	ڑ [ɽ]	17	ړ [ɽ]	26	ڑ [ɽ]
						27	ڑھ [ɽʰ]
11	ز [z]	16	ز [z]	18	ز [z]	28	ز [z]
		17	ژ [ʒ]	19	ژ [ʒ]		
				20	ږ [ʐ]		
12	س [s]	18	س [s]	21	س [s]	29	س [s]
13	ش [ʃ]	19	ش [ʃ]	22	ش [ʃ]	30	ش [ʃ]
				23	ښ [ʂ]		
14	ص [sˤ]	20	ص [s]	24	ص [s]	31	ص [s]
15	ض [dˤ]	21	ض [z]	25	ض [z]	32	ض [z]
16	ط [tˤ]	22	ط [t]	26	ط [t]	33	ط [t]
17	ظ [ðˤ]	23	ظ [z]	27	ظ [z]	34	ظ [z]
18	ع [ʕ]	24	ع [a]	28	ع [a]	35	ع [ø]
19	غ [ɣ]	25	غ [ɣ]	29	غ [ɣ]	36	غ [ɣ]
20	ف [f]	26	ف [f]	30	ف [f]	37	ف [f]
						38	ڤ [pʰ]
21	ق [q]	27	ق [q]	31	ق [q]	39	ق [q]
22	ك [k]	28	ک [k]	32	ک [k]	40	ک [k]

	Arabic		Urdu		Pushto		Sindhi
						41	ک [kʰ]
		29	گ [g]	33	ک [g]	42	گ [g]
						43	ګ [gʰ]
						44	ڳ [ɠ]
						45	گ [ŋ]
23	ل [l]	30	ل [l]	34	ل [l]	46	ل [l]
24	م [m]	31	م [m]	35	م [m]	47	م [m]
25	ن [n]	32	ن [n]	36	ن [n]	48	ن [n]
		33	ں [-]				
				37	ڼ [ɳ]	49	ڻ [ɳ]
26	ه [h]	34	ه [h]	38	ه [h]	50	ه [h]
				49	ۀ		
27	و [w]	35	و [v/u/o]	40	و [v/u/o]	51	و [v/u/o]
		36	ھ [h]				
		37	ء				
28	ي [j]	38	ی [j/i/e/ɛ]	41	ی [j/i/e/ɛ]	52	ي [j/i/e/ɛ]
				42	ي [i]		
				43	ې [e]		
				44	ۍ [ay]		
				45	ئ [a]		
		39	ے [j/i/e/ɛ]	46	ے [j/i/e/ɛ]		

Fig. 1. Alphabet set of cursive script languages: Arabic, Urdu, Pushto and Sindhi.

The additional characters in Urdu make its script more complex to Arabic in appearance and introduce more challenges and subtleties in conceiving an OCR for the Urdu script. Therefore, any successful attempt in this direction would not only have a high commercial value but also benefit the Arabic and Persian readers. The pathway to an Urdu OCR is wrought with many challenges. The most important being the calligraphic style, the non-availability of commercial software, and the cursive nature of the script (especially the complex nature of the Nasta'liq style).

In our opinion, there is an obvious dearth of literature concerning the works par rapport the Urdu-like script character recognition. Therefore, this survey constitutes one of the rare efforts in compiling the works regarding Urdu-like script recognition with special reference to the Naskh and Nasta'liq writing styles. There has been some limited attempts, such as those by Sattar [3] and Fareen et al. [4]. The former survey presents Arabic script complexities and a few research papers focusing Arabic, Urdu, Persian, Jawi, and Uyghur languages [3]. The latter work is more specific and presents a concise review of the offline Urdu OCR, spanning the period of 2002–2009. However, the survey is far from being exhaustive and does not cover all of the the efforts. In addition to the aforementioned works, there are surveys pertaining to the offline and online Arabic character recognition systems, such as [5] and [6]; however, the focus of these surveys is on Naskh and the Nasta'liq script is not considered in the study. Therefore, these surveys overlook the peculiarities of Nasta'liq (Urdu) par rapport the Naskh, being discussed in Section 2, especially in the presence of additional letters, as well as the issued pertaining to the diagonality.

The rest of the paper is organized as follows. Section 2 introduces the cursive languages, followed by a brief discussion on the peculiarities of Urdu-like scripts. The next three sections review the contemporary literature since the year 2000 with the printed, handwritten, and OCR being dealt with, in Sections 3, 4, and 5, respectively. In Section 6, we conclude the survey.

2. Background

In this section, we report various characteristics and properties, peculiarities and issues of cursive script languages, with an added emphasis on Urdu. We also present some databases of Urdu scripts which are affected by the limited size and are not available commercially.

2.1. Cursive languages

Beside Urdu and English, there are many languages, spoken in Pakistan; Punjabi (in *Shahmuki* script), Pushto, Sindhi, Balochi, Saraiki, Hindko, and Brahui being the most important. Arabic to some extent is there because of being the language of Quran, the holy book of Muslims—the majority of population of the country is Muslim[1] [7]. Associated to the spreading of Islam, the Arab conquests of countries where people spoke languages not belonging to the Semitic family, the local population was compelled to adopt Arabic or at least use Arabic script for their local languages. They were bound to employ new letters to represent a variety of sounds that are absent in the standard Arabic scripts. Significantly, new shapes were invented to represent new sounds instead of borrowing from other scripts.

Barring English, all of the aforementioned languages follow cursive scripts and are Arabic based. Same is the case with other minor Pakistani languages, such as Persian, Gujarati, and Kashmiri. The most immediate superset of Arabic is the Persian script that in turn is a subset of the Urdu language. Sindhi and Pushto are supersets of Urdu script but no such relationship exist between the two and both have their own peculiar characters. Balochi and Punjabi, using the *Shahmuki* script, have scripts similar to Urdu. However, the Gurumuki script of Punjabi is not Arabic based. For this work, we are focusing on the Sindhi, Pushto, Urdu and Arabic scripts for which some research work can be found in the literature in the field of character segmentation and recognition with special references to Arabic. Fig. 1[2] tabulates the alphabet of the chosen four languages. The differences among the alphabets are due to some additional characters, with regards to the Arabic language, which are shown in red.

[1] http://www.bhurgri.com/bhurgri/downloads/PakLang.pdf
[2] http://en.wikibooks.org/wiki/Urdu/Alphabet

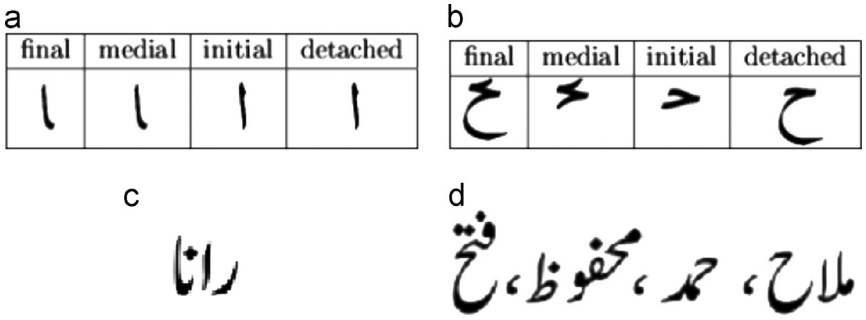

Fig. 2. Examples from Urdu, (a) the two shapes of *Alif*, (b) the four shapes of *Heh*, (c) the word "Rana" demonstrating both the forms of *Alif*, (d) demonstration of *Heh* shapes: R to L—"Mallah", "Hamd", "Mehfooz", "Fatah".

The Arabic language consists of 28 letters while the Urdu, Pushto and Sindhi languages are having 39, 46, and 52 letters, respectively. These languages are written cursively from right to left and due to cursive nature, the letters are (normally) joined with the adjacent letters within a word. However, a word may or may not have all of its letters joined together. Therefore, we have the terms *sub-word* or *ligature*. Ligature refers to the part of the word that has all of its letters (can be a single letter) joined together without any space in between. Therefore, a word may have one or more ligatures. Moreover, the shape of a letter depends on whether it is at the beginning (*initial*), middle (*medial*), end (*final*) of the ligature, or isolated (*detached*). Therefore, each letter has two to four different shapes depending on its position in the ligature. The letter *Hamza* has only one shape. The letters having just two forms are called *non-joiners*. They may either be isolated or may join with their preceding letter but not with any of the subsequent letters. For example, *Alif* has the two forms shown in Fig. 2(a). The letters with four different shapes are called *joiners*, e.g. *Heh* shown in Fig. 2(b). The two shapes of *Alif* are well demonstrated in the word *Rana* as shown in Fig. 2(c). Similarly, each word in Fig. 2(d) demonstrates one of the four shapes of *Heh*. The adaptation of the shape of these letters depend on the context. A sentence each from the Urdu, Pushto, Sindhi and Arabic langauge is shown in Fig. 3 for illustration purposes.

As already stated, a word may comprise one or more ligatures, such as the word "Pakistan" has three ligatures, as shown in Fig. 4 (a). The word itself is a combination of seven letters, separately shown in Fig. 4(b). The word "Tasbih" has just a single ligature (Fig. 4(c)), a combination of five letters (Fig. 4(d)).

A ligature may either be primary or secondary. A primary ligature is the longest continuous portion of the character that is written without lifting the pen. Such ligatures have also been referred to as Pieces of Words (PWs) or main strokes in some articles [8–10]. There are three ligatures in "*Pakistan*", namely "*Pa*", "*kista*" and "*n*", and one ligature in "*Tasbih*".

The secondary ligature is a set of diacritics—marks, accents or dots—that are written above or below the main ligature. Urdu has some significant number of marks, such as *Toy*, *Hamza*, and *Madaa* as represented in Fig. 5(a). Unlike Arabic, the diacritics *zabar*, *zeir*, *pesh*, and *shadd* (shown in 5(b)) are not that common in Urdu scripts. Dots that are part of the letters, have also been considered as a diacritics in some works about Urdu scripts [5]. However, it is not a standard practice and we will consider that beyond the scope of this survey.

2.2. Peculiarities and Challenges in Urdu Script Languages

Of the cursive script OCR's, a lot of research work has been published on Arabic, followed by Persian (also called *Farsi* in the literature). As far as the Urdu script languages are concerned, an established OCR is still a far cry. However, there are still some

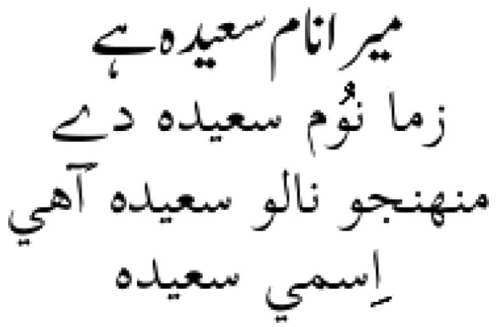

Fig. 3. The sentence "My name is Saeeda" in Urdu, Pushto, Sindhi, and Arabic, respectively.

commendable efforts in the context of Urdu and Jawi (spoken in Malaysia), but other languages, such as Pushto and Sindhi, have not received the attention they deserve.

The development of Arabic calligraphy led to the creation of several decorative styles that were designed to accommodate special needs. The most outstanding of these styles, as already stated, are: Nasta'liq, Koufi, Thuluthi, Diwani, Rouqi, and Naskh. An example of each one of these styles is given in Fig. 6. The Nasta'liq is mostly followed for Urdu, Punjabi, and Sindhi. Whereas, the Naskh is mostly followed for Arabic, Persian, and Pushto (see Fig. 12). The difference between the two styles is very significant for Urdu. The Naskh style is closer to the traditional Arabic style except for some final letters. The aesthetic style is more pronounced in Nasta'liq with very oblique ligatures. It also shows more pronounced variations of the letters according to their position in the word. For example, the character *bey* contains 32 shapes shown in Fig. 13.

A large set of characters and similar-shaped-characters make the case of the Urdu-like scripts more complex and challenging [11]. When it comes to the Nasta'liq font style, the Urdu scripts are written diagonally with no fixed baseline. The lack of any standard for slopes, escalates the matter further. Moreover, the style is highly context sensitive due to the shape, and to some extent due to the existence of filled or false loops. Furthermore, the position of the characters as well as character/ligature overlap introduce myriad issues[3] [12,13]. All of these complexities pose a significant challenge for Nasta'liq in comparison to the Naskh style. The main problems and peculiarities will be presented here from the recognition point of view for the Nasta'liq style:

2.2.1. Bidirectionality

The Urdu script languages are bidirectional languages. Mainly, the right to left direction is used for reading and writing the text.

[3] http://en.wikipedia.org/wiki/Nastaliq_script

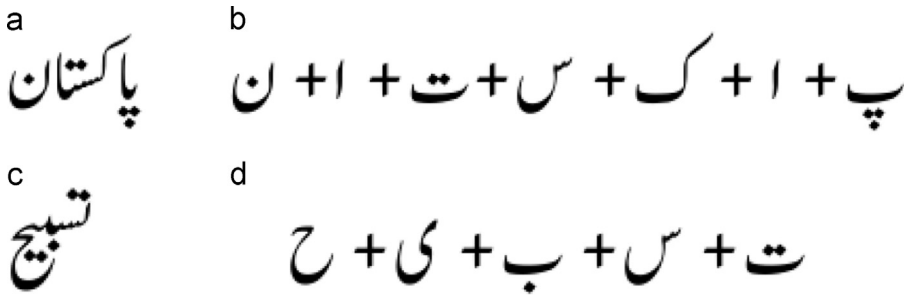

Fig. 4. Examples of Urdu words and their configuration: (a) "Pakistan", (b) separate letters of "Pakistan", (c) "Tasbih", (d) separate letters of "Tasbih".

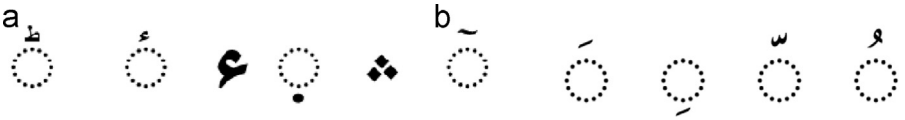

Fig. 5. Urdu diacritics. (a) Common: Toy, Hamza (2), dots (2) and Madaa, (b) uncommon: zabar, zeir, shadd and pesh.

Nastaliq	
Koufi	
Thuluthi	
Diwani	
Rouq'i	
Naskh	

Fig. 6. Different styles of Cursive scripts.

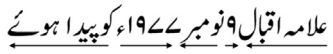

Fig. 7. Bidirectional nature of urdu scripts languages [16].

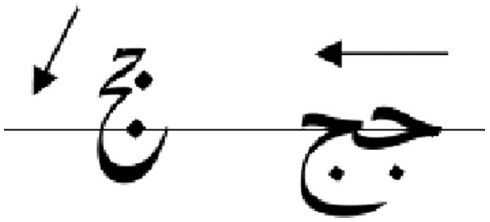

Fig. 8. Diagonality nature of Urdu scripts languages.

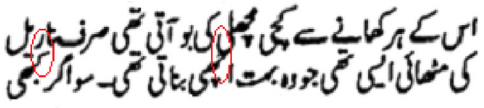

Fig. 9. Touching of ligatures in two different lines.

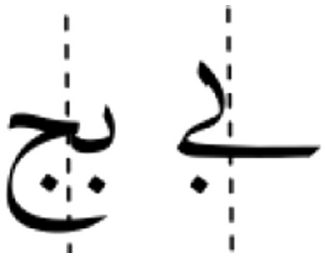

Fig. 10. Non-monotonic writing in Urdu [19].

However, left to right direction is used for reading/writing the numbers in both printed and handwritten forms [14,15]. The bidirectional behavior is demonstrated in Fig. 7.

2.2.2. Diagonality

The diagonality is attributed to the fact that the Nasta'liq font style is written from top right to the bottom left using some well-defined rules, which means that all of the ligatures are tilted at a certain angle towards the right side and the angle is variable depending on the characters being written [12–14]. Fig. 8 demonstrates the diagonality.

Although, the diagonality economizes the horizontal space consumption as compared to the Naskh typesetting, the downside is the possible vertical overlap of characters within and across the ligatures. Therefore, a word with excessive number of letters in a given line may clash with the words written in the underneath line [17,18]. Fig. 9 illustrates this behavior.

2.2.3. Non-monotonic writing

Urdu-like script languages do not have the monotonic writing style of English in which the characters start from the left and are written towards the right direction. However, in Urdu-like scripts, one frequently goes back to the already written character as certain letters consist of a stroke that goes back and beyond the previous character. It is shown in Fig. 10 that the strokes for the second letters, *Barhi Yay* and *Jeem*, go towards the right (or backwards) beyond the previous characters [19,20]. The first letter is *Bay* in both of the cases. It poses complexities and greatly limits the implementation of a robust OCR [14].

2.2.4. Cursive writing style

Urdu-like script languages use the cursive writing style in which the letters connect at delicate joints in a word [19,16]. A single word in the script can be composed of several ligatures that are formed by combining several characters cursively joined together. Some examples are illustrated in Fig. 11.

2.2.5. Graphism multiplication/Context sensitivity

Each character changes its glyph shape in accordance with the neighboring character, which is called context-sensitivity [13,21,22,10]. To clarify, it means that a letter assumes different shapes according to the context in which it occurs [19]. Usually,

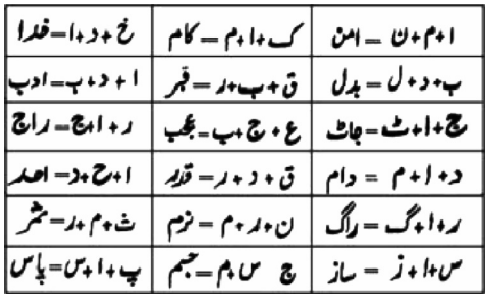

Fig. 11. Cursive writing style of Urdu scripts languages [11].

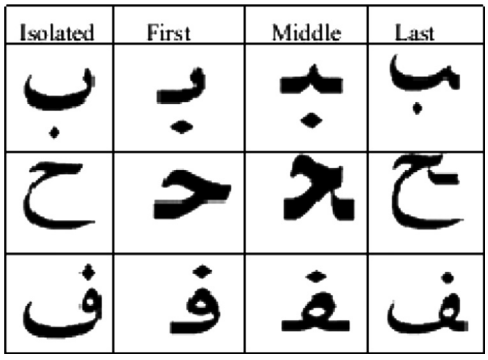

Fig. 12. Basic four shapes of characters of Urdu.

characters have two to four shapes in Urdu scripts and character changes its shape according to its position in the ligature. The basic four shapes of three characters from Urdu are given as an example in Fig. 12. With Nasta'liq, the situation becomes more complex as the number of possible shapes may escalate to as high as 60 excluding the isolated shapes[4] [13,19]. These shapes depend on the preceding and the following characters [12,23]. Various possible shapes of *Bay* are illustrated in Fig. 13 as an example. Therefore, special coding rules are required to cater for the context sensitivity in Nasta'liq, as there are about thousand different shapes for less than 50 characters [19].

2.2.6. Characters' overlap

The complexity of the overlapping may be very much between characters and subwords (ligatures) in the Urdu-like script languages. The characters are overlapped vertically. However, the characters do not touch each other. The overlapping in ligature is required to avoid the unnecessary white space. For example, *Kaf* is overlapping with *Tay* in the sentence given in Fig. 14, as shown by the rightmost oval in red color. An overlap may also be intra-ligature (Fig. 15) or inter-ligature (Fig. 16), both leading to ambiguities [24,13].

2.2.7. Upper/lower cases

There are no upper or lower cases in Urdu-like scripts [5]. However, the last character in a word is in its complete shape and it is (commonly) considered as an upper case [23].

2.2.8. Baseline

Every script needs a base line—a horizontal line—on which the text can be written, which cut all the words at some point [10]. A baseline plays an important role in skew detection. The Naskh style has a baseline on which characters are written independent of their connectivity; however, the Nasta'liq style has a virtual line as a baseline, on which the text is combined/joined [26]. The modeling of Urdu text-line on different descender lines is reported in Fig. 17.

Each character of the Nasta'liq style may appear at baseline and the two descender lines depending on the associated characters, whereas the last character appears on one baseline and does not depend on its connected character, as shown in Fig. 18.

2.2.9. The number and position of dots

Some cursive script characters have dots associated with the character that can be above, below, or within the character. These dots also distinguish the character from any other similar character. For example, with two similar characters:

- one character may have a dot while the other lacks it, or
- both may have different number of dots, or,
- both may have the same number of dots but placed at different positions.

Dots may become as touched dots, hat, or a stroke combining two dots by the computer-editor/type-writer/individual, e.g the dots touch each other variously in Fig. 19.

2.2.10. Complex placement of dots

In Urdu-like script languages, a character can have one to three (four in the case of Sindhi) or zero dots placed above, below, or inside. However, the rules for the standard positions of dots can alter due to the features of slopping and context sensitivity. This is due to the fact that these features do not provide enough space for the dots' placement at standard positions (inside or right below) the character in many situations. Therefore, the dots will move from their standard position to some other nearby position. This displacement may lead to a situation where it is difficult to associate dots to the correct primary component [27,13]. This factor is particularly important in the case of characters given in Fig. 19.

2.2.11. Stretching

In Nasta'liq, justification is usually carried out via stretching (Fig. 20), if space is to be occupied. Stretching means that the characters change their standard version into a longer versions due to which some characters even change their default shape while some only change their width. For example, the characters *Seen, Sheen, Bey, Fey, Gaaf, Qaaf* frequently exhibit the aforementioned phenomenon. Fig. 21 illustrates the phenomenon in the case of *Seen*. The stretching factor may make the machine recognition difficult.

2.2.12. Positioning

The justification feature is also achieved using positioning, if the objective is to economize the space (see Fig. 22). By positioning we mean that a ligature or a sub-word is placed on top of a previous ligature in the same word or any adjacent word. Such positioning is commonly and extensively used in the Urdu scripts to accommodate long and big headings within small spaces, which introduces more challenges and difficulties in machine recognition.

2.2.13. Spacing

In Urdu-like script languages, the spaces have the same definition as in English and may occur between two words, inside a word, or between ligatures. However, these spaces may vary in size. These are also used to justify a text in Urdu script, a practice uncommon and usually used as a second alternative. Fig. 23 illustrates various spacings in an Urdu text example. Intra-word

[4] http://www.cle.org.pk/clt10/acceptedpaper.htm (Article: Urdu Writing Rules for Online Input in PDA's).

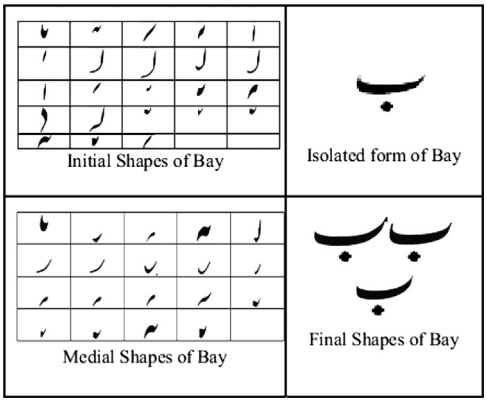

Fig. 13. Different shapes and isolated form of a character of Urdu [12].

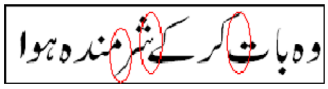

Fig. 14. The overlapping nature.

Fig. 15. Intra-ligature overlap.

Fig. 16. Inter-ligature overlap.

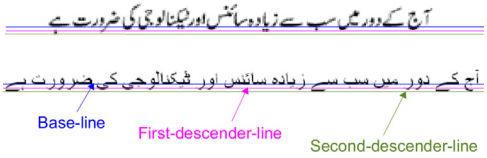

Fig. 17. Baseline and two descenter lines for Naskh and Nasta'liq [25].

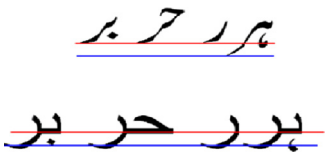

Fig. 18. Red is Baseline for Naskh and Nasta'liq, Blue Line is Challenges in Baseline [26]. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

Fig. 19. Complexity in dot placement and association with base character.

spacing breaks a word into its separate constituent ligatures that is not allowed in Urdu-like scripts. The feature of inter word spacing also introduces complexities in the character recognition of Urdu-like scripts.

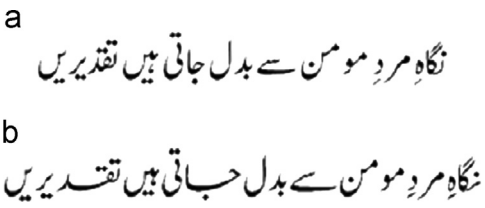

Fig. 20. Stretching of characters in Urdu [22]. (a) Unstretched version and Streched version.

Fig. 21. Stretching of *Seen*.

Fig. 22. Positioning of character in Urdu [22].

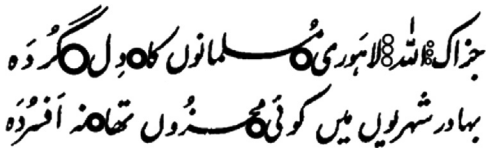

Fig. 23. Spacing is shown by circle in Urdu sentence [22].

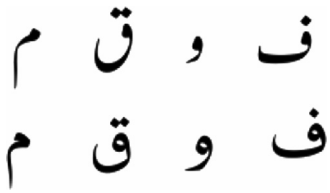

Fig. 24. Filled-loop characters in Nasta'liq and open-loop characters in Naskh.

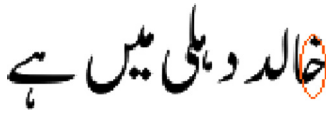

Fig. 25. An example of false loop.

2.2.14. Filled Loops

There are some characters—such as *Meem, Qaaf, Wao and Fey* – having a small loop in Urdu-like scripts. These character loops are usually filled from inside in the Nasta'liq script and open in the Naskh script, as shown in Fig. 24. This property of Nasta'liq makes the character recognition more complex, because it causes the characters to become identical with other characters. For example, it becomes difficult for an OCR system to distinguish *Wao* from *Daal*, especially after applying the filling method [13].

2.2.15. False Loops

There are also some characters, such as *Jeem, Chey, Hey, Khey*, when written in Nasta'liq, the starting point joins with the base resulting a false loop (see in Fig. 25). This is challenging for OCRs to

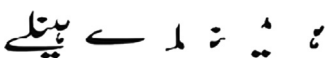

Fig. 26. Horizontal and vertical Segmentation of Ligatures in Nasta'liq [29].

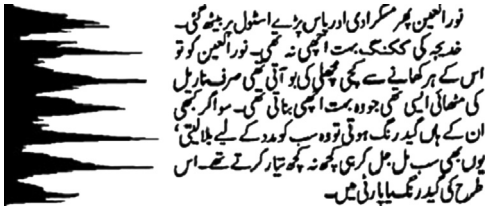

Fig. 27. Horizontal profiling in Nasta'liq [28].

recognize false loops or distinguish them from the characters with real loops.

2.2.16. Ambiguous Segmentation Cue Point

Unlike Naskh, the Nasta'liq scripts have more than one imaginary baseline [28] that makes segmentation cue point identification of the ligature very difficult. As a result, there are non-identical cue points in each of the ligature depending on whether the connected character precedes or follows. Fig. 26 illustrates the complexity of cue point extraction wherein the Nasta'liq ligature is segmented into the character or smaller pieces thereof.

2.2.17. Direction of segmentation

As Nasta'liq is written diagonally from right-to-left and top-to-bottom, the characters are to be segmented in both the horizontal and vertical directions, as shown in Fig. 26.

2.2.18. Text line Segmentation and Interline Spacing

Usually, the projection profile method is used for Arabic scripts, which computes the horizontal profile (wrongfully called *histogram* in some texts) of text lines (due to the latter's large interline spacing) and segments where the profile has zero values. However, this method cannot be applied to Nasta'liq, where the ligatures overlap in horizontal/vertical projections and exhibit smaller interline spacing as shown in Fig. 27.

Although the problems of graphism multiplication, characters' overlap, ambiguous cue point segmentation and text line segmentation are not unique to Nasta'liq, such issues become far more challenging in comparison to Naskh, in the light of the above discussion. For instance, the problem of graphism multiplication becomes far more glaring in the case of Nasta'liq, where it may result in 32 different shapes for one character form in contrast to four in the case of Naskh. Challenges, such as diagonality, multiple baselines, complex placement of dots, stretching, positioning, and filled and false loops are properties that are unique to the Nasta'liq script. All of these are still open research problems for Urdu Nasta'liq OCR. It is to be concluded that on one hand the context-sensitivity and sloping makes the character segmentation and recognition a very difficult task and on the other hand, the positioning, stretching, and filled loops makes the ligature and sub-word segmentation even more difficult.

2.3. Databases

One of the central issue in OCR is of finding a good database for offline (both machine printed and handwritten) and online recognition. There are many large databases available for Latin script as the underlying research is far more mature. Even the case of Arabic databases is envious and far more established when compared to the Urdu-like script languages, such as Urdu, Pushto, and Sindhi. Because the aforementioned are more complicated and complex in nature, the Urdu script OCR and related databases are still in their infancy and little resources are available, currently. Therefore, it necessitates the construction of an exhaustive database for each of the above mentioned languages. This section dwells on the few available contemporary databases.

At the Center for Pattern Recognition and Machine Intelligence (CENPARMI) in Montreal, Canada, Sagheer et al. [30] have designed a database of 109,588 images from 343 various writers (both sexes) for the recognition of Urdu offline handwriting containing:

- 60,329 samples of isolated digits,
- 12,914 samples of numeral strings with/without decimal points,
- 1705 samples of five special symbols,
- 14,890 samples of 44 isolated characters,
- 19,432 samples of 57 financial related Urdu words and
- 318 samples of Urdu dates in different patterns.

A large number of Urdu native speakers from various regions of the world had been involved in collecting the data. The database has text images in different forms, such as true color, gray scale, and binary.

The authors in [31] discuss about the development of the FAST-NU Pushto image database having text in four font sizes for the potential researchers working towards a robust Pushto OCR. The database contains about 4000 denoised images from 1000 ligatures of Pushto language without any skew. However, the database, as is evident, suffers from its limited size.

An unconstrained database has recently been announced [32] that contains 400 forms produced by 200 different writers with automatic line segmentation for Urdu offline handwriting. It had involved intakes from six sources: (a) sports, (b) science fiction, (c) entertainment, (d) Urdu blogs, (e) religious writings, and (f) editorials. The database, having 23,833 printed Urdu words in 2051 lines of text, is named the Center for Image Processing-Urdu Corpus Construction Project (CENIP-UCCP).

2.4. Optical character recognition (OCR)

Based on the mode of input, OCR is classified as offline and online. The offline OCR deals with the image of the already written text—handwritten and machine printed—and its input is obtained through an optical scanner or digital camera. In contrast, in the online OCR, the input text is written directly using a tablet, a PDA, or a stylus. The online character recognition is probably easier than its offline counterpart as more information is available, such as time information, stroke coordinates, and handwriting style of the user. A typical OCR system may consist of some or all of the five components, namely the: (a) image acquisition, (b) preprocessing, (c) segmentation, (d) feature extraction, and (e) recognition/classification [33,5,34,35], followed by some post-processing [36,12]. Fig. 28 shows the block diagram of a character recognition system.

The *preprocessing step* is an important step that may consists of binarization, filtering, smoothing, slant correction, skew detection, thinning (Skeletonization), and baseline detection to improve performance of the OCR system. It directly affects the reliability and efficiency in the rest of steps [37,38]. This step prepares the input image for the subsequent segmentation and feature extraction steps. What we do in this step, solely depends on what kind of input image we expect and what we need to do with the feature extraction at a higher precision. The preprocessing step of Urdu text recognition may require special finesse to carry out tasks, such as separation of dots touching the base ligature. Due to the diagonality and slantiness of Urdu text, skew detection and correction may eventually prove very troublesome.

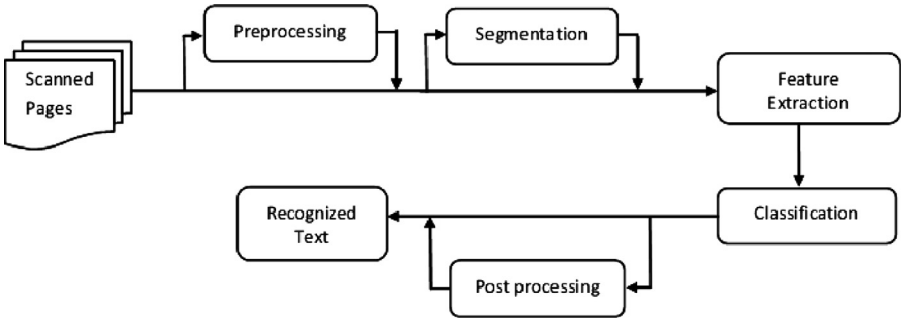

Fig. 28. Steps of OCR for cursive languages.

In the *segmentation* step, the text of a paragraph is segmented into lines and then the lines into words or sub-words/ligatures. To further segment the ligatures or not, categorizes the OCR approaches into:

1. The segmentation-free or holistic methods.
2. The segmentation-based or analytical methods.

In the first approach [23,33], the ligatures are not further segmented and the system seeks to recognize the ligature as a whole word. Usually, the paragraphs are split into text lines in the document images using a horizontal projection method. Thereafter, the text is separated into words or sub-words with a vertical projection or connected component method in the Urdu OCR systems. The second approach, in which the ligatures are resolved into letters/independent units/strokes, is further classified into two strategies, namely the direct and indirect segmentation. In the *direct segmentation*, a word is separated directly into the required letters using a number of heuristics that identify all of the segmentation points of a character. In the *indirect segmentation*, a ligature is resolved into primitives by splitting it into smaller elements or strokes that might be letters or less than letters, such as sub-letters or small strokes. The latter approach is much expensive in terms of time for finding the optimum word from the units' combinations. Segmentation is a challenging and important task in recognizing Urdu script and this issue needs special attention because it directly affects feature the extraction and classification steps [39,40].

Feature extraction, in pattern recognition problems, involves the extraction of unique and salient patterns from the input image to enhance the discriminatory power and reduce the data for classification. The success of a classifier depends on the feature extraction. The extracted features can be classified as [41]:

1. *Structural features*
2. *Statistical features*
3. *Global transformations*

The *structural features* are those features that are obtained from the structure or typologies of the characters or letters. These may be closed curves, zigzags, intersection points, vertical and horizontal lines, dots, width, height, number of crossing, branch points, and end points. Usually, these features are computed from the skeleton of the underlying text image. Structural features are particularly important for cursive scripts, in comparison to the Latin scripts due to their success in capturing the dot/accent information explicitly. The above mentioned is quite essential in differentiating the letters having the same shapes but differ in accents or the number of dots. The *statistical features* give us a statistical distribution of some measurable event or phenomenon of interest in images or region of images. These are usually computed as some quantifiable numerical measure. These may be information leading to zoning, characteristic loci, crossings and distance, number of pixels, and moments [33]. Such features can be extracted quickly and effectively. However, they are prone to external noise. The *global transformations* are used to shorten the text representation. The affect being a reduction in the feature data to obtain better results. The global transformations can be horizontal and vertical projections, Freeman chain code, Fourier transform, Hough transform, and Gabor transform.

Based on the extracted features, the *classification and recognition* is the main decision making stage. The pattern is identified and recognized from the input features. As stated earlier, there are two general approaches, namely; holistic and segmentation based (analytical) [41]. In the *holistic recognition* approach, the ligatures are processed as a whole, eliminating the segmentation problem, and the recognition system is trained using some machine learning algorithm. However, it has a limited vocabulary available for processing. In the *segmentation based recognition*, each word segments into a sequence of smaller components, such as characters, strokes or graphemes from an unlimited vocabulary. Thereafter, it tries to recognize each segmented unit. This approach is generally used with structural features.

The *post-processing* is the last stage in the OCR systems and includes tasks, such as spell-checking, grammar corrections, and language dictionary with an aim to improve the recognition output.

Urdu script OCR is quite a challenging task because of the cursive nature of the script. The literature is replete with the works dealing with the complexities of the Urdu script. The rest of the paper presents a comprehensive overview of the OCR methods proposed since the year 2000.

3. Printed Urdu character recognition

With the turn of the century, works on the automatic recognition of printed Urdu script started to appear [42–44,11]. The interest can be gauged from the fact that a number of theses were dedicated to the subject [45,23,46–48]. Beginning March 2011, a serious effort is already underway to develop the first commercial level OCR system for printed Urdu Nasta'liq script.[5] Work on Pushto and Sindhi is also gaining momentum day by day [49,50]. For the sake of clarity, we are classifying the works, reported for machine-printed Urdu OCR, according to the stages of the OCR pipeline.

3.1. Preprocessing approaches

Preprocessing includes document scan, orientation detection, skew correction, noise removal, binarization, and many other

[5] http://www.cle.org.pk/research/projects/Details/ocrnew.htm

operations that may be needed to prepare for the successful execution of the segmentation techniques. Therefore, it plays an important role in both the layout analysis and robust OCR construction. A recapitulation of different preprocessing techniques is given in [51], for Urdu, Arabic, Persian, and Jawi. The authors employed spatial max and median filter, histogram equalization, and frequency domain Gaussian low-pass filter with the objective of enhancing the dark and noisy Urdu document for the later steps of segmentation and feature extraction.

Page orientation is defined as the printing direction of the text lines with the upright position of characters in a document (whether in the portrait mode or in landscape mode). The angle that the text lines make with the horizontal direction in a digital text image is called the skew angle of the document [52]. The proper page orientations and accurate skew corrections enable better document analysis [53]. Using a discriminative learning approach, such as convolutional neural networks, Rashid et al. [54] proposed an orientation detection method for Urdu document images with different layouts and fonts. The authors tested their system on a dataset of Urdu document images categorized into the layouts of book—prose and poetry—and achieved almost 100% accuracy in the orientation detection. The approach is script independent and can be applied accurately to other scripts. A moment based method for the estimation and correction of skew angle in Urdu document images is given in [53], wherein the central moments and the centroid of the image have been used to calculate the skew. The ensued results were found to be far enviable for printed documents than the handwritten ones. Pal and Sarkar [11] have exploited the Hough transform, for skew detection and correction, which picks the selected components and computes the results on the chosen candidate points.

In an OCR system, the layout analysis is another key preprocessing step for the effective text-lines extraction and reading order determination. While relying on an existing system for Roman script, the Recognition by Adaptive Subdivision of Transformation Space (RAST) [55], Shafait et al. [25] present a layout analysis system for extracting text lines from an image of an Urdu document. The authors made some modifications to RAST that include:

1. For column separators, they examined the empty white space rectangles followed by the extraction of text lines by introducing two descender lines.
2. Right to left reading/writing order, which the Urdu script follows.

The approach demonstrated considerably accurate results with many documents, such as books and newspapers. However, the accuracy suffered a bit with small inter-line spacing, and in the presence of enumerated lists and high diversity in font sizes. Another layout analysis strategy was reported in [56] that uses a combination of well-established techniques for text and non-text segmentation (by a multi-resolution morphology method [57,58]), text-line extraction (a ridge based method) and reading order determination (topological sorting [25,55]). The approach exhibited good accuracy and the system was evaluated on a variety of complex single and multi-column document images. However, the system underperformed on large datasets. The same authors put forward another amalgamation of methods in [28]—which essentially is an extension of [25]—by employing some well-established and robust techniques of Latin script documents for the layout analysis of machine-printed document images having a variety of styles (Naskh and Nasta'liq). Their binarization step employed the well-known Otsu's [59] and Sauvola's [60] methods. For the text/non-text segmentation, the authors relied on their own multi-resolution morphology technique [57] that has the ability to segment drawing type non-text elements. Standard image processing methods [61,62], such as filter bank Gaussian smoothing were involved—for robust text line detection—in combination with the ridge detection [63,64] that had been successfully used in [65,66] for Arabic script document images. The aforementioned was followed by the whitespace analysis of [25,67] before line labeling. Finally, the authors used the approaches of [25,55] for partial text line reading ordering detection and then the topological sorting algorithm [68] for the detection of a complete text reading order.

3.2. Segmentation approaches

A scanned text document is usually a collection of paragraphs, each of which is a collection of sentences and sentences have connected or partially connected character strings. After the layout preprocessing—orientation detection, skew elimination, and layout analysis—one ought to carry out segmentation. The segmentation refers to the separation of the the paragraphs, text lines, words, characters, and strokes for effective feature extraction. Segmentation is a challenging task, especially in cursive script OCR [40,39], and directly affects the subsequent stages of feature extraction and classification. The pre-recognition process, outlined in [36], based on font size 36 for Urdu Nasta'liq has been claimed to achieve 100% accuracy in the baseline identification and 94% accuracy in the ligature identification. The method relies on the horizontal projection profile to segment the lines and separate ligatures from the diacritics while looking to the 8-neighbors. Some errors may ensue that are avoided by using a number of heuristics. The association of dots and marks, with the relevant base forms, is carried out with the help of centroid-to-centroid distance. For analysis, a total of 1282 unique ligatures were identified, out of 5000 high frequency words, from a corpus mentioned in [69]. The same set of ligatures were also employed in the segmentation free method outlined in [12]. The horizontal profiling had already been exploited by Ref. [43] for the separation of connected components and the baseline, segmentation of words into ligatures and segregation of primary and secondary ligatures. Similarly, for the segmentation of words into ligatures, the centroid-to-centroid method had also been formerly applied in [44], in conjunction with the connected component labeling.

The image signature scaling based method of Azam et al. [70], carries out both of the horizontal and vertical scaling of Urdu, Arabic, and English text images, for the signature calculation. The count of black pixels in the binary image of the character, is the signature value. The authors computed 600 different signature values of a normalized character by segmention into three horizontal and three vertical sub-segments as per signature scale for of the both horizontal and vertical segments. In [71], each word is scanned and analyzed for segmentation using three levels of complexity, *viz.* simple, semi-complex, and complex. The three levels measure the number, width, height, direction of holes and points in lines, and the distance between two lines of a character by using a fixed font size of Nasta'liq. A segmentation based technique proposed in [23] for the recognition of font size 36 *Noori* Nasta'liq script on compound ligatures, handled (without formatting) the six classes of characters in single-column documents. The diacritics and main bodies are separated first and then thinning is applied. The ligatures are segmented at the point where more than one outgoing directions are found. In a segmentation free approach [72,73] for the extraction of ligature from an image of Nasta'liq Urdu text, horizontal projection profile was used. This approach isolated the character shapes, enabling the ligatures and diacritical marks to be identified via connected component labeling. For the segmentation of Naskh Font, the system, proposed in [74], measures the strength of the pixels to identify words in a sentence and joins of characters in a compound or connected word.

Table 1
Review of segmentation approaches.

Authors	Fonts	Data sets	Segmentation approach	Accuracy
Shah et al. [43]	Nasta'liq	Unspecified	Connected component labeling and baseline method	79%
Husain et al. [44]	Nasta'liq	200 ligatures	Connected component labeling and centroid to centroid distance method	100%
Pal et al. [11]	Nasta'liq	Small variety characters	Horizontal and vertical profile, component labeling	96.9%
Azam [70]	Nasta'liq	Unspecified	Horizontal and vertical method with signature scaling factories	unspecified
Ahmad [71]	Nasta'liq	Unspecified	Measuring number, width, height and direction of holes; points in lines and distance between two consecutive lines of characters	unspecified
Sattar et al. [72,73]	Nasta'liq	Unspecified	Horizontal projection profile and connected component labeling	unspecified
Ahmad [74]	Naskh	Unspecified	Measured strength of pixels and use of joints	unspecified
Javed et al. [12]	Nasta'liq	1282 unique sub-words from 5000 frequently sub-words	Horizontal and vertical profile	92%
Javed et al. [36]	Nasta'liq	1282 unique sub-words from 5000 frequently sub-words	8-neighboring method, and horizontal pixel count for diacritics and main body separation; heuristics for baseline detection; centroid to centroid method for diacritics and dots association	94%
Malik and Fahiem [75]	Nasta'liq	Unspecified	Horizontal pixel count, FCC, label matrix, vertical scanning	99.4%
Akram and Hussain [76]	Nasta'liq	150 sentences composed of 6075 ligatures and 2156 words	Ligatures used as a structural method, trigram trained on co-occurrence information of ligatures and words in the corpus	96.10% Known words 65.63% Unknown words
Rehman et al. [77]	Nasta'liq	not given	A portion between two points used as a segment point, FCC	unspecified
Abidi et al. [8]	Nasta'liq	50 text images	horizontal profile for line segmentation, connected component extracted the ligatures	unspecified
Shaikh etal. [50]	Naskh	Unspecified	Calculate height profile vector of primary ligature skeleton and determine segmentation point and use some rules	Unspecified
Mahar et al. [78]	Naskh	16,601 words	Holistic approach using five methods for segmentation	9.54% Segmentation error rate (SER)

Separation of primary and secondary ligatures is carried out in [75] by applying the freeman chain codes (FCC) and the vertical scanning for primary ligature extraction. The work in hand exploits the statistical information about the height and width of each ligature to identify the font size. While relying on the co-occurrence information of ligatures and words, Akram and Hussain [76] have realized the word segmentation through the construction of the trigram probabilities normalized over the number of ligatures and words in the sequence. The authors evaluated their model on the manually developed and cleaned corpus and claimed to have achieved an identification rate of up to 96.10% with known words and 65.63% with unknown words. The FCC oriented segmentation, in [77], is based on the scale invariant and optimized boundary representation code for the Nasta'liq characters. The authors developed a small-scale invariant code, after computing the ratio of each segment to the whole image, instead of storing and computing the chain code of each pixel for the segments of characters. For the retrieval of documents by word spotting, Ref. [8] extracts the connected components in the binarized image of printed Urdu text to segment the later into ligatures or partial words (PWs) whereby each of the segmented ligature is represented by a set of two scalar and four vector features that are stored in the underlying database. In the context of recognition of Sindhi characters, authors in [78] have presented five algorithms for the segmentation of the words into meaningful linguistic sequences and created a lexicon resource containing 16,601 words for the proposed algorithms. The proposed system exhibited cumulative segmentation error rate (SER) of 9.54%. Another Sindhi based work [50] addresses the sub-word segmentation into characters by calculating the height profile vector of thinned primary strokes of various types and sizes followed by the number and locations of possible segmentation points (PSP).

All of the aforementioned methods are summarized in Table 1, along with the data sets and claimed accuracies. The calligraphic nature of the Nata'liq writing style, character segmentation of Urdu text is a highly challenging problem as compared to Arabic and Persian. Consequently, works already achieved for the latter languages cannot be directly applied to Urdu-like languages (Tables 2–6).

3.3. Feature extraction approaches

The extraction of features of all of the individual Urdu characters and numerals are performed by discovering the extreme points, such as left, right, top, and the bottom in both the training and testing document images [79]. A number of works can be found on the detailed treatment of generic feature extraction [80–82].

Among the earlier research works, Megherbi et al. [42] relied heavily on the structural features. These include the number of dots present in the character, place of the dot, branch or presence of secondary stroke, aspect ratio (height to width) and slope between the initial point and the final point. Structural features also play an important role in the method, outlined in [43], wherein visible features, such as the location of the dots and placement of other diacritics are extracted for every ligature. Ref. [71] extracts structural features, such as character lengths, number/position of loops or holes, and distance between two consecutive lines for their OCR system. The structural features are again important in the works of [72,73]. In yet another work [83], the list of structural features includes loop, curve, cross, height of character, width of character, number of the dots, and position of the dots. In [84], the extracted features include height, width, and checksum from each character that differentiate one character from another. The width of a character is calculated by counting the black pixels and then calculating the difference between the first and the last black pixel in the direction from left to right. Similarly, the height of the character is calculated by counting the black pixel and then by calculating the difference between the first and the last black pixel in the direction from top to bottom.

The work reported in [44] is based on the extraction of the statistical features, namely the axes ratio, solidity, eccentricity, moment based features, normalized length features,the number of holes, and the curvature feature to identify the ligatures in Urdu *Noori* Nasta'liq script. First, special ligatures, such as *Mada*, *Dots*,

Table 2
Machine printed isolated character recognition.

Authors	Features	Classification	Dataset	Accuracies
Hussain et al. [82]	Signatures	Template matching	Unspecified	81.82%
Azam et al. [70]	Signatures scaling factor	Template matching	Unspecified	70.29%
Pal et al. [11]	Topological, contour & water reservoir	CART Tree	3050 characters	97.8%
Nawaz et al [90]	Chain code	Matching technique	Unspecified	89%
Shamsher et al. [79]		FFNN	Unspecified	98.3%
Ahmad et al. [71]	Distance between lines, joining points, topological features, and the number, width, height & direction of holes	FFNN	Unspecified	93.4%
Tariq et al. [84]	Height, width, and checksum	Matching using Data base	Unspecified	100% Soft matching and 97.3% Hard matching
Zaman et al. [88]	Pixels values using row-major or column-major order	Feed-Forward Neural Network	106	95%
Megherbi et al. [42]	Structural	Fuzzy logic rules		

Table 3
Machine printed ligature or word recognition.

Authors	Features	Classification	Dataset	Accuracies
Shah et al. [43]	Structural	Template Matching	Uspecified	79%
Husain et al. [44]	Solidity, number of holes, axis ratio, eccentricity, moments, normalized segment length, curvature, ratio of bounding box width and height	FFNN Back	200 ligatures	100%
Ahmad et al. [74]	Pixels strength, joints of characters	Multilayer FFNN	56 different classes of characters each having 100 samples	70%
Sattar [72,48]	Structural	Cross-correlation	Unspecified	Unspecified
Sattar [73]	Height, thickness, angle, rotation & joining ends	Finite state model	Unspecified	Unspecified
Hussain et al. [83]	Height, width, loop, curve, lines, & joint	Kohonen Self-organizing Map (K-SOM)	104 segmented character ligatures	80%
Javed et al. [12]	DCT	HMM	1259 unique ligatures from 5000	92%
Decerboet et al. [49]	Statistical	HMM	27000 characters	1.6% CER and 5.1% WER
Ahmad et al. [86]	SIFT	Matching	1000 ligatures	74%.
Lodhi et al. [85]	Geometric transformation	–	–	–
Sabbour and Shafiat [92]	Contour and shape context	k-Nearest Neighbor	over 10,000 ligatures	91%

Tay, and *Hamza* are separated out. Thereafter, information with rotation, translation, and scaling (RTS) invariant features are extracted from the base ligature into a set using the RTS invariant moment. Last, the extracted special ligatures are linked to the most probable neighboring base ligature using the centroid-to-centroid distance. A research study [11] extracted a combination of topological, contour, and water reservoir features for the isolated or individual characters of Urdu language. Lodhi and Matin [85] have proposed a RST invariant method that considers Fourier descriptors for the feature selection of Urdu characters. Fourier descriptors are used to uniquely represent the given characters' polygonal signatures.

In [12], the global transformation method is utilized for the extraction of features from a non-segmented ligature. Ref. [86] extracts the scale, rotation, and location invariant features from a Pushto document images using the scale invariant feature transform (SIFT) descriptors [87] on four various sizes and orientations. A comparison of the ensued recognition results against the classical methods, such as principal component analysis (PCA) attempts to establish the effectiveness of the former. In [88], the row-major or the column-major order is used after applying various preprocessing methods for the conversion of a normalized image into a row vector of binary values as a feature input.

3.4. Recognition and classification approaches

One of the pioneering effort on Urdu character recognition, classified the Urdu characters in seven classes by developing fuzzy logic rules [42]. The method in [43] performs recognition and classification of Urdu Nasta'liq ligatures in two passes. The first pass employs the template matching for recognizing the diacritics. In the second pass, the recognition strategy is applied to the main body for the visible features to correct any mis-identifications of diacritic. Another template matching approach is reported in [72,48], which relies on the cross-correlation and maintains a file of the character shapes of the Nasta'liq font. Each of the character shape in the font file is matched, line-by-line, with the shapes identified in the text image using cross-correlation. Concurrently, the system writes the character codes into a text file in the sequence in which the characters are encountered. The same authors constructed a finite state Nasta'liq text recognizer [73] for reading and recognizing each of the character shapes of Nasta'liq in the segmented text image. The method in [70] improves on the template matching method of [81,82] and relies, for recognition, on the signature values that are calculated from the input character image. Thereafter, it computes the difference with the corresponding values in the database for a match, using a confusion matrix. A two-stage recognition of printed segmented isolated characters of Naskh Urdu has been reported in [89]. First, a template matching technique is used to generate the confusion matrix of the templates pool. Thereafter, a cluster prediction technique is applied to the confusion matrix to predict three different sets of clusters using K-means clustering. The work in [90] outlines a training based offline OCR of an isolated font for 36 Urdu Naskh characters. This pattern matching technique employees the creation and subsequent matching of chain codes of the

Table 4
Comparison of machine printed numeral recognition.

Authors	Features	Classification	Database	Accuracies (%)
Pal et al. [11]	Topological, contour & water reservoir	CART Tree	Unspecified	97.8
Shamsher et al. [79]	Statistical	FFNN	Unspecified	98.3
	templates	Correlation	100	76

Table 5
Comparison of handwritten isolated character, ligature or word and numeral recognition.

Authors	Type	Features	Classification	Dataset	Accuracies
Pathan et al. [104]	Isolated character	Number & position of secondary ligatures, and moment Invariant	SVM	36,800 characters	93.59%
Ali et al. [95]	Ligature or Word	Curvature, slope & variance of stroke, Axis ratio and end point sequence	Neural Network	25 images	70%–80%
Mukhtar et al. [102]	Ligature or Word	Gradient, structural and Cavity	SVM	1600 words	70%–82%
Sagheer et al. [101]	Ligature or word	Structural and gradient	SVM	19,432 words	97.00%
Yusuf and Haider [96]	Numeral	Shape context & Bending energy	Bipartite graph matching	40	Unspecified
Sagheer et al. [30]	Numeral	Gradient	SVM	60,329	98.61%
Basu et al. [103]	Numeral	QTLR	SVM	3,000	96.2%
Haider and Yusuf [99]	Numeral	Shape context & bending energy	Bipartite Graph matching & Inter Object Distance	40	92.6%

characters. For the extracted and pre-segmented Urdu Naskh characters, the recognition system, in [83], uses the Kohonen self-organizing map (K-SOM). A total of 104 segmented character ligatures are handled by the system at about 80% accuracy rate.

A hidden Markov model (HMM) is used for the classification of segmented primitives of the ligature by calculating the DCTs as features for improving the performance of recognition [23]. The classifier for the recognition of *Noori* Nasta'liq font style was used as a HMM in [12], wherein the implementation had been done through the HMM Tool Kit (HTK). The *BBN Byblos Pushto OCR System* [49] implements a script independent OCR using fourteen-state HMMs for Pushto. The system has also been successfully tested for Arabic, English, and Chinese documents. Due to the unavailability of a Pushto corpus, the authors collected one, a corpora of 27,000 characters, from: (a) the BBC Pushto service,[6] (b) Pushto Reader [91], (c) scanning of the printed pages at 300 dpi, (d) faxed printed pages, and (e) digital faxed pages. An extension[7] of [23], extracts *Noori Nasta'liq* ligatures independent of the font size. The Splines technique is applied on the input image of the ligature, resulting in the outlines. The outlines are then scaled to control the points in the splines and the input ligatures is resized to train the OCR. The scaled outline is subsequently converted to the image form so that the system can perform recognition. The system had been evaluated on the Urdu single character ligatures and achieved a 98% accuracy rate for the manually generated data and a 96% accuracy rate for the scanned data from various books and magazines.

The system outlined in [44] trains a feed-forward, back propagation, neural network model on a pre-defined set of ligatures from Urdu *Noori* Nasta'liq script. Despite having a good accuracy, when tested on 200 character ligatures, the system is somewhat lax in performance when it comes to unknown ligatures. For the basic printed isolated/single characters or alphabet recognition, Ref. [79] adopts a Multi-Layer Perceptron (MLP) network classifier that comprises three layers, *viz.* the input layer (150 neurons), the hidden layer (250 neurons), and the output layer (16 neurons). Furthermore, the OCR operates on an input in the form of a binary image of size 10×15 pixels or 150 neurons. The 250 neurons at the hidden layer are decided on a trial and error basis, while the count of the neurons in the output layer is attributed to the 16-bit Unicode. The limitation of the system is that it cannot recognize joined or connected or compound characters of Urdu script. The segmented characters are used to train a three-layered multilayer feed forward neural network (FFNN) for the classification and recognition in [74], by using an Ariel font of size 36. However, the system does not consider diacritics characters. We suspect that it is done to reduce the error rate as the diacritics characters are responsible for the large frequency of errors at the ending character of a word. The word recognition and classification are performed, in a similar work, reported in [71]. The aforementioned is achieved by inputting the segmented characters of printed Urdu Nasta'liq to feed a neural network classifier without lexicon. The authors claim a 93.4% accuracy, but do not provide any supportive proof of procedures for evidence. Moreover, they assume the input text to be diacritic free and of fixed font size. The FFNN is employed for the classification of extracted features in [88] to get one of the 53 classes. In their method, the best training performance of the FFNN is obtained after 757 epochs. The limitation of the system is that it does not use all of the character set of Urdu and excluds the characters shown in Fig. 29. Ref. [84] proposes an approach for isolated Urdu characters, without the use of any statistical model. The work relies on what the authors call a *softconverter*, which recognizes a character from the underlying database and its width, height, and "X" value (count of the black pixels). The authors claim recognition rates of up to 100% and 97.3% for the hard matching and soft matching, respectively.

Another technique, reported in [11], passes the extracted features as an input for training to a classification and regression tree (CART). In the CART, the decision at each node of the tree is taken on the basis of the presence (or absence) of a particular feature. Although the technique performs well on isolated characters and numerals, it suffers in the case of compound characters, and diverse sizes and fonts. The RST-invariant approach of [85] employs Fourier descriptors for the recognition and classification of Urdu characters. The PCA has been applied in [31] for the recognition of four different font size images from 1,000 Pushto ligatures in the database, as discussed in Section 2.3. However, the approach affects the recognition rate due to fixed dimensions.

[6] http://www.bbc.co.uk/pashto/
[7] http://www.cslhr.nu.edu.pk/GCCS/Spring2010/papers/Quratulain.pdf

Table 6
Comparison of online isolated character, ligature or word and numeral recognition.

Authors	Type	Features	Classification	Dataset	Accuracies
Malik and Khan [106]	Isolated character	Structural	Tree Based Dictionary with rules	49 Ligatures	93%
Shehzad et al. [114]	Isolated character	Length and angel of the bounding box diagonal, distance, sine and cosine of the angle between first and last point, length of the primary stroke, angle traversed, absolute value of angle at each point	Weighted, linear classifier	Five samples of each 38 character	92.8%
Haider and Khan [116]	Isolated character	Finishing Half Plane, Initial y-trend, Character Box Slope, Half Strokes Slope, Final y-trend, Cusp, Intersection with centroid axes, Finishing Quadran, and Finishing x-trend	BPNN (Scalar Targets), BPNN (Template Targets) Correlation Classifier and PNN	85 instances of each character	87%, 89%, 92% and 95%
Sardar [21]	Isolated character	Sliding window and Hu moment	KNN	1050	97.12%
Malik and Khan [106]	Ligature or word	Structural	Tree Based Dictionary with rules	200 words of 2 characters	78%
Hussain et al. [107]	Ligature or word	Primitives	Spatio-temporal Artificial Neuron	300 ligatures	85%
Husain et al. [10]	Ligature or word	Loop, intersection, writing styles of ligatures	BPNN	18000 ligatures	93% main ligatures, 98% secondary ligatures
Razzak et al. [24]	Ligature or word	Structural and Statistical	HMM and Fuzzy	1800 Ligatures	87.6% (Nasta'liq) and 74.1% (Naskh)
Razzak et al. [113]	Ligature or word	Bio inspired features	Fuzzy Logic	Unlimited Ligatures set - both Nastaliq and Naskh writing style	86.2%
Razzak et al. [112]	Ligature or word	Fuzzy logic	HMM and Fuzzy	1800 Nasta'liq and 1000 Naskh ligatures	89.2%
			Finite State Machine	1800 Nasta'liq ligatures	89.4%
Malik et al. [106]	Numeral	Structural	Tree Based Dictionary	Not given	93%
Razzak et al. [118]	Numeral	Structural	Rules based	900 images	96.3%
Razzak et al. [119]	Numeral	Fuzzy logic	Fuzzy rule, Hybrid and HMM	900 ligatures	97.4%, 96.2% and 97.8%

The key points of the testing image are extracted using SIFT descriptors. Thereafter, the points are matched with the already extracted features reported in [86]. The analysis showed a recognition rate of up to 74%. Moreover, the approach is script independent and can be easily adapted to other cursive languages.

4. Handwritten Urdu character recognition systems

An example illustrating handwritten words, characters, sentences, and numerals is given in Fig. 30. The challenging task is that the handwritten characters may deviate from the standard ones, in shapes, and the extent of deviation may vary from a writer to a writer. With the peculiarities of cursive scripts, the challenge is further escalated and that is why the literature has fewer references to handwritten Urdu script character recognition, as compared to the printed OCR. For example, Pal et al. [93] referred an initial work reported on handwritten Urdu characters recognition by Guru et al. [94]. Another earlier effort on offline handwritten OCR for cursive scripts was outlined in [95,96] but the next significant contributions came with a gap of about four to five years [30,97]. However, recently, the pace has took off and the domain is getting increased attention of the researchers day by day (Fig. 31).

4.1. Preprocessing and segmentation approaches

Preprocessing includes tasks, such as binarization, smoothing, noise removal, background-elimination, cropping black boundaries,

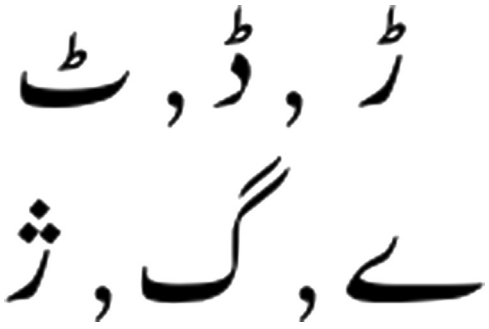

Fig. 29. Characters with secondary stroke of *chotey thoy* some other characters.

gray scale normalization, and size normalization [98]. Many approaches are employed for binarization, smoothing, and normalization of gray scale images in the literature. In [30], all of the gray scale images are binarized using the Otsu's Method [59]. The medial axis transformation is employed for thinning in [95].

In relation to the segmentation, some handwritten documents may have non-texts objects, such as stamp-seal. Therefore, these documents are to be segmented into text and non-text parts. The text parts are needed to be separated into lines—a crucial and complex task. In Urdu offline OCR for handwritten text document the aforementioned task becomes extremely difficult due to the variations in the interline spacing, inconsistent skew in the baseline, and overlapping of words (either in the same line or between two consecutive text lines). Incorrect segmentation of the text lines directly affect the rest of steps. Many techniques have been reported

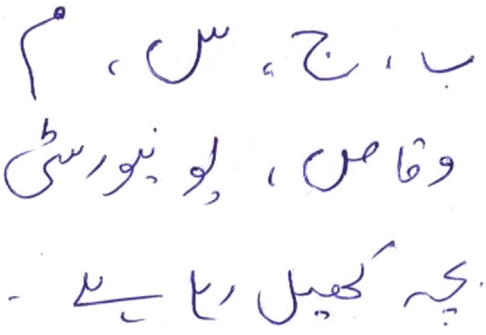

Fig. 30. Handwritten letters, words and sentences of Urdu.

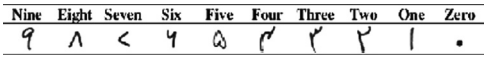

Fig. 31. Handwritten numerals of Urdu.

for the text line segmentation in the literature, which are based on the projection profile, thinning strategy, and Hough transform. In addition to announcing a sentence database, the authors in [32], proposed a horizontal projection profiling based method for text line segmentation. They first filtered the connected components using some threshold to handle the intersection issues and then computed the centroids of the connected components. If the centroid lies above the cutting line of the horizontal projection profile, then a component is assigned to the above line. Otherwise, it is assumed to belong to the line below.

The most important and crucial step is the word or character segmentation that becomes more challenging in the case of Urdu-like languages. In [95], a four-connected component method is employed for the ligature separation and an eight-connected component approach is used to ascertain the slopes and curvatures for the primitive level segmentation. Compound features from the integration of spatial information of diacritics have been used for the extraction of the principal connected components of handwritten Urdu words in [97]. The authors claim a segmentation rate of up to 92.11%.

4.2. Feature extraction approaches

On the basis of twelve elementary geometric shapes, the authors of [95] carry out the feature extraction of three languages, *viz.* Urdu, Hindi, and English. The extracted statistical features rely on the computation of curvature, slope, end-points, axes ratio, and the length variations of strokes. Shape context, which is a novel feature based object descriptor, has been applied for the feature extraction of handwritten Urdu digits [96,99]. Using the Robert's operator [100], the method in [30] extracted 400 gradient features from a normalized image for the classification of Urdu digits. The same authors applied a segmentation free approach [101] and extracted compound features (structural and gradient) using the Robert's filter. In [102], the gradient, structural, and cavity (GSC) features were extracted from the handwritten Urdu words. Basu et al. [103] employed the quad-tree based longest-run (QTLR) approach for the feature extraction from the numeric digit patterns of Urdu and Bangla language. In [104], invariant moments are utilized to extract the features for offline handwritten isolated Urdu OCR. An extension of the method [8], on printed Urdu documents, is reported in [9] that presents the same segmentation into PW's for information retrieval, in the case of handwritten Urdu words.

4.3. Recognition and classification approaches

The work reported in [104] proposes a recognition and classification system using the support vector machine (SVM). While evaluating 36,800 handwritten characters, the authors claim to have achieved an overall performance rate of about 93.59% for the offline handwritten isolated Urdu characters. Another technique [30] conducted recognition experiments on isolated Urdu digits using a SVM classifier with a radial base function (RBF) kernel function. The authors claimed an accuracy of up to 98.61% with 47,151 images in the training set and 13,178 images in the testing set. Ref. [102] claims to have presented the first handwritten Urdu words recognition system after performing some experiments on 1300 handwritten Urdu words using a SVM with about 70–82% accuracy rates. Another work [103] recognizes the Urdu and Bangla numerals on 3000 postal douments using a SVM and achieved about 96.2% of accuracy. A variation of the method [101], on Urdu word recognition and evaluation on their own database, resulted in an accuracy rate of up to 97%, according to the authors. Neural networks have been applied in [95] for the construction of offline handwritten OCR, wherein the recognizer is implemented through self-organizing feature maps (SOM's) [105]. The recognizer was trained for each stroke in Nasta'liq Urdu and tested on 25 document images. The authors claimed an accuracy rate between 70% and 80%. In [96], the weighted sum of the cost of matching shape contexts and the bending energy (BE) was used to measure the similarity between two instances. (The BE is the cost of work that it takes to transform one instance to another.) The work claimed zero percent error on a set of 28 test digits out of the 40 datasets. In [99], the extension of the work reported in [96], presented a gradual pruning approach based on the differences between the test object and the objects in the prototype set for faster processing.

5. Online Urdu character recognition systems

The mushroom growth of hand-held devices, especially the smart phones, has necessitated, among other things, the urgency to develop a reliable and efficient online character recognition system for effective human machine interaction. The earlier works in this context on Urdu are [106,107]. Among the most recent efforts is the work reported in [108], in which the authors tried to present a complete online Urdu OCR.

As it has been the practice in the last two sections, we again partition our analysis with reference to the typical OCR pipeline in the rest of this section, namely a categorization with respect to preprocessing, segmentation, feature-extraction, recognition, and classification.

5.1. Preprocessing and segmentation approaches

Traditionally, the research on online OCR's has focused on the Latin and Chinese scripts. We find only a limited literature on the cursive scripts [109]. Most of the Arabic and Urdu script online recognition systems use input of document images with the inherent assumption of no noise. The preprocessing, as well as the feature extraction, (totally) depends on whether the input is online or offline. The online recognition is a bit easier than the offline counterpart as it serves the better cause of the writing and gives more precise information on the order of the pixels and the writing physiology.

Preprocessing methods compiled from the various sources include smoothing and de-hooking [10,26], repetition removal and filtering [106], and displacement computation (4,8,16 displacements between the two points) [107]. In [24], the online preprocessing

stage encompasses stroke segmentation, interpolation, smoothing, and de-hooking. The offline preprocessing stage includes the combining of strokes and baseline detection. The Urdu baseline detection methods have not received much attention. Only the most common horizontal projection [110] has traditionally been the preference among the baseline detection methods. Baseline extraction in [26] for Urdu online handwritten Nasta'liq and Naskh amalgamates two techniques. The horizontal projection is used for the detection of primary baseline after the separation of secondary strokes or diacritics. Thereafter, the features of each of the ligature with the additional knowledge of the previous words and primary baseline are used for the estimation of local baseline. The system is claimed to have an accuracy rate of about 80.3% for the Nasta'liq and 91.7% for the Naskh font. In [111], the skeletonization analysis is performed using linear regression. This is followed by the mixing of offline and online information to investigate the spatial morphology of the Urdu scripts' writing for baseline detection. Ref. [24] detects the baseline by calculating the minimum enclosing rectangle and drawing a vertical projection. The technique of [112] performs de-hooking, smoothing, interpolation, estimation of slant, baseline detection, and skew correction in the preprocessing steps for handwritten online Urdu script character recognition to improve and normalize the raw input for recognition system. The preprocessing step in two other works [113,108] employ a fuzzy and context knowledge based biological technique for local baseline estimation, stroke mapping, and slant correction. In this regard, the angle of the current word is computed with the additional knowledge of the previous word angle for the detection of baseline.

5.2. Feature extraction approaches

The segmentation free approach of [10], relies on 20 different structural features for the recognition of 850 single character, two-character and three-character ligatures, to achieve a recognition of 18,000 common words of Urdu dictionary. The effort depicted in [114], defines nine features of the primary strokes and four features of the secondary strokes for an online recognition system with reference to the isolated handwritten Urdu characters. For applying spatial/temporal neural networks during their recognition and classification phase, the authors in [107] extracted the primitives as features for Urdu handwriting.

Using fuzzy logic, Ref. [24] extracts 32 directional and structural features of the Arabic script languages of Nasta'liq and Naskh style. Moreover, the authors have used fuzzy logic to reduce the HMM dataset, based on the starting and ending shape of the character. The work presents a hybrid approach using fuzzy rules for preprocessing of the features to normalize the input, and at the post processing step to improve the results. The preprocessing, feature extraction, feature purification and clustering are performed through the fuzzy rules constituting the inner shell; whereas, the HMM is used as a main classifier to recognize the ligature. In related attempts [112,113], fuzzy logic rules were used to extract the strokes and then combine the strokes to form ligatures using some linguistic rules. A bio-inspired feature extraction procedure, based on the human visual system (HVS), was reported in [108]. The methodology relies on two levels, namely level 3 and level 4. The level 3 is responsible for dividing the strokes into sub-patterns and extracting the unique sub-patterns. The level 4 is responsible for complex feature extraction and subsequent fusion. Once the levels are achieved, the authors perform the feature level fusion to form more complex patterns. Moreover, postprocessing on the extracted features is performed to remove the unnecessary features. This postprocessing step is based on the language properties. In 2012, Razzak and his co-authors [112] segmented the ligatures into strokes or sub patterns and extracted the strokes, such as vertical lines, end point, and junction points. These were combined to form more complex sub-patterns, such as long vertical up, long vertical down, small vertical up, cusp, loop, and hedge.

5.3. Recognition and classification approaches

Using a tree based dictionary, with a rule based slant analysis, and the conversion approach, the work reported in [106] implements an online recognition system for the isolated Urdu characters, numbers, and 200 words of two characters. The work claims an accuracy rate of 93% for the isolated characters and numbers, and 78% for the two character words. In [107], the authors develop a spatio-temporal artificial neuron (STAN) to classify the spatio-temporal patterns (STPs) for the recognition and classification of Urdu handwriting on a digital tablet. This neuron is embedded in ordinary artificial neural network (ANN) to build a spatio-temporal neural network (STNN). Another work [10] uses a back-Propagation Neural Network (BPNN) as a classifier and claims an accuracy of 93% for base ligatures and 98% for the secondary ligatures that were proposed in [115], which had shown a 92.8% accuracy rate for the native Urdu writers and 73% for non-native Urdu writers. According to authors, some similar characters, such as *Tday* and *Daal* were incorrectly classified, despite having different primary strokes. This implies that the work only used the initial character of cursive script for recognition. An online isolated single-stroke handwritten Urdu character recognition system [116] relies on the extraction of seven novel features for a classification involving the following:

- BPNN using template vectors as targets,
- BPNN based classifier using scalar targets,
- probabilistic neural network (PNN), and
- correlation based classifier.

The authors tested the different classifiers on 85 instances of character set taken from 35 individuals of different age groups and achieved a recognition rate of 87%, 89%, 92% and 95% for BPNN (Scalar Targets), BPNN (Template Targets), Correlation Classifier, and PNN, respectively. An extension of the work [117] considered multiple classifiers, such as correlation based classifier, back propagation neural network classifier, and probabilistic neural network classifier for the development of online isolated handwritten Urdu character recognition system for multi stroke. The system was tested for characters with two to four strokes on a database of 110 instances of handwritten Urdu characters from 40 individuals of different age groups. The authors found that the best classifier was the probabilistic neural network that achieved a recognition rate from 94% to 98% depending on number of strokes.

The online numerals recognition system in [118] adopts a fuzzy rule based approach in an unconstrained environment and has been shown to achieve a recognition rate of up to 96.3%. A related online multifont numeral recognition system [119] relies on fuzzy rules, HMM, and a melange of the two in the case of Arabic and Urdu. The work dwells on the similarities and dissimilarities of the two languages with respect to both the online and offline OCR. The ensued results were considerably accurate with all of the three approaches. The proposed system evaluated on 900 samples that were taken from 30 trained users, provided an accuracy of 97.4%, 96.2%, and 97.8% by with the fuzzy rule, the HMM, and the Hybrid approach, respectively. According to authors, the proposed system, still has many problems due to the complexities of the two languages. An extension of the work is proposed in [24], which applies the HMM and the fuzzy logic for online recognition of 1800 ligatures for the Nasta'liq and Naskh fonts. The dataset is trained for each stroke by the HHM, while further classification is done through the application of the fuzzy logic rules to the starting and ending shape of the characters. The proposed hybrid approach

provides a suitable result for a large variation in handwritten strokes, and reduces the comparison and computation. Finally, the mapping of secondary strokes and investigated primary strokes are performed using the fuzzy logic for the recognition of valid ligature. The authors reported a 87.6% accuracy for the Nasta'liq font and a 74.1% accuracy rate for the Nasakh font. Ref. [112] proposes a yet another fuzzy logic rules based method, which is supplemented by the linguistic rules to recognize characters from the strokes. A fuzzy triangular member function is used for mapping the diacritical marks. The mapping of secondary strokes is performed by using the linguistics rules, associated dots, recognized sub-unit with associated units, and diacritical positional information based on projection. The proposed approach claimed a recognition rate of 86.2% for the Nasta'liq style. The method described in [113], relies on an online character recognition system that was presented in [120], for the Nasta'liq style of 1800 ligatures and Naskh style of 1000 ligatures. The authors reported a recognition rate of upto 89.2% A font-independent online and offline Urdu OCR proposal [21] for isolated characters, uses the K-Nearest Neighbor (KNN) algorithm based on five features. A sliding window approach was used for computing the first four features and the Hu Moment approach was used for the fifth feature.

6. Conclusions and future recommendations

Due to a number of challenges, a reliable Urdu script OCR is still a far cry and a lot is yet to be done in this connection. Of particular importance is the peculiar nature of the Nasta'liq style which makes the task even more challenging. With the images of the printed text, the situation is far more encouraging as is evident by the number of efforts cited above. However, researchers have not been that enthusiastic as far as handwritten text is concerned, whether offline or online. The research has been mostly focused on the isolated or ligature based recognition for Urdu-like scripts. Moreover, the works targeting Urdu line text are scarce. Furthermore, Urdu page recognition is also a neglected area. Current limitations include restrictive lexicons, writing style restrictions, and lesser accuracy with Nasta'liq. Overall, the situation with Urdu language is not as bad as with Pushto and Sindhi where the research is at its rudimentary stage, prima facie at least.

From the perspective of the future work/directions, there is a need to develop algorithms that can incorporate unlimited or large lexicon; the ultimate being the capability of multilingual cursive script character recognition. Section 2.1 briefly described the properties of Urdu-like script languages. These languages share a common character set and writing style. More sophisticated algorithms are required for the baseline estimation and ligature segmentation that may eventually prove helpful in tackling the restricted lexicon problem. Based on the reassembling of characters, according to their basic shapes, the concept of what is called the Ghost Character Theory, seems to have a lot of research potential in the near future. A proponent of this theory states: "there are some problems in Urdu ASCI code plate, when I analyzed that some symbols and all the language of Pakistan is possible from one code plate and one font. Then I proposed the idea of Ghost Character" [121]. According to the approach all of the Arabic script based language can be written with only 44 ghost characters. Ghost character consists of 22 basic shapes called *Kashti* (Fig. 32) and diacritical marks (Fig. 33) [122].

There is no multilingual OCR available, while there is very high similarity between Arabic script languages. According to the Ghost character theory the glyphs (ghost character) are same for all Urdu script languages. The character set is different because of diacritical marks and their placement. Moreover, there are different writing styles followed by different languages; Nastaliq and Naskh

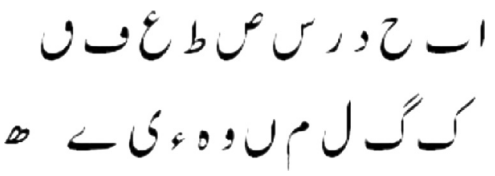

Fig. 32. Ghost character for Urdu script.

Fig. 33. Placement of diacritical marks.

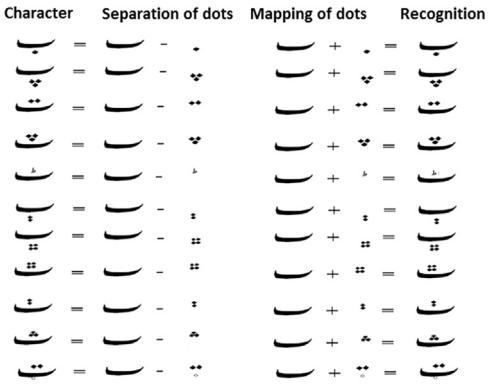

Fig. 34. Multilingual character recognition based on ghost character theory.

being the most common. A multilingual character recognition system can be possible by following the ghost character recognition theory—an effort is outlined in [123]. The theory may not only help us to build a multilingual OCR but also reduce the lexicon. In a nutshell, efforts are needed to focus on the across the board, language independent, recognition of the two commonly used Nasta'liq and Naskh scripts wherein the ligature or word formation can be done using the lexicon for specific language. Fig. 34 shows the proposed process for character recognition on the basis of the ghost character theory. A multilingual character recognition system is graphically outlined in Fig. 35.

Subsequent to the lexicon reduction, the task of character recognition begins, possibly through the ghost character theory. For application specific aspects, such as city names, a holistic/ligature based approach is preferred. While for the character recognition systems, segmentation based approach may be a viable a solution for unlimited lexicons. Most of the techniques based on the ligatures, may not be applicable directly to unconstrained and unlimited Urdu Nasta'liq text. As stated elsewhere, the segmentation prospects are far more complex in Nasta'liq for the fact that a character may assume as many as 32 shapes. Therefore, a more concerted research effort is needed to streamline the segmentation in the case of unconstrained Urdu text recognition. For instance, the HMM and neural networks have been quite successful in segmenting Naskh. However, it remains to be seen how accurate these are with Nasta'liq, and in the case of poor showing, what modifications are needed for improvements? In the domain of feature extraction, structural features are much more important for an Urdu text recognition system, and may require a deeper insight into the word morphology. With the post processing step, an extensive spell checker and lexicon may significantly enhance the accuracy and performance. With the database part, an open source representative benchmark repository of text images is the need of the hour. The reasonable accuracy, in

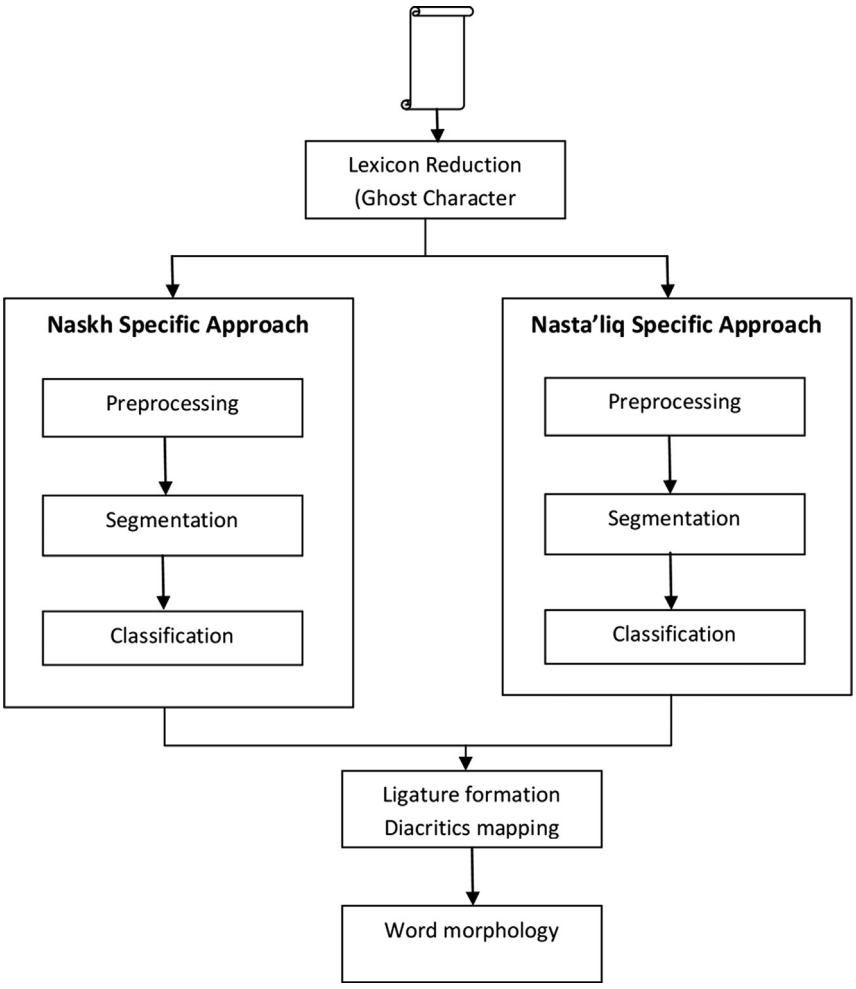

Fig. 35. Multilanguage Urdu script like character recognition system.

recognizing the Naskh style (Arabic/Farsi), may be attributed to the availability of databases, such as IFN/ENIT.[8] The same argument can be cited as the *raison d'être* for the maturity of the Latin script OCRs due to the databases, such as MNIST[9] and CEDAR.[10] In essence, the challenges with Nasta'liq script are manifold and dedicated collaborative efforts are needed on the part of the research community to cross the Rubicon.

Conflict of interest statement

None declared.

References

[1] S. Hussain, M. Afzal, Urdu computing standards: development of Urdu Zabta Takhti (UZT) 1.01, in: Proceedings of the 5th International Multitopic IEEE Conference (INMIC'01), 2001, pp. 216–222.
[2] V. Govindan, A. Shivaprasad, Character recognition – a review, Pattern Recognition 23 (7) (1990) 671–683.
[3] S.A. Sattar, S. Shah, Character recognition of Arabic script languages, in: Proceedings of the International Conference on Computer and Information Technology (ICCIT'12), 2012.
[4] N. Fareen, M.A. Khan, A. Durrani, Survey of urdu OCR: an offline approach, in: Proceedings of the Conference on Language & Technology 2012 (CLT12), University of Engineering & Technology (UET), Lahore, Pakistan, 2012, pp. 67–72.
[5] L.M. Lorigo, V. Govindaraju, Offline Arabic handwriting recognition: a survey, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (5) (2006) 712–724.
[6] N. Tagougui, M. Kherallah, A.M. Alimi, Online Arabic handwriting recognition: a survey, International Journal on Document Analysis and Recognition (2012) 1–18.
[7] T. Rahman, Language policy and localization in Pakistan: proposal for a paradigmatic shift, in: Proceedings of the SCALLA Conference on Computational Linguistics, vol. 99, 2004, pp. 1–18.
[8] A. Abidi, I. Siddiqi, K. Khurshid, Towards searchable digital urdu libraries – a word spotting based retrieval approach, in: Proceedings of the International Conference on Document Analysis and Recognition (ICDAR'11), 2011, pp. 1344–1348.
[9] A. Abidi, A. Jamil, I. Siddiqi, K. Khurshid, Word spotting based retrieval of urdu handwritten documents, in: Proceedings of the International Conference on Frontiers in Handwriting Recognition (ICFHR'12), 2012, pp. 331–336.
[10] S.A. Husain, A. Sajjad, F. Anwar, Online urdu character recognition system, in: Proceedings of the IAPR Conference on Machine Vision Applications (MVA'07), 2007, pp. 98–101.
[11] U. Pal, A. Sarkar, Recognition of printed Urdu script, in: Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003), 2003, pp. 1183–1187.
[12] S.T. Javed, S. Hussain, A. Maqbool, S. Asloob, S. Jamil, H. Moin, Segmentation Free Nastalique urdu OCR, World Academy of Science, Engineering and Technology.
[13] D.A. Satti, K. Saleem, Complexities and implementation challenges in offline urdu Nastaliq OCR, in: Proceedings of the Conference on Language & Technology 2012 (CLT12), University of Engineering & Technology (UET), Lahore, Pakistan, 2012, pp. 85–91.
[14] S. Hussain, ⟨www.LICT4D.asia/fonts/Urdu_Nasta'leeq⟩, in: Proceedings of the 12th AMIC Annual Conference on e-Worlds: governments, Business and Civil Society, Asian Media Information Center, Singapore, 2003.

[8] http://www.ifnenit.com/
[9] http://yann.lecun.com/exdb/mnist/
[10] http://www.cedar.buffalo.edu/Databases/

[15] S.A. Sattar, S. Haque, M.K. Pathan, Nastaliq optical character recognition, in Proceedings of the 46th Annual Southeast Regional Conference, ACM, 2008, pp. 329–331.
[16] A. Wali, A. Gulzar, A. Zia, M.A. Ghazali, M.I. Rafiq, M.S. Niaz, S. Hussain, S. Bashir, Features for Noori Nastaleeq, Akhbar-e-Urdu, A Journal of National Language Authority, Islamabad, Pakistan (2002) 303–308.
[17] M.I. Rafiq, A. Wali, M.A. Ghazali, S. Bashir, S. Hussain, A. Zia, A. Gulzar, M. S. Niaz, Contextual shape analysis of nastaliq, Akhbar-e-Urdu, A Journal of National Language Authority, Islamabad, Pakistan (2002) 288–302.
[18] M.G.A. Malik, C. Boitet, P. Bhattacharyya, Analysis of Noori Nasta'leeq for Major Pakistani Languages, in: Proceedings of the 2nd Workshop on Spoken Language Technologies for Under-resourced Languages (SLTU'2010), Penang, Malaysia, 2010, pp. 95–103.
[19] S. Hussain, Complexity of Asian writing systems: a case study of Nafees Nasta'leeq for urdu, in: Proceedings of the 12th AMIC Annual Conference on e-Worlds: Governments, Business and Civil Society, Asian Media Information Center, Singapore, 2003.
[20] A. Wali, S. Hussain, Context Sensitive Shape-Substitution in Nastaliq Writing System: Analysis and Formulation, Springer (2007) 2007; 53–58, Innovations and Advanced Techniques in Computer and Information Sciences and Engineering.
[21] S. Sardar, A. Wahab, Optical character recognition system for Urdu: online and offline OCR irrespective of fonts, in: Proceedings of the International Conference on Information and Emerging Technologies (ICIET), Karachi, Pakistan, 2010, pp. 167–171.
[22] M. Asad, A.S. Butt, S. Chaudhry, S. Hussain, Rule-based expert system for urdu Nastaleeq justification, in: Proceedings of the 8th International Multitopic IEEE Conference (INMIC'04), 2004, pp. 591–596.
[23] S.T. Javed, Investigation into a segmentation based OCR for the Nastaleeq writing system (Master's thesis). National University of Computer & Emerging Sciences, Lahore, Pakistan, 2007.
[24] M.I. Razzak, F. Anwar, S.A. Husain, A. Belaïd, M. Sher, HMM and fuzzy logic: a hybrid approach for online urdu script-based languages' character recognition, Knowledge Based Systems 23 (8) (2010) 914–923.
[25] F. Shafait, D. Keysers, T.M. Breuel, Layout analysis of Urdu document images, in: Proceedings of the 10th International Multitopic IEEE Conference (INMIC'06), 2006, pp. 293–298.
[26] M.I. Razzak, M. Sher, S.A. Hussain, Locally baseline detection for online arabic script based languages character recognition, International Journal of the Physical Sciences 5 (7) (2010) 955–959.
[27] A. Gulzar, S. Ur-Rahman, Nastaleeq: a challenge accepted by omega, in: Proceedings of the 17th European TEX Conference (TUGboat), vol. 29, no. 1, 2007, pp. 89–94.
[28] S.S. Bukhari, F. Shafait, T.M. Breuel, Layout analysis of Arabic script documents, in: Computer Analysis of Images and Patterns, ser. Lecture Notes in Computer Science, vol. 5702. Springer, 2012, pp. 35–53.
[29] G.S. Lehal, Choice of recognizable units for urdu OCR, in: Proceedings of the Workshop on Document Analysis and Recognition (DAR'12). ACM, New York, NY, USA, 2012, pp. 79–85.
[30] M.W. Sagheer, C.L. He, N. Nobile, C.Y. Suen, A new large Urdu database for off-Line handwriting recognition 5716 (2009) 538–546.
[31] M. Wahab, H. Amin, F. Ahmed, Shape analysis of pashto script and creation of image database for OCR, in: Proceedings of the International Conference on Emerging Technologies (ICET'09), 2009, pp. 287–290.
[32] A. Raza, I. Siddiqi, A. Abidi, F. Arif, An unconstrained benchmark urdu handwritten sentence database with automatic line segmentation, in: Proceedings of the International Conference on Frontiers in Handwriting Recognition (ICFHR'12), 2012, pp. 491–496.
[33] B. Al-Badr, S.A. Mahmoud, Survey and bibliography of arabic optical text recognition, Signal Processing 41 (1) (1995) 49–77.
[34] S.N. Nawaz, M. Sarfraz, A. Zidouri, W.G. Al-Khatib, An Approach to offline Arabic character recognition using neural networks, in: Proceedings of the 10th International Conference on Electronics, Circuits and Systems (ICECS'03), vol. 3, 2003, pp. 1328–1331.
[35] A.M. Al-Shatnawi, F.H. Al-Zawaideh, S. Al-Salameh, K. Omar, Offline arabic text recognition – an overview, World of Computer Science and Information Technology (WCSIT) 1 (5) (2011) 184–192.
[36] S.T. Javed, S. Hussain, Improving Nastalique-specific pre-recognition process for Urdu OCR, in: Proceedings of the 13th International Multitopic IEEE Conference (INMIC'09), 2009, pp. 1–6.
[37] F. Farooq, V. Govindaraju, M. Perrone, Pre-processing methods for handwritten Arabic documents, in: Proceedings of the 8th International Conference on Document Analysis and Recognition (ICDAR'05). IEEE Computer Society, Washington, DC, USA, 2005, pp. 267–271.
[38] H. Al-Rashaideh, Preprocessing phase for arabic word handwritten recognition, Information Transmissions in Computer Networks 6 (1) (2006) 11–19.
[39] A.M. Zeki, The segmentation problem in arabic character recognition the state of the art, in: Proceedings of the 1st International Conference on Information and Communication Technologies (ICICT'05), 2005, pp. 11–26.
[40] R. Safabakhsh, P. Adibi, Nastaaligh handwritten word recognition using a continuous-density variable-duration HMM, The Arabian Journal for Science and Engineering 30 (1B) (2005) 95–118.
[41] M.S.M. El-Mahallawy, A large scale HMM-based Omni font-written OCR system for cursive scripts (Ph.D. dissertation), Faculty of Engineering, Cairo University Giza, Egypt, 2008.
[42] D.B. Megherbi, S.M. Lodhi, A.J. Boulenouar, Fuzzy-logic-model-based technique with application to Urdu character recognition, Proceedings of the SPIE Applications of Artificial Neural Networks in Image Processing V 3962 (2000) 13–24.
[43] Z.A. Shah, Ligature based optical character recognition of Urdu-Nastaleeq font, in: Proceedings of the 6th International Multitopic IEEE Conference (INMIC'02), 2002, pp. 25–25.
[44] S.A. Husain, A multi-tier holistic approach for urdu Nastaliq recognition, in: Proceedings of the 6th International Multitopic IEEE Conference (INMIC'02), 2002, pp. 528–532.
[45] U.R. Ahmed, Design and implementation report of optical Urdu text recognition (Master's thesis). COMSATS Institute of Information Technology, Lahore, Pakistan, 2004.
[46] A. Muaz, Urdu optical character recognition system (Master's thesis). National University of Computer & Emerging Sciences Lahore, Pakistan, 2010.
[47] U. Iftikhar, Recognition of Urdu Ligatures (Master's thesis). VIBOT Consortium and German Research Center for Artificial Intelligence (DFKI), 2011.
[48] S.A. Sattar, A technique for the design and implementation of an OCR for printed Nastalique text (Ph.D. dissertation). NED University of Engineering & Technology, Karachi, Pakistan, 2009.
[49] M. Decerbo, E. MacRostie, P. Natarajan, The BBN Byblos Pashto OCR system, in: Proceedings of the 1st ACM Workshop on Hardcopy Document Processing (HDP '04). ACM, New York, NY, USA, 2004, pp. 29–32.
[50] N.A. Shaikh, G.A. Mallah, Z.A. Shaikh, Character segmentation of Sindhi, an Arabic style scripting language, using height profile vector, Australian Journal of Basic and Applied Sciences 3 (4) (2009) 4160–4169.
[51] R.J. Ramteke, I.K. Pathan, Noise reduction in urdu document image–spatial and frequency domain approaches, in: Proceedings of the 4th International Conference on Signal and Image Processing 2012 (ICSIP'12), ser. Lecture Notes in Electrical Engineering. Springer India, 2013, vol. 222, pp. 443–452.
[52] D.S. Le, G.R. Thoma, H. Wechsler, Automated page orientation and skew angle detection for binary document images, Pattern Recognition 27 (10) (1994) 1325–1344.
[53] R.J. Ramteke, K.P. Imran, S.C. Mehrotra, Skew angle estimation of urdu document images: a moments based approach, International Journal of Machine Learning and Computing 1 (1) (2011) 7–12.
[54] S.F. Rashid, S.S. Bukhari, F. Shafait, T.M. Breuel, A discriminative learning approach for orientation detection of urdu document images, in: Proceedings of the 13th International Multitopic IEEE Conference (INMIC'09), 2009, pp. 1–5.
[55] T.M. Breuel, High performance document layout analysis, in: Proceedings of the Symposium on Document Image Understanding Technology (SDIUT '03), 2003, pp. 209–218.
[56] S.S. Bukhari, F. Shafait, T.M. Breuel, High performance layout analysis of Arabic and Urdu document images, in: Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR'11), 2011, pp. 1275–1279.
[57] S.S. Bukhari, F. Shafait, T.M. Breuel, Improved document image segmentation algorithm using multiresolution morphology, in: Proceedings of the IS&T/SPIE Electronic Imaging Symposium – Document Recognition and Retrieval XVIII. the International Society for Optics and Photonics, 2011, pp. 78 740D.
[58] D.S. Bloomberg, Multiresolution morphological approach to document image analysis, in: Proceedings of the International Conference on Document Analysis and Recognition (ICDAR).
[59] N. Otsu, A threshold selection method from gray-level histograms, IEEE Transactions on Systems, Man and Cybernetics 9 (1) (1979) 62–66.
[60] J. Sauvola, M. Pietikäinen, Adaptive document image binarization, Pattern Recognition 33 (2) (2000) 225–236.
[61] J.M. Geusebroek, A.W.M. Smeulders, J. van de Weijer, Fast anisotropic gauss filtering, IEEE Transactions on Image Processing 12 (8) (2003) 938–943.
[62] C.H. Lampert, O. Wirjadi, An optimal nonorthogonal separation of the anisotropic Gaussian convolution filter, IEEE Transactions on Image Processing 15 (11) (2006) 3501–3513.
[63] S.S. Bukhari, F. Shafait, T.M. Breuel, Ridges based curled textline region detection from grayscale camera-captured document images, in: Computer Analysis of Images and Patterns, ser. Lecture Notes in Computer Science, vol. 5702. Springer, Berlin Heidelberg, 2009, pp. 173–180.
[64] S.S. Bukhari, F. Shafait, T.M. Breuel, Script-independent handwritten textlines segmentation using active contours, in: Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR'09), 2009, pp. 446–450.
[65] M. Riley, Beyond quasi-stationarity: designing time-frequency representations for speech signals, in: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'87), vol. 12, 1987, pp. 657–660.
[66] M.D. Riley, Time-frequency representation for speech signals (Ph.D. dissertation), Massachusetts Institute of Technology - Artificial Intelligence Lab, 1987.
[67] T. Breuel, Two geometric algorithms for layout analysis, in: Document Analysis Systems V, ser. Lecture Notes in Computer Science, vol. 2423. Springer, Berlin, Heidelberg, 2002, pp. 687–692.
[68] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, MIT Press, 2009.
[69] M. Ijaz, S. Hussain, Corpus based Urdu Lexicon development, in: Proceedings of the Conference on Language Technology, 2007.

[70] S.M. Azam, Z.A. Mansoor, M. Sharif, On fast recognition of isolated characters by constructing character signature database, in: Proceedings of the International Conference on Emerging Technologies (ICET'06), 2006, pp. 568–575.
[71] Z. Ahmad, J.K. Orakzai, I. Shamsher, A. Adnan, Urdu Nastaleeq optical character recognition, in: Proceedings of the World Academy of Science, Engineering and Technology, vol. 26, 2007.
[72] S.A. Sattar, S. Haque, M.K. Pathan, Q. Gee, Implementation challenges for nastaliq character recognition, in: Wireless Networks, Information Processing and Systems, ser. Communications in Computer and Information Science, vol. 20. Springer, Berlin, Heidelberg, 2009, pp. 279–285.
[73] S.A. Sattar, S. ul Haq, M.K. Pathan, A finite state model for urdu nastalique optical character recognition, International Journal of Computer Science and Network Security (IJCSNS) 9 (9) (2009).
[74] Z. Ahmad, J.K. Orakzai, I. Shamsher, Urdu compound character recognition using feed forward neural networks, in: Proceedings of the 2nd International Conference on Computer Science and Information Technology (ICCSIT'09), 2009, pp. 457–462.
[75] H. Malik, M.A. Fahiem, Segmentation of printed urdu scripts using structural features, in: Proceedings of the 2nd International Conference in Visualisation (VIZ'09), 2009, pp. 191–195.
[76] M. Akram, S. Hussain, Word segmentation for urdu OCR system, in: Proceedings of the 8th Workshop on Asian Language Resources. Asian Federation for Natural Language Processing, Beijing, China, 2010, pp. 87–93.
[77] M.A.U. Rehman, A new scale invariant optimized chain code for nastaliq character representation, in: Proceedings of the 2nd International Conference on Computer Modeling and Simulation (ICCMS'10), vol. 4, 2010, pp. 400–403.
[78] J.A. Mahar, G.Q. Memon, S.H. Danwar, Algorithms for Sindhi word segmentation using Lexicon-driven approach, International Journal of Academic Research 3 (3) (2011) 28.
[79] I. Shamsher, Z. Ahmad, J.K. Orakzai, A. Adnan, OCR for printed urdu script using feed forward neural network, Proceedings of the World Academy of Science, Engineering and Technology 23 (2007) 172–175.
[80] R.G. Casey, G. Nagy, Recursive segmentation and classification of composite character patterns, in: Proceedings of the 6th International Conference on Pattern Recognition, vol. 2, 1982, pp. 1023–1026.
[81] F. Hussain, J. Cowell, Extracting features from arabic characters, in: Proceedings of the 2nd International Conference on Computer Graphics and Imaging (CGIM'01), Honolulu, Hawaii, USA, 2001, pp. 201–206.
[82] J. Cowell, F. Hussain, A fast recognition system for isolated arabic characters, in: Proceedings of the 6th International Conference on Information Visualisation, London, UK, 2002, pp. 650–654.
[83] S.A. Hussain, S. Zaman, M. Ayub, A self organizing map based urdu Nasakh character recognition, in: Proceedings of the International Conference on Emerging Technologies (ICET'09), Islamabad, Pakistan, 2009, pp. 267–273.
[84] J. Tariq, U. Nauman, M.U. Naru, Softconverter: a novel approach to construct OCR for printed urdu isolated characters, in: Proceedings of the 2nd International Conference on Computer Engineering and Technology (ICCET'10), vol. 3, Singapore, 2010, pp. V3–495–V3–498.
[85] S.M. Lodhi, M.A. Matin, Urdu character recognition using Fourier descriptors for optical networks, in: Proceedings of the Photonic Devices and Algorithms for Computing VII, vol. SPIE 5907, 2005, pp. 59 070O–13. [Online]. Available: + http://dx.doi.org/10.1117/12.612650.
[86] R. Ahmad, S.H. Amin, M.A.U. Khan, Scale and rotation invariant recognition of cursive Pashto script using SIFT Features, in: Proceedings of the 6th International Conference on Emerging Technologies (ICET'10), Islamabad, Pakistan, 2010, pp. 299–303.
[87] D.G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.
[88] S. Zaman, W. Slany, F. Sahito, Recognition of segmented Arabic/Urdu characters using pixel values as their features, in: Proceedings of the 1st International Conference on Computer and Information Technology (ICCIT'2012), 2012, pp. 507–512.
[89] T.K. Khan, S.M. Azam, S. Mohsin, An improvement over template matching using K-means algorithm for printed cursive script recognition, in: Proc. 4th IASTED International Conference on Signal Processing, Pattern Recognition, and Applications, ser. SPPRA '07, ACTA Press, Anaheim, CA, USA, 2007, pp. 209–214. [Online]. Available: ⟨http://dl.acm.org/citation.cfm⟩, id=1710523.1710563.
[90] T. Nawaz, S.A.H.S. Naqvi, H. ur Rehman, A. Faiz, Optical character recognition system for Urdu (Naskh font) using pattern matching technique, International Journal of Image Processing (IJIP) 3 (3) (2009) 92–104.
[91] H. Tegey, B. Robson, Pashto Reader Originals, 1992, sponsored by the Office of International Education (ED), Washington, DC.
[92] N. Sabbour, F. Shafait, A segmentation-free approach to arabic and urdu OCR, in: Proceedings of the SPIE 8658, Document Recognition and Retrieval XX. International Society for Optics and Photonics, 2013, pp. 86 580N–86 580N–12.
[93] U. Pal, R. Jayadevan, N. Sharma, Handwriting recognition in indian regional scripts: a survey of offline techniques, ACM Transactions on Asian Language Information Processing (TALIP) 11 (1) (2012) 1:1–1:35.
[94] D.S. Guru, S.K. Ahmed, K. Irfan, An attempt towards recognition of handwritten urdu characters: a decision tree approach, in: Proceedings of the National Conference on Computers and Information Technology (NCCIT'01), 2001, pp. 75–83.
[95] A. Ali, M. Ahmad, N. Rafiq, J. Akber, U. Ahmad, Akmal, Language independent optical character recognition for hand written text, in: Proceedings of the 8th International Multitopic IEEE Conference (INMIC'04), 2004, pp. 79–84.
[96] M. Yusuf, T. Haider, Recognition of handwritten urdu digits using shape context, in: Proceedings of the 8th International Multitopic IEEE Conference (INMIC'04), 2004, pp. 569–572.
[97] M.W. Sagheer, N. Nobile, C.L. Hev, C.Y. Suen, A novel handwritten urdu word spotting based on connected components analysis, in: 20th International Conference on Pattern Recognition (ICPR'10)., 2010, pp. 2013–2016.
[98] C.L. Liu, K. Nakashima, H. Sako, H. Fujisawa, Handwritten digit recognition: investigation of normalization and feature extraction techniques, Pattern Recognition 37 (2) (2004) 265–279.
[99] T. Haider, M. Yusuf, Accelerated recognition of handwritten urdu digits using shape context based gradual pruning, in: Proceedings of the International Conference on Intelligent and Advanced Systems (ICIAS'07), 2007, pp. 601–604.
[100] M. Shi, Y. Fujisawa, T. Wakabayashi, F. Kimura, Handwritten numeral recognition using gradient and curvature of gray scale image, Pattern Recognition 35 (10) (2002) 2051–2059.
[101] M.W. Sagheer, C.L. He, N. Nobile, C.Y. Suen, Holistic urdu handwritten word recognition using support vector machine, in: Proceedings of the 20th International Conference on Pattern Recognition (ICPR'10), 2010, pp. 1900–1903.
[102] O. Mukhtar, S. Setlur, V. Govindaraju, Experiments on urdu text recognition, in: Guide to OCR for Indic Scripts, ser. Advances in Pattern Recognition. Springer, London, 2010, pp. 163–171.
[103] S. Basu, N. Das, R. Sarkar, M. Kundu, M. Nasipuri, D.K. Basu, A novel framework for automatic sorting of postal documents with multi-script address blocks, Pattern Recognition 43 (10) (2010) 3507–3521.
[104] I.K. Pathan, A.A. Ali, Recognition of offline handwritten isolated urdu character, Advances in Computational Research 4 (1) (2012) 117–121.
[105] T. Kohonen, Self-Organizing Maps, 3rd ed., vol. 30, ser. Springer Series in Information Sciences. Springer, Heidelberg, 2001.
[106] S. Malik, S. A. Khan, Urdu online handwriting recognition, in: Proceedings of the IEEE Symposium on Emerging Technologies, 2005, pp. 27–31.
[107] M. Hussain, M.N. Khan, Online urdu ligature recognition using spatial temporal neural processing, in: Proceedings of the 9th International Multitopic IEEE Conference (INMIC'05), 2005, pp. 1–5.
[108] M.I. Razzak, Online urdu character recognition in unconstrained environment (Ph.D. dissertation), International Islamic University, Islamabad, Pakistan, 2011.
[109] F. Bouchiareb, M. Bedda, S. Ouchetai, New preprocessing methods for handwritten Arabic word, Asian Journal of Information Technology 5 (6) (2006) 609–613.
[110] B. Parhami, M. Taraghi, Automatic recognition of printed farsi texts, Pattern Recognition 14 (1–6) (1981) 395–403.
[111] M.I. Razzak, S.A. Hussain, M. Sher, Z.S. Khan, Combining offline and online preprocessing for online urdu character recognition, in: Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS'09), vol. I, Hong Kong, 2009, pp. 912–915.
[112] M.I. Razzak, S.A. Husain, A.A. Mirza, A. Belaïd, Fuzzy based preprocessing using fusion of online and offline trait for online urdu script based languages character recognition, International Journal of Innovative Computing, Information and Control 8 (5(A)) (2012) 3149–3161.
[113] M.I. Razzak, S.A. Husain, A.A. Mirza, M.K. Khan, Bio-inspired multilayered and multilanguage Arabic script character recognition system, International Journal of Innovative Computing, Information and Control 8 (4) (2012) 2681–2691.
[114] N. Shahzad, B. Paulson, T. Hammond, Urdu Qaeda: recognition system for isolated urdu characters, in: Proceedings of the IUI Workshop on Sketch Recognition, Sanibel Island, Florida, 2009.
[115] D. Rubine, Specifying gestures by example, in: Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '91), New York, USA, 1991, pp. 329–337.
[116] I. Haider, K.U. Khan, Online recognition of ssingle stroke handwritten urdu characters, in: Proceedings of the 13th International Multitopic IEEE Conference (INMIC'09), 2009, pp. 1–6.
[117] K.U. Khan, I. Haider, Online recognition of multi-stroke handwritten urdu characters, in: Proceedings of the International Conference on Image Analysis and Signal Processing (IASP'10), Xiamen, China, 2010, pp. 284–290.
[118] M.I. Razzak, A. Belaïd, S.A. Hussain, Effect of ghost character theory on arabic script based languages character recognition, in: Proceedings of the WASE Global Conference on Image Processing and Analysis (GCIA'09), Taiwan, China, 2009.
[119] M.I. Razzak, S.A. Hussain, A. Belaïd, M. Sher, Multi-font numerals recognition for urdu script based languages, International Journal of Recent Trends in Engineering (IJRTE) (2009).
[120] K. Horio, T. Yamakawa, Handwritten character recognition based on relative position of local features extracted by self-organizing maps, International Journal of Innovative Computing, Information and Control 3 (4) (2007) 789–798.
[121] A. Durrani, Urdu Informatics, Center of Excellence for Urdu Informatics, National Language Authority, Islamabad, Pakistan, 2008.
[122] A. Durrani, Pakistani: Lingual Aspect of National Integration of Pakistan, in: Ministry of Education Curriculum Review.
[123] M.I. Razzak, A.A. Mirza, Ghost character recognition theory and Arabic script based languages character recognition, Przegla,d Elektrotechniczny R 87 (11) (2011) 234–238.

Saeeda Naz received her BS degree from the University of Peshawar, Pakistan in 2006 and MS (Computer Science) degree from the COMSATS Institute of Information Technology (CIIT), Pakistan in 2012. Currently, she is doing her PhD in Computer Science from CIIT, Pakistan. She is a lecturer at the Higher Education Department Khyber-Pakhtunkhwa, Pakistan, since Dec 2008. Her areas of interest are Optical Character Recognition and Pattern Recognition.

Khizar Hayat, an Associate Professor by designation, is currently heading the Computer Science Department at the Abbottabad campus of COMSATS Institute of Information Technology (CIIT), Pakistan. Before joining CIIT in December 2009, he was working as a lecturer at the Higher Education Department Khyber-Pakhtunkhwa, Pakistan, since May 1995. He received his PhD (Computer Science) degree in June 2009 from the University of Montpellier II (UM2), France, while working at the Laboratory of Informatics, Robotics and Microelectronics Montpellier (LIRMM). His areas of interest are image processing and information hiding.

Muhammad Imran Razzak is currently working as assistant professor at King Saud bin Abdulaziz University for Health Sciences (KSUAHS, Saudi Arabia). Previously, he has served in University of Technology, (UTM, Malaysia), King Saud University, (KSU, Saudi Arabia) Air University, (AU, Islamabad, Pakistan), and Umm ul Qura University, (UQU, Saudi Arabia). Dr. Imran has one patient, two book chapters and more than 35 publications published in international journal and conferences. His area of research mainly focuses on Document Image Processing, Medical Imaging and Biometrics Security. Moreover, he is an editorial member of Journal of Health Informatics (in Developing Countries (JHIDC) and The Computer Science Journal (CSJ).

Muhammad Waqas Anwar is currently working at COMSATS Institute of Information Technology, Pakistan as an Associate Professor since April 2008. He got his PhD degree in Computer Application Technology from Harbin Institute of Technology, P.R. China in 2008. He did Masters in Computer Science from Hamdard University, Pakistan in 2001. He is an active researcher and his areas of interest are Natural Language Processing and Computational Intelligence.

Sajjad A. Madani works at COMSATS Institute of Information technology (CIIT) Abbottabad Campus as associate professor. He joined CIIT in August 2008 as Assistant Professor. Previous to that, he was with the institute of computer technology from 2005 to 2008 as guest researcher where he did his Ph.D. research. Prior to joining ICT, he taught at COMSATS institute of Information Technology for a period of two years. He has done M.S.in Computer Sciences from Lahore University of Management Sciences (LUMS), Pakistan with excellent academic standing. He has already done B.Sc. Civil Engineering from UET Peshawar and was awarded a gold medal for his outstanding performance in academics. His areas of interest include low power wireless sensor network and application of industrial informatics to electrical energy networks. He has published more than 40 papers in peer reviewed international conferences and journals.

Samee U. Khan received a BS degree from Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, Pakistan, and a PhD from the University of Texas, Arlington, TX, USA. Currently, he is Assistant Professor of Electrical and Computer Engineering at the North Dakota State University, Fargo, ND, USA. Prof. Khan's research interests include optimization, robustness, and security of: cloud, grid, cluster and big data computing, social networks, wired and wireless networks, power systems, smart grids, and optical networks. His work has appeared in over 200 publications. He is a Fellow of the Institution of Engineering and Technology (IET, formerly IEE), and a Fellow of the British Computer Society (BCS).