A Novel Similar Character Discrimination Method for Online Handwritten Urdu Character Recognition in Half Forms

Quara tul Ain Safdar[a,*], Kamran Ullah Khan[b,**], Liangrui Peng[c]

[a]*Department of Electrical Engineering, Pakistan Institute of Engineering and Applied Sciences, Islamabad, Pakistan.*
[b]*Department of Electrical Engineering, Pakistan Institute of Engineering and Applied Sciences, Islamabad, Pakistan.*
[c]*Department of Electronic Engineering, Tsinghua University, Beijing, China.*

Abstract

Online handwritten Urdu character recognition is one of the key technologies for intelligent interface on smart phones and touch screens. It is a challenging research topic as Urdu script has many similar character groups. A novel similar character discrimination method for online handwritten Urdu character recognition is proposed in this paper which includes pre-classification, feature extraction and fine classification process. The pre-classifier enables the discrimination of similar characters by putting them in distinct smaller subsets according to stroke number and diacritics. Then structural features and wavelet features are extracted. Finally, Support Vector Machines (SVM), Artificial Neural Networks (ANN), and Recurrent Neural Network (RNN) classifiers are compared for fine classification within subsets. Results of RNN classifier without using the proposed pre-classifier and features have also been obtained to check the end-to-end capability of the RNN classifier. Experimental results show that the proposed method is efficient and achieves an overall accuracy of 96% on a large-scale self-collected dataset. It is feasible to extend this method for other Arabic scripts.

Keywords: online, handwritten character recognition, half forms, multistroke Urdu characters, wavelets, ANN, SVM, RNN

1. Introduction

Development of portable computing devices, writing pads, and smartphones with non-keyboard-based input interface are receiving much attention in the research communities and commercial sector. The provision of an interface that can recognize handwritten inputs efficiently is a non-trivial task owing to complexities involved in handwritings, limited memory, and relatively lesser processing resources available in mobile devices. From a user's point of view, online recognition systems, compared with off-line ones, are receiving more acknowledgment because of their convenience in writing compared to typing, due to their usefulness in situations when typing is hard, owing to the inadequate keyboard facility on small computers, and due to difficulty to type in some languages for their large number of letters [1]. From a developer's point of view, advantage of a pen-tablet environment (the online system) is that it facilitates the process of recognition with some important information, which is missing in its counterpart (that is, the off-line system). For example, the handwritten stroke coordinates are available in online systems as a function of time along with respective pen's pressure values. Moreover, auxiliary information of writing speed, stroke order, and pen-up/down events can be tracked as well in these online systems. In spite of these difficulties, *state of the art* character-recognition systems development work (either in off-line or online mode) on alphanumeric handwritings has been reported for various languages, like English [2], Persian [3], Chinese [4], Japanese [5], Mongolian [6], Kannada [7], and Arabic [8, 9, 10, 11, 12, 13, 14]. In fact, there are well-matured online handwriting recognition software for Latin-script-based languages (like English) and for character-based languages (such as Chinese).

Urdu is one of the script-based languages derived from Arabic and Persian. According to estimates, it is written, spoken, and used by more than 200 Million people around the world. Urdu is officially recognized in India due to existing 70 Million native Urdu speakers. Urdu language is also spoken and understood in Nepal, Bangladesh, the Middle East, Fiji, USA and many other countries around the globe, including UK (having about 400,000 native Urdu speakers). In Pakistan, populated with approximately 200 Millions, Urdu is the primary language of communication and there are about 130 Million mobile phone users [15]. According to market estimates, based on current trends in the e-commerce sector, there could be 40 million smartphones in Pakistan by December 2016 [16]. In that scenario, there is a need to carry out research in the field of design and development of online Urdu handwriting recognition systems for computing devices (like smartphones) to provide benefit to the large Urdu speaking population of the world. Online Urdu handwriting recognition system can also extend its benefits to the users of other Arabic script based languages like Persian, Uyghur, Sindhi, Punjabi and Pushto with minor modifications.

Urdu script comprises of a larger character-set with cursively written and contextually dependent alphabets. Be-

*Principal corresponding author
**Corresponding author
Email addresses: qa@pieas.edu.pk (Quara tul Ain Safdar), kamran@pieas.edu.pk (Kamran Ullah Khan), penglr@tsinghua.edu.cn (Liangrui Peng)

ing context dependent, Urdu alphabets adjust their shapes according to the 'preceding and following' characters. In this way, for an Urdu alphabet there are one full and at least three different half forms with few exceptions. Moreover, complexities for Urdu handwriting recognition arise not only due to cursiveness and context dependency but also because of the very nature of an alphabet structure, word formation in a particular font-style, and diacritics involved in alphabets. Overlapping ligatures, delicate joints of characters in a word, aslant traces, neither fixed baseline nor standard slope (in Nastalique font style), associated dot and other diacritics which may be above, below or within the character, displacement of dots with base stroke's slope and context [17, 18, 19] are a few to shed light on the complexities of Urdu script.

On the basis of target-set, Urdu handwriting recognition (both off-line and online) can be placed into three categories: isolated or full form character recognition [20, 21, 22, 23], selecting ligatures for recognition or holistic approach (also known as segmentation-free approach) [17, 24, 25, 26, 27, 28], and segmentation-based or analytical approach [29, 30, 31, 32, 33, 34, 35].

Moreover, different researchers have tried to address the recognition problem by focusing on different aspects. For example, the authors in [36] worked out the baseline (an imaginary line on which characters are combined to form the ligatures) of the character stroke, the work in [37] discussed the diacritical marks associated with characters and ligatures, and the approach in [38] emphasized the preprocessing operations.

Following the analytical approach along with dictionary based search to obtain valid characters and words, Malik et al. [20] recognized 39 isolated characters with an overall accuracy of 93% and 200 two-unattached-character ligatures with an accuracy of 78%. Hussain et al. [25] preferred the holistic approach, proposed spatial temporal artificial neuron for the recognition, and reported an accuracy of 85% for 15 selected ligatures only. However, their data set lacks the aspect of generality as it was acquired from only two different writers. Husain et al. [26] investigated the recognition system for one-, two-, and three-character-ligatures and obtained separate results of 93% and 98% for base and secondary strokes, respectively. Shahzad et al. [21] studied the recognition of 38 isolated Urdu characters using 9 geometric features for primary stroke and 4 for secondary stroke to achieve the accuracy of 92.8% for the data obtained only from two native writers; however, the recognition rate diminished to 31% when the characters were scribbled by an untrained non-native writer. With trained non-native writer's scribbled data, the recognition rate barely increased to 73%. Razzak et al. [27, 28] investigated the recognition system for 1800 ligatures. By utilizing the features based on fuzzy rules and hidden Morkov model, they secured 87.6% recognition rate for Urdu Nastalique font and 74.1% for Naskh font. Most of the work available in the online domain of Urdu character recognition deals with ligatures and full form recognition. Segmentation-based approaches have been applied either to segment the ligatures from each other present in a word or to dissociate the diacritics from the base-character [19]. Here is to note that, to the best of authors' knowledge, no work is found there using wavelet analysis for recognition of Urdu characters. However, studies have been reported for Arabic and Persian characters recognition using wavelets. Therefore on the basis of alike-script and wavelet analysis the work presented in this paper is compared with Arabic and Persian work as well. Table 1 accounts the comparison of proposed work with Arabic and Persian recognition systems using wavelet analysis.

Inspired from [22, 23, 39] the authors propose in this work the online Urdu character recognition problem for context-dependent shapes of Urdu characters, that is, for half-forms. For the development of online cursive Urdu handwriting recognition system, recognition of half form Urdu characters is a primary step because of the following four reasons: First, Urdu characters appear in half forms in a word. Although full form letters are also used within a word, yet the role of half forms is much more than that of full forms. Second, half form characters are the building blocks for ligatures and therefore segmentation-based systems eventually attempt to recognize the constituent half forms [29, 33, 34, 35]. Third, there are a lot more ligatures in Urdu, which cannot be entirely enclosed within the scope of a single study. That is why, researchers in their works have tried to recognize selective number of ligatures through which many words can be composed, but not all. Consequently, such systems have limited vocabulary available for processing [27, 38, 19]. Furthermore, for acquiring a valid ligature or finding an optimum word, dictionary based search becomes a necessary part of the work [26], however, this is not the case with the half forms. Last, targeting half forms would mean independence from dictionary. Even new words not present in dictionary can be recognized.

Instead of putting all the characters at once into a single recognizer, we opted to pre-classify the larger half-form character set into smaller subsets. A pre-classifier has been proposed which puts similar characters into distinct smaller subgroups. Then, these smaller subgroups are targeted for further classification through ANN and SVM classifiers by employing wavelet and structural features, and also by RNN classifier using the raw stroke data (without using the extracted features). In character recognition problems, wavelet transform has been used for languages like English [40], Chinese [41], Arabic [42], Persian [43, 44], and different Indian languages [45, 46]. There is availability of Arabic handwritten words database (Arabic DAtaBase: ADAB [47]) but for Urdu there is a lack of standard handwritten character database. The end-to-end recognition capability of the RNN classifier [48, 49] has also been utilized in which all the characters are fed to the RNN classifier without any feature extraction or pre-classification. A large database of Urdu handwritten characters has also been developed by the authors which can be provided for research purposes. The main contributions of this work are as follows:

1. A framework for development of online Urdu handwriting recognition for smartphones has been presented.

2. Based on the number of strokes in a character and the position and shape of diacrtics, segregation of larger character set into smaller subsets is obtained through the proposed pre-classification in contrast to the previous online Urdu character recognition approaches like [20, 25, 26, 21, 27, 28, 38].
3. To cope with the demand of robust and accurate recognition along with relatively low computational power and limited memory available to mobile devices, banks of computationally less complex classifiers are developed, from which the appropriate classifier would be loaded to the memory to achieve the recognition task.
4. A comparison of different classifier/feature combinations is presented in this study to distinguish between features' discrimination and classifiers' recognition ability.
5. A comparison of feature-based classifiers (ANN, SVM) and end-to-end classifier (RNN) is presented.
6. Noting the small databases of existing Urdu character recognition works [25, 21, 27, 28], a large database of handwritten Urdu characters is developed and employed in this study, which contains 10800 samples of all Urdu half form characters (100 samples of each character) acquired from 100 writers. The database can be obtained from the authors for the research purposes.

For different classifier/feature combinations, the overall accuracies obtained through the proposed methods are 81.9%, 92.8%, 95.8%, 96.1%, 84.7%, 87.2%, and 60% (to be detailed in results). The best overall recognition rate is procured through SVM. For individual characters, the recognition rates obtained are up to 100% by application of the resultant schemes.

The organization of the paper is as follows. A brief introduction about the Urdu character set in half forms is provided in Section 2. The proposed online Urdu handwriting recognition system is explained in Section 3. Results and discussions are presented in Section 4. The paper is concluded in Section 5.

2. About the Urdu Character Set in Half Forms

In this section we will analyze the way in which Urdu words are handwritten by the native writers. Urdu handwriting is inherently cursive and there are many Urdu font styles available, such as *Naskh*, *Nastalique*, *Kofi*, *Thuluth*, *Diwani*, and *Rouq'i* . *Nastalique* style is mostly adopted for Urdu writing whereas Arabic is penned in *Naskh* style. With respect to the position in a word, *Nastalique* font style reveals writing with atilt ligatures and distinctive variations in letters [19]. For example, the character 'ٹ' adapting three different shapes as per context is shown in Fig. 1.

Most of the characters in Urdu words appear in three different forms as shown in Fig. 1 (see also [50]). The form in which a character appears in a word depends on the position in which it occurs in the word. These forms are described below:

- **Full form**: Every character in Urdu has a full form. The full forms always occur in isolated positions in a word (not ligature). Urdu character set consists of 37 characters (letters /alphabets) in full form [23], however, this count is reported 39 in [19]. The difference is due to the addition of some characters to the basic Urdu character set.
- **Initial half form**: A character falling in the beginning of a word (more generally, a ligature) adopts the initial form. Not every character has an initial half form. There are 36 initial half forms.
- **Medial half form**: Characters falling in the middle of a word (or ligature) adopt their medial forms. Some characters do not have medial half forms. These are 30 in number.
- **Terminal half form**: Character falling in the end of a word (or ligature) adopts the terminal half form. Every character has a terminal form. The terminal forms have very much similar shapes as compared to their corresponding full forms. There are 42 terminal half forms.

All Urdu characters in half forms (108 in number) are shown in Fig. 2. The target of this work is to classify all these 108 shapes while handwritten online. Some characters possess more than one *shapes* at initial, medial or terminal positions. The usage of those shapes depend on the **context** in which that character appears. The context means which other characters appear, in a ligature, before and after a particular character.

Analyzing Urdu characters in further detail, we find that an Urdu character consists of a major stroke and may have none, one, two, or three minor strokes. There are few groups of characters in which the major stroke is common to the group and the distinction among the characters is made on the basis of the type, count, and position of the minor strokes. Depending on the number of strokes, we categorize Urdu characters into the four subsets namely single-, two-, three-, and four-stroke characters.A few examples specifying the use of two-, three-, and four-stroke characters as initial half forms have been shown in Fig. 3.

There are five different types of minor strokes on the basis of shape drawn: dot or *nuqta* ('.'), *towey* ('ط'), inverted *hay* ('ہ'), *hamza* ('ء') and *kash* ('/'). Moreover, minor stroke(s) may be placed above or below a major stroke depending upon the character. Multistroke characters can also be grouped on the basis of the position of secondary stroke(s) with respect to primary stroke as well as on the basis of the shape of secondary stroke.

Due to the presence of similar characters, various half forms, context dependency of shape of a character (108 shapes), and different types of minor strokes, recognition of online handwritten Urdu characters is a complex and challenging pattern recognition problem.

3. Proposed Online Handwritten Urdu Character Recognition System

In this section we present the proposed online handwritten Urdu character recognition system. The whole system consists of the data acquisition, preprocessing, pre-classification, feature extraction, and classification stages. The block diagram of the whole system is shown in Fig. 4 and all the stages are explained in the following subsections.

3.1. *Data Acquisition*

Handwriting data can be acquired using a pen-tablet device connected to a computer. Data may also be acquired by writing on the touch sensitive screen of a smartphone. In our study, 100 native Urdu writers of different age groups have provided their handwriting samples using a stylus and digitizing tablet. Online handwritten character signals contain the information of the digitized coordinates $(x(t), y(t))$, and the pressure values and time-stamps for each point $(x(t), y(t))$. During the data acquisition, the following attributes of character strokes were acquired:

1. Number of times the pen is up/down.
2. Number of strokes in a character.
3. Starting/ending index of each stroke.
4. Temporal order of each sample of $(x(t), y(t))$ coordinates.
5. Pressure value at $(x(t), y(t))$. Note: pressure value is not utilized in this work except for detecting pen up/down events.

3.1.1. About the Data

The data obtained from the writers is in segmented form. Figure 5 shows few examples of full Urdu words and ligatures composed from the segmented characters obtained from the participating writers, and demonstrates that the words composed from these segmented characters do resemble the words as if written continuously. To use a recognition system based on our proposed method in its current form, it is required to draw the characters in their segmented forms. If the visual feeling of continuous word is required then the segmented characters should be drawn at appropriate positions as shown in fig. 5. We are also working on segmentation of characters from ligatures and will be reported in future. A related work on segmentation of handwritten Arabic text can be found at [51] that presents an efficient skeleton-based grapheme segmentation algorithm. With some modifications, this segmentation algorithm along with our proposed methodology may serve as a full system for online Urdu handwriting recognition. Segmentation of printed Urdu script can be found in [33, 35, 34].

3.1.2. Instructions for writing

For non-native audience, here we present some instructions that should be followed while writing Urdu characters (These instructions are implicitly followed by native Urdu writers)

- There should be no pen-up event while drawing the major stroke i.e. the major stroke should be drawn continuously without raising the pen,
- In case of multistroke characters, the major stroke should precede the minor stroke(s), and
- Minor strokes should be penned one at a time, i.e. there must be pen up events between two or three dots or between two '*kashes*'. In some cases, this instruction is violated by the native writers, but for this work, we stress on following this instruction.

Although two minor strokes drawn together (for example, two dots) can be separated using the variation in pressure values however this is not implemented in this paper.

3.2. *Preprocessing*

The raw data obtained from hardware contains artifacts like jitters, hooks in start and end of a stroke, speed variations etc. To reduce the effect of artifacts, the following preprocessing steps have been performed on the raw data.

3.2.1. Re-Sampling

Algorithm 1 has been implemented to remove repeated data samples (those occurring consecutively in temporal order). Then a downsampled version of this signal has been obtained by keeping every second data sample starting with the first. Few samples of downsampled data are shown in Fig. 6(a).

Algorithm 1 Repeated elements removed

```
procedure REMOVEREPEATEDDATAPOINTS(S)
        ▷ S_(Mx2) contains X and Y coordinates of a given stroke
    initialize k ← 1
    initialize Sr(k) ← S(1)
    for i = 2 to M do
        if ||(S(i − 1) − S(i)||_2 = 0 then
            Sr(k) ← S(i)
        else
            k ← k + 1
            Sr(k) ← S(i)
        end if
    end for
    return Sr
end procedure
```

3.2.2. Smoothing

Drawing on a tablet by inexperienced users, or roughness of pen tip or writing surface may result in jitters and trembles in writing. To mitigate jittering effects the character data is smoothed using a 5-point moving average filter given by the following difference equation:

$$y_s(i) = \frac{1}{2N+1}(y(i+N) + y(i+N-1) + ... + y(i-N)) \quad (1)$$

where $y_s(i)$ is smoothed value for the *ith* data point, N is the number of neighboring data points on either side of $y_s(i)$ (in this case N=2), and $2N+1$ is the span. The results of smoothing function are shown in Fig. 6(b).

3.3. *Pre-Classification*

There are many groups of characters in Urdu that share the same major stroke and differ from each other due to their minor strokes. These similar characters pose difficulty in classification. A concept of pre-classifier is presented here. The pre-classifier classifies the characters in to smaller subgroups. The classification criterion is derived from the properties of Urdu characters as presented in section 2. The pre-classification for initial half forms is explained here, whereas the pre-classification of medial and terminal half forms is similar.

In the first phase of pre-classification, the character set (initial half forms) is divided into different groups on the basis of number of pen-up events. The number of pen-up events actually represents the number of strokes in a character. Four subsets are yielded (see details in section 2): single-, two-, three-, and four-stroke subsets. In the second phase of pre-classification, on the basis of position of the diacritics, every multistroke subset obtained in the first phase is segregated into two sub-subsets. Position of the diacritics for multistroke Urdu characters is either above or below the major stroke. Therefore, in the end of the second phase of pre-classification, we get 6 sub-subsets of multistroke initial half form characters.

For Urdu characters, we place diacritics into two types on the basis of shape:

1. *Dot* or *nuqta* ('.') diacritic.
2. *Other-than-dot* diacritic. These are *towey* ('ط'), inverted *hay* ('ہ'), *hamza* ('ء') and *kash* ('/').

In third phase of pre-classification, sub-subsets obtained in second phase are further divided on the basis of shape of the diacritic to produce 9 sub-sub-subsets. As a result, finally we get 10 subsets for initial half form characters. Figure 7 gives a pictorial description of pre-classification of initial half form characters. Table 2 shows the pre-classification of Urdu character set in initial, medial, and terminal half forms.

In Urdu, there occurs no such case where we could have 3-stroke characters with *other-than-dot* diacritic below the major stroke. 4-stroke characters exist only with *dot* diacritic and no *other-than-dot* diacritic is present there. In this way of subdivision, we see that character 'ٹ', in three-stroke characters group, stands alone in its subset having no competition for classification. Table 3 shows those characters which face no competition in their respective subsets. These are fully recognized at pre-classification stage and do not require any further recognition.

With the small subsets produced by the pre-classifier, it becomes possible to design banks of simple ANN or SVM classifiers for fine classification within the subsets.

3.4. ***Feature Extraction***

Selection of appropriate features for recognition tasks is necessary for achieving high performance [52]. Computing suitable features, in every online system, helps reducing the computational complexity of a pattern recognition problem [45]. However, selection and extraction of such features does not follow any specific technique. Variations involved in one kind of a problem manifests that a feature set designated for a particular problem may not necessarily be satisfactory for a similar problem. One can deduce the fact that no widely accepted feature set contemporarily exists that can survive successfully for at least one kind of problems [53]. To reduce computational complexity prominent features are acquired from preprocessed data. However, optimum size of feature vector to recognize a handwritten character depends on the complexity involved.

For Arabic/Urdu handwritten characters recognition, different types of features have been presented in literature, namely structural features, statistical features and global transformation features. Using structural features [20, 25, 26], a model/standard template is designed for each class of letters that contains all the significant information with which test classes are compared. Statistical approach uses the information of the underlying statistical distribution of some measurable events or phenomena of interest in the input data [22, 23]. With global transformation features, the recognition problem is taken up in frequency domain using transformations like Fourier, discrete cosine, Gabor, and Walsh-Hadamard etc. [19].However, to determine and analyze localized features of a signal/image, a time-scale representation of that signal/image is used i.e. wavelet transform. In [54] wavelet transform has been used for optical character recognition of multifont English text. The Wavelet transform is a multiresolution technique that clips data into different frequency components, and then analyzes each component with a resolution matched to its scale [55]. Wavelet series expansion of a a function $f(x)$ is given in eq.2.

$$f(x) = \sum_k c_{j_o}(k)\varphi_{j_o,k}(x) + \sum_{j=j_o}^{\infty}\sum_k d_j(k)\Psi_{j,k}(x) \qquad (2)$$

where $c_{j_o}(k)$ are approximation (or scaling) coefficients, and $d_j(k)$ are detailed (or wavelet coefficients) [56]. Details about the wavelets can be studied from [55] however for a brief review of wavelet properties can be studied from [57].

3.4.1. Wavelet Features

To discriminate characters from each other, a human reader looks for the exact location of smooth regions, sharp turns, and cusps as the landmarks of interest. With structural, statistical and global transformation features as used in [20, 25, 26, 22, 23, 19] it is not possible to find out these landmarks exactly. In proposed study, wavelet transformation of handwritten stroke data enables us to accurately pinpoint the mentioned landmarks and leads to attain better recognition rates. To verify the discriminating potential of wavelet features, a multilevel one-dimensional wavelet analysis is applied to the preprocessed data. Approximation and detail coefficients are obtained for the $x(t)$ and $y(t)$ coordinates of the handwritten strokes. In order to obtain better classification accuracy and keeping the feature vector as small as possible, it was found after some trials that the level-2 approximation coefficients and level-4 detail coefficients were providing the best classification accuracy. The feature vector is

$$\mathbf{W} = \left[\ \overrightarrow{cA2}_x \quad \overrightarrow{cD4}_x \quad \overrightarrow{cA2}_y \quad \overrightarrow{cD4}_y\ \right]^T \in \mathbb{R}^n \qquad (3)$$

where $\overrightarrow{cA2}_x$ and $\overrightarrow{cA2}_y$ are the vectors of level-2 approximation coefficients, and $\overrightarrow{cD4}_x$ and $\overrightarrow{cD4}_y$ are the vectors of level-4 detail coefficients of the one dimensional $x(t)$ and $y(t)$ signals of the stroke coordinates $(x(t), y(t))$. C++ or Matlab codes may be used to obtain the wavelet coefficients.

Figures 8 and 9 (a,b), show four different handwritten characters in medial half form. Each of these figures show the handwritten stroke and $x(t)$, and $y(t)$ signals of the major stroke in the top row, the second row shows the $\overrightarrow{cA2}_x$ and $\overrightarrow{cA2}_y$ coefficients, while the third row shows the $\overrightarrow{cD4}_x$ and $\overrightarrow{cD4}_y$ coefficients.

Figure 10 is representative of the case where *other-than-dot* minor stroke is involved. In this case there were characters having similar major strokes and were distinguishable from each other only on the basis of the shape of their minor strokes. Since the minor stroke in this case is significantly long, the wavelet coefficients of the minor stroke was also included along with the wavelet coefficients of the major stroke to form the feature vector.

It can be easily observed from Fig. 8 to 10 that the wavelet coefficients of different characters are quite different from each other. Such variability provide the promise of wavelet features to present good discrimination power. The results have verified that using wavelet features, in the way presented above, provided high recognition rates.

3.4.2. Structural Features

In this study, for comparison purpose, in addition to wavelet based features, structural features proposed by Khan and Haider [22, 23], have also been employed and tested. It is shown in the results (Section 4) that with wavelet features the recognition accuracy is far better than that obtained with structural features.

*3.5. **Classification***

For fine classification of each character within the subsets produced by the pre-classifier, a dedicated classifier was designed for each of the subsets. In this work, the responses of ANN and SVM classifiers along with different input features have been studied. Moreover, RNNs have also been applied to compare the responses obtained through ANN and SVM.

3.5.1. Artificial Neural Networks

For pattern recognition problems, developing a multilayer perceptron (MLP) neural network with backpropagation algorithm is very popular approach [58, 59, 60, 61, 62]. The ANNs used in this work are single or multilayer Back Propagation Neural Networks (BPNN). For each of the 19 subsets (cardinality ≥ 2), an ANN was configured, trained, and tested. In this way a bank of ANNs was obtained in which each neural network serves to recognize a specific character subset. There are two different banks of ANNs:

1. ANNs which are trained using *structural* features.
2. ANNs which are trained using wavelet *db2* approximation and detailed coefficients. Table 4 presents configurations of these ANNs.

Using MATLAB environment, all ANNs were trained on 40% (40 instances of each character) and tested for remaining 60% (4260 samples) of the data set.

3.5.2. Support Vector Machines (SVM)

SVM are also widely used for pattern classification and recognition [62, 63]. Speciality of SVM is that the minimization of empirical classification error and maximization of geometric margins occur simultaneously. To make a comparison of recognition results obtained through ANN classifiers using wavelet features, recognition results using SVM have also been obtained. Two banks of SVM classifiers using wavelet features (*db2* and *bior1.3*) were trained and tested. SVM was setup using LIBSVM (Matlab) [64]. LIBSVM offers to select different types of kernel functions (e.g. linear, polynomial, radial basis function (RBF), sigmoid etc.) with various parameters of these kernels. For the proposed study, C-SVM (multiclass classification) with radial basis function is employed. For the selection of good parameters, the training set is used with 5-fold cross validation and optimized values are obtained (of cost of constraint violation C and γ in radial basis function). All the SVMs are then trained with randomly selected 40% of sample data, while tested on 60% of the remaining data.

3.5.3. Recurrent Neural Networks: Long Short-Term Memory

Recurrent neural networks (RNNs) introduce a notion of time to traditional feedforward artificial neural networks which enables the network to make use of the temporal patterns present in the sequential data. In a sequential set of data, the current output depends on previously computed values. RNNs are elevated with the inclusion of edges that span the adjacent time steps. For sequence learning, Long Short-Term Memory (LSTM) and Bidirectional Recurrent Neural Networks (BRNN) are considered to be the most successful RNN architectures. In LSTM RNNs traditional nodes in the hidden layer of a network are replaced by a memory unit. The architecture of Bidirectional Recurrent Neural Networks utilize the information from both the past and the future to compute the output at any point in the sequence [65]. It helped the recurrent neural networks to be applicable to cursively handwritten scripts more efficiently.

In this work, using RNNLIB [66], RNNs with LSTM architecture, without any feature extraction and with/without using the proposed pre-classification, are applied to the handwritten data . With proposed pre-classification, each subset is presented to a recurrent neural network which is specifically trained for that subset. Results of RNN classifier without using the proposed pre-classifier have also been obtained to check the end-to-end capability of the RNN classifier. Using the raw stroke data, each RNN is trained, validated, and tested on 30%, 20%, and 50% of randomly selected subsets of the data set respectively.

4. Results and Discussion

The pre-classifier produced a total of 28 subsets from the set of 108 half form characters (Table 2). Out of these 28 subsets there are 6 subsets containing only one characters and do not need any further classification (Table 3). There are 3 subsets containing single stroke characters for which some results are presented in [39]. The remaining 19 subsets contain multistroke characters for which six different combinations of classifiers and features were tried to classify the individual characters in the subsets:

1. ANN classifiers using structural features.
2. ANN classifiers using Daubechies' family *db2* wavelet features.
3. SVM classifiers using Daubechies' family *db2* wavelet features.
4. SVM classifiers using Biorthogonal family *bior1.3* wavelet features.
5. RNN classifiers using Single LSTM hidden layer of size 100 (no feature extraction, with pre-classification)
6. RNN classifiers using Multi LSTM hidden layers of varying sizes (no feature extraction, with pre-classification)
7. RNN classifier using Multi LSTM hidden layers of varying sizes (no feature extraction, no pre-classification)

These 19 subsets contain 8, 7, and 4 subsets of initial, medial, and terminal half form characters respectively containing a total of 71 multistroke characters. The discrimination of similar characters from each other is made easier by the pre-classifier because it puts similar character into different subsets. Since a subset contains quite dissimilar characters, the pre-classifier also allows the use of computationally simpler ANN or SVM classifiers for fine classification of individual characters within a subset.

The recognition results are shown in Table 5. Among these seven classifier/features combinations, the best overall recognition accuracy of 96.1% was obtained by using *db2* wavelet features with SVM classifier, but SVM with *bior1.3* wavelet features also provided comparable overall accuracy of 95.8%. ANN with *db2* wavelet features provided somewhat lesser overall accuracy of 92.8% as compared to SVM. For ANNs with structural features the overall accuracy of 81.9% is significantly lower as compared to the other three combinations. Note that the data set contains 100 instances of each character; For each ANN 40 instances were used for training purpose, while 60 instances were used for testing purpose of the classifiers. Overall recognition results using RNNs were 84.7% (RNN with single LSTM hidden layer of size 100) and 87.2% (RNN with multi LSTM hidden layers of varying sizes).

The end-to-end recognition capability of RNN was also checked without utilizing the proposed pre-classifier and any features. The raw stroke data of all the 108 character classes were used to train a single RNN classifier. Different configurations were tried for the RNN classifier. The best recognition rate of 60% was obtained and the training time was more than 100 hours. Note that for the RNNs, 30%, 20%, and 50% of sample data was randomly selected for training, validation, and testing purposes, respectively.

4.1. *Error Analysis using Confusion Matrices*

Some confusion matrices will be presented in this section for the best and worst cases of the best classifier/features combination i.e. SVM+db2-wavelet-features.

Table 6 shows the confusion matrix of a subset containing 6 characters. The recognition accuracy of 91.9% for this subset is among the lowest accuracies obtained with the SVM+db2-wavelet-features combination. The character 'ـؤ' is 3 times misclassified as ؤ and 4 times misclassified as ـڑ. This should be expected because of the shape similarity among these characters. Similarly ـڑ is 7 times misclassified as ڑ and 4 times misclassified as ـؤ for the same reason.

Table 7 shows the confusion matrix for another subset yielding low overall recognition accuracy (93.6%) with the SVM+db2-wavelet-features combination. The main culprits for the low accuracy in this subset are the characters فـ and غـ. Although فـ and غـ have distinct major strokes in standard form with غـ having a cusp in its major stroke, many writers ignore this cusp while handwriting غـ casually. The غـ then appears very much similar to فـ. This is confirmed by the confusion matrix which shows that غـ is 7 times misclassified as فـ. Removing غـ and فـ from this subset results in 97.9% accuracy. Removing only فـ results in 95% accuracy, while removing only غـ gives 97.9% accuracy.

Confusion matrix of another subset yielding low overall accuracy of 93.3% is presented in Table 8. Here ـئـ and ـٹـ are responsible for the low recognition rate. Both characters have same major stroke but distinct minor strokes, so minor stroke was also utilized for feature vector formation. But casual penning of minor strokes results in similar shapes of the minor strokes. Consequently ـٹـ is 9 times misclassified as ـئـ.

Table 9 and Table 10 present two subsets showing high overall recognition accuracy.

4.2. *Confusing Characters*

In Urdu there are few groups of characters in which the major stroke is common to the group and the discrimination is made on the basis of minor strokes. This similarity is inherent to Urdu and the similar characters were put into different subsets by the pre-classifier. There is another kind of similarity between different characters which arises from the careless writing by the user. This user imposed similarity occurs inside the subsets produced by the pre-classifier and results in confusing pairs of characters within a subset.

Figure 11 shows few handwritten samples of two confusing characters فـ (*Fay*) and غـ (*Ghain*) present in the subset presented in Table 7. If drawn according to rules, the character غـ should have a well defined cusp in its major stroke. Some users do not draw the cusp while writing casually or in hurry. The غـ drawn in this way appears like فـ to a human reader as can be seen in Fig. 11. The classifier also many times misclassified غـ as فـ as shown in Table 7.

Another pair of confusing characters is shown in Fig. 12. These are the characters ـئـ (*hamza*) and ـٹـ (*Tay*) in medial form. The major strokes for both the characters is the same and the discrimination was made on the basis of minor stroke. Many users casually draw the minor stroke of ـٹـ in a very much similar way to the minor stroke of ـئـ. The confusion matrix in Table 8 for this subset confirms this where it can be seen that ـہؕ has been 9 times misclassified as ـۂ.

4.3. *RNN and Pre-classification*

Urdu character subsets resulted from pre-classification are presented to (configuration-wise) two types of RNNs. In the first set of RNNs, the configuration of each RNN consists of a single LSTM hidden layer of size 100 and trained separately for each character subset. It produced overall 84% recognition rate. In the second set of RNNs, each RNN is configured with multi LSTM hidden layers of different sizes. After many hit and trials the best RNNs are selected and the recognition rate obtained is 87%. Comparison of recognition rates obtained through fixed-size and varying-size configurations accounts that the later is more acceptable. Recurrent neural networks are also applied to handwritten Urdu data without going through pre-classification process. All the 108 classes are presented to only one RNN. After many hit and trials the best RNN thus obtained resulted in a recognition rate of 60%. It can be observed that without pre-classification the recognition rate is substantially lower than that with pre-classification. The recognition rate may be improved for the RNN further if more data is added to the handwritten Urdu database. However, in the context of current study, the difference among the three results obtained with RNNs show that due to the complexity of and similarity among Urdu characters, the proposed pre-classification proved helpful to obtain better results.

5. Conclusions

In this study, a novel character recognition system for Urdu language online handwritten characters is presented. All multistroke initial, medial, and terminal half form characters have been recognized. A large scale handwriting data set was obtained from 100 native Urdu writers of different age groups and educational qualifications. The data was acquired using a digitizing tablet. Spatial coordinates in temporal order with their respective pressure values, and pen up/down events were recorded. The raw data was refined after its manipulation with different preprocessing operations. A novel pre-classifier was designed to pre-classify Urdu characters set into smaller subsets. The pre-classifier yielded smaller subsets based on the number of strokes to yield two-, three-, four-stroke subsets. The pre-classifier further divided the subsets based on the position of the minor stroke with respect to the major stroke, and also on the basis of whether the minor stroke is a *dot* or *other-than-dot*. The pre-classifier helped in discriminating similar characters from each other by putting them in different subsets. Two types of features, namely structural features and wavelet transform features were extracted. Wavelet features were obtained using Daubechies *db2* coefficients and Biorthogonal *bior1.3* coefficients. ANN, SVM and RNN classifiers were used for fine classification of the individual characters in the subsets generated by the pre-classifier. Results of RNN classifier without using the proposed pre-classifier and features were also been obtained to check the end-to-end capability of the RNN classifier. Since there is no sufficient previous work for comparison, different combinations of features and classifiers were tried to find the best recognition results. Seven different classifier/feature combinations were tried which resulted in overall accuracies of 81.9%, 92.8%, 95.8%, and 96.1% with classical approaches and 84.7%, 87.2%, and 60% with RNNs. The best overall recognition rate of 96.1% was found for SVM+db2-wavelet-features combination. For individual characters, recognition rates obtained were between 80% to 100% and overall accuracy for different subsets was between 88.8% to 100% for SVM+db2-wavelet-features combination. We have followed the segmentation-based approach which requires extraction of half forms of characters from the ligatures. The data was actually obtained in segmented form from the users. Research on segmentation of ligatures into half form characters is also being carried out in parallel to this work. RNNs promise of end-to-end recognition capability was also explored but was found to yield inferior results as compared to the classical feature-based approaches of SVM and ANN. The results with RNNs may be improved if more data is added to the database. In future, with the increased size of database, other deep learning methods like deep belief networks and convolutional neural networks may be employed. Other kinds of features may also be explored.

Acknowledgement

The authors are thankful to the Higher Education Commission (HEC) of Pakistan for funding this work.

References

1. Ghods, V. and Sohrabi, M.K. "Online Farsi Handwritten Character Recognition Using Hidden Markov Model", *Journal of Computers*, **11**, (2016).
2. Mahasukhon, P., Mousavinezhad, H., and Song, J.Y. "Hand-printed english character recognition based on fuzzy theory", *2012 IEEE International Conference on Electro/Information Technology*, pp. 1-4 (2012). ISSN: 2154-0357, DOI:10.1109/EIT.2012.6220772
3. Khodadad, I., Sid-Ahmed, M., and Abdel-Raheem, E. "Online Arabic/Persian character recognition using neural network classifier and DCT features", *IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*, (2011).
4. Yao, C. and Cheng, G. "Approximative bayes optimality linear discriminant analysis for chinese handwriting character recognition", *Neurocomputing*, **207**, pp. 346-353 (2016). ISSN: 0925-2312, DOI:https://doi.org/10.1016/j.neucom.2016.05.017. URL: http://www.sciencedirect.com/science/article/pii/S0925231216303551
5. Xiang-Dong, Z., Da-Han, W., Feng, T., Cheng-Lin, Liu., S.M.I., and Masaki, Nakagawa., M.I. "Handwritten Chinese/Japanese text recognition using semi-Markov conditional random fields", *IEEE Transactions on pattern analysis and machine intelligence*, **35**, (2013).
6. Peng, L., Liu, C., Ding, X., Jin, J., Wu, Y., Wang, H., and Bao, Y. "Multi-font printed mongolian document recognition system", *International Journal on Document Analysis and Recognition (IJDAR)*, **13**(2), pp. 93-106 (2010). ISSN 1433-2825, DOI:10.1007/s10032-009-0106-8
7. Rao Kunte, R.S. and Sudhaker Samuel, R.D. "On-line character recognition for handwritten Kannada characters using wavelet features and neural classifier", *IETE Journal of Research*, **46**, (2000).
8. Tagougui, N., Kherallah, M., and Alimi, A.M. "Online arabic handwriting recognition: a survey", *International Journal on Document Analysis and Recognition (IJDAR)*, **16**(3), pp. 209-226 (2013). ISSN 1433-2825 DOI:10.1007/s10032-012-0186-8 URL: http://dx.doi.org/10.1007/s10032-012-0186-8

9. Harouni, M., Mohamad, D., Shafry, M.M.R., Halawani, S.M., and Afzali, M. "Handwritten Arabic character recognition based on minimal geometric features", *International Journal of Machine Learning and Computing*, **2**, (2012).
10. Kherallah, M., Bouri, F., and Alimi, A. "On-line arabic handwriting recognition system based on visual encoding and genetic algorithm", *Engineering Applications of Artificial Intelligence*, **22**(1), pp. 153-170 (2009). ISSN 0952-1976 DOI: https://doi.org/10.1016/j.engappai.2008.05.010
URL: http://www.sciencedirect.com/science/article/pii/S0952197608001176
11. Kherallah, M., Haddad, L., Alimi, A.M., and Mitiche, A. "On-line handwritten digit recognition based on trajectory and velocity modeling", *Pattern Recognition Letters*, **29**(5), pp. 580-594 (2008). ISSN 0167-8655 DOI:https://doi.org/10.1016/j.patrec.2007.11.011
URL: http://www.sciencedirect.com/science/article/pii/S0167865507003662
12. Mahmoud, S.A. and Mahmoud, A.S. "Arabic character recognition using modified fourier spectrum (MFS) vs, fourier descriptors", *Cybernetics and Systems: An International Journal*, **40**(3), pp. 189-210 (2009). DOI:10.1080/01969720802714758
13. Alimi, A.M. "An evolutionary neuro-fuzzy approach to recognize on-line arabic handwriting", *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, **1**, pp. 382-386 (1997). DOI:10.1109/ICDAR.1997.619875
14. Hamdani, M., Abed, H.E., Kherallah, M., and Alimi, A.M. "Combining multiple hmms using on-line and off-line features for off-line arabic handwriting recognition", *2009 10th International Conference on Document Analysis and Recognition*, pp. 201-205 (2009). ISSN 1520-5363, DOI:10.1109/ICDAR.2009.40.
15. "Telecom indicators", [Online; accessed 14-June-2016] (2016).
URL: http://www.pta.gov.pk/index.php?Itemid=599
16. Baloch, F. "Telecom sector: Pakistan to have 40 million smartphones by end of 2016", .
URL: http://tribune.com.pk/story/953333/telecom-sector-pakistan-to-have-40-million-smartphones-by-end-of-2016/
17. Javed, S.T., Hussain, S., Maqbool, A., Asloob, S., Jamil, S., and H., M. "Segmentation free Nastalique Urdu OCR", *World Academy of Science, Engineering and Technology*, **4**, (2010).
18. Satti, D.A. and Saleem, K. "Complexities and implementation challenges in offline Urdu Nastaliq OCR", *Conference on Language & Technology 2012 (CLT12), University of Engineering & Technology(UET), Lahore, Pakistan*, (2012).
19. Naz, S., Hayat, K., Razzak, M.I., Anwar, M.W., Madani, S.A., and Khan, S.U. "The optical character recognition of Urdu-like cursive scripts", *Pattern Recognition, Elsevier*, **47**, pp. 1229-1248 (2014).
20. Malik, S. and Khan, S.A. "Urdu online handwriting recognition", *IEEE International Conference on Emerging Technologies*, (2005).
21. Shahzad, N., Paulson, B., and Hammond, T. "Urdu Qaeda: recognition system for isolated Urdu characters", *IUI Workshop on Sketch Recognition*, (2009).
22. Haider, I. and Khan, K.U. "Online recognition of single stroke handwritten Urdu characters", *IEEE 13th International Multitopic Conference (INMIC2009)*, (2009).
23. Khan, K.U. and Haider, I. "Online recognition of multi-stroke handwritten Urdu characters", *Image Analysis and Signal Processing (IASP)*, (2010).
24. Shabbir, S. and Siddiqi, I. "Optical Character Recognition System for Urdu Words in Nastaliq Font", *International Journal of Advanced Computer Science and Applications*, **7**(5), pp. 567-576 (2016).
25. Hussain, M. and Khan, M.N. "Online Urdu ligature recognition using spatial temporal neural processing", *IEEE International Multitopic Conference (INMIC05)*, (2005).
26. Husain, S.A., Sajjad, A., , and Anwar, F. "Online Urdu character recognition system", *IAPR Machine Vision Applications (MVA2007), Conference on*, (2007).
27. Razzak, M.I., Anwar, F., Hussain, S.A., Belaid, A., and Sher, M. "HMM and fuzzy logic A hybrid approach for online Urdu script-based languages' character recognition", *Knowledge-Based Systems, Elsevier*, **23**, (2010).
28. Razzak, M.I., Hussain, S.A., Abdulrahman, A.M., and Khan, M.K. "Bio-inspired multilayered and multilanguage arabic script character recognition system", *International Journal of Innovative Computing Information and Control*, **8**(4), pp. 2681-2691 (2012).
29. Pal, U. and Sarkar, A. "Recognition of printed Urdu script", *7th International Conference on Document Analysis and Recognition (ICDAR'03)*, (2003). DOI:0-7695-1960-1/03.
30. Ahmad, Z., Orakzai, J.K., Shamsher, I., and Adnan, A. "Urdu Nastaleeq optical character recognition", *Internation Journal of Computer, Information, Systems and Control Engineering*, **1**(8), (2007).
31. Lehal, G.S. "Choice of recognizable unit for Urdu OCR", *Workshop on Document Analysis and Recognition (DAR12)*, (2012). DOI:10.1145/2432553.2432569
32. Zaman, S., Slany, W., and Saahito, F. "Recognition of segmented Arabic/Urdu characters using pixel values as their features", *ICCIT*, (2012).
33. Javed, S.T. and Hussain, S. "Segmentation based Urdu Nastalique OCR", *18th Iberoamerican Congress (CIARP2013)*, pp. 41-49 (2013). DOI:10.1007/978-3-642-41827-3_6
34. Naz, S., Umar, A.I., Bin Ahmed, S., Shirazi, S.H., Razzak, M.I., and Siddiqi, I. "An OCR system for printed Nasta'liq script: A segmentation based approach", *IEEE 17th International, Multi-Topic Conference (INMIC'2014)*, pp. 255-259 (2014). DOI:10.1109/INMIC.2014.7097347
35. Naz, S., Arif, I.U., Ahmad, R., Saad, B.A., Shirazi, S.H., Siddiqi, I., and Razzak, M.I. "Offline cursive Urdu-Nastaliq script recognition using multidimensional recurrent neural networks", *Neurocomputing*, **177**, pp. 228-241 (2016).
36. Razzak, M.I., Sher, M., and Hussain, S.A. "Locally baseline detection for online arabic script based languages character recognition", *International Journal of the Physical Sciences*, **5**(7), pp. 955-959 (2010).
37. Razzak, M.I., Hussain, S.A., Khan, M.K., and Muhammad, S. "Handling diacritical marks for online arabic script based languages character recognition using fuzzy c-mean clustering and relative position", *Information- an International Interdisciplinary Journal*, **14**(1), pp. 157-165 (2011).
38. Razzak, M.I., Husain, S.A., Mirza, A.A., and Belaid, A. "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*, **85**, (2012).
39. Safdar, Q. and Khan, K.U. "Online Urdu Handwritten Character Recognition: Initial Half Form Single Stroke Characters", *12th International Conference on Frontiers of Information Technology*, pp. 292-297 (2014). DOI:10.1109/FIT.2014.61
40. Patel, D.K., Som, T., Yadav, S.K., and Singh, M.K. "Handwritten character recognition using multiresolution technique and euclidean distance metric", *Journal of Signal and Information Processing*, **3**, pp. 208-214 (2012).
41. Wei, W., Ming, L., Weina, G., Dandan, W., and Jing, L. "A new mind of wavelet wransform for handwritten Chinese character recognition", *Second International Conference on Instrumentation, Measurement, Computer, Communication and Control (IMCCC)*, (2012).
42. Aburas, A. and Rehiel, S.M.A. "Off-line omni-style handwriting Arabic character recognition system based on wavelet Compression", *Arab Research Institute in Sciences & Engineering ARISER*, **3**(4), pp. 123-135 (2007). ISSN 1994-3253
43. Mowlaei, A., Faez, K., and Haghighat, A.T. "Feature extraction with wavelet transform for recognition of isolated handwritten Farsi/Arabic characters and numerals", *IEEE 13th Workshop on Neural Networks for Signal Processing, NNSP'03*, pp. 547-554 (2003). ISSN 1089-3555 DOI:10.1109/NNSP.2003.1318054
44. Jenabzade, M.R., Azmi, R., B., P., and Shirazi, S. "Two methods for recognition of handwritten Farsi characters", *International Journal of Image Processing (IJIP)*, **5**, (2011).
45. Singh, P. and Budhiraja, S. "Handwritten Gurmukhi character recognition using wavelet transform", *International Journal of Electronics, Communication & Instrumentation Engineering Research and Development*, **2**, (2012).
46. Primekumar, K.P. and Idiculla, S.M. "On-line Malayalam handwritten character recognition using wavelet transform and SFAM", *3rd International Conference on Electronics Computer Technology (ICECT)*, **1**, (2011).
47. Abed, H.E., Märgner, V., Kherallah, M., and Alimi, A.M. "Icdar 2009 online arabic handwriting recognition competition", *2009 10th International Conference on Document Analysis and Recognition*, pp. 1388-1392 (2009). ISSN 1520-5363 DOI:10.1109/ICDAR.2009.284

48. Zhang, X.Y., Yin, F., Zhang, Y.M., Liu, C.L., and Bengio, Y. “Drawing and recognizing chinese characters with recurrent neural network”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **40**(4), pp. 849-862 (2018). ISSN 0162-8828 DOI: 10.1109/TPAMI.2017.2695539
49. Graves, A. *Supervised sequence labelling with recurrent neural networks*, Ph.D. thesis, Technical University Munich, (2008). URL: http://d-nb.info/99115827X
50. Javed, S.T. and Hussain, S. “Improving Nastaliq specific pre-recognition process for Urdu OCR”, *13th IEEE International Multitopic Conference*, (2009).
51. Abandah, G.A. and Jamour, F.T. “Recognizing handwritten arabic script through efficient skeleton-based grapheme segmentation algorithm”, *10th International Conference on Intelligent Systems Design and Applications*, (2010).
52. Wahi, A., Sundaramurthy, S., and Poovizhi, P. “Recognition of handwritten Tamil characters using wavelet”, *International Journal of Computer Science & Engineering Technology (IJCSET)*, **5**, (2014).
53. Jaeger, S., Manke, S., Reichert, J., and Waibel, A. “Online handwriting recognition: the NPen++ Recognizer”, *International Journal of Document Analysis and Recognition, IJDAR*, (2001).
54. Al-Hassani, M.D. “Optical character recognition system for multifont English texts using DCT and Wavelet Transform”, *International Journal of Computer Engineering and Technology (IJCET)*, **4**(6), (2013). ISSN 0976-6367(Print), (Online)
55. Mallat, S. *A wavelet tour of signal processing, The Sparse Way*, (Academic Press Elsevier Inc. San Dieago), (2008).
56. Gonzalez, R.C. and Woods, R.E. *Digital Image Processing (3rd Edition)*, (Prentice-Hall, Inc., Upper Saddle River, NJ, USA), (2006). ISBN 013168728X.
57. Amar, C.B., Zaied, M., and Alimi, A. “Beta wavelets. synthesis and application to lossy image compression”, *Advances in Engineering Software*, **36**(7), pp. 459-474 (2005). ISSN 0965-9978 URL: http://www.sciencedirect.com/science/article/pii/S0965997805000116
58. Murru, N. and Rossini, R. “A bayesian approach for initialization of weights in backpropagation neural net with application to character recognition”, *Neurocomputing*, **193**, pp. 92-105 (2016). ISSN 0925-2312 DOI:https://doi.org/10.1016/j.neucom.2016.01.063 URL: http://www.sciencedirect.com/science/article/pii/S0925231216001624
59. Prieto, A., Prieto, B., Ortigosa, E.M., Ros, E., Pelayo, F., Ortega, J., and Rojas, I. “Neural networks: An overview of early research, current frameworks and new challenges”, *Neurocomputing*, **214**, pp. 242-268 (2016). ISSN 0925-2312 DOI: https://doi.org/10.1016/j.neucom.2016.06.014 URL: http://www.sciencedirect.com/science/article/pii/S0925231216305550
60. Shamsher, I., Ahmad, Z., Orakzai, J.K., and Adnan, A. “OCR for printed Urdu script using feed forward neural network”, *World Academy of Science, Engineering and Technology*, **1**, (2007).
61. Salameh, W.A. and Otair, M.A. “Online handwritten character recognition using an optical backpropagation neural network”, *Issues in Informing Science and Information Technology*, **3**, (2005).
62. Theodoridis, S. and Koutroumbas, K. *Pattern Recognition, 4th Edition*, (Academic Press), (2008). ISBN 1597492728, 9781597492720
63. John, S.T. and Nello, C. *Kernel Methods for Pattern Analysis*, (Cambridge University Press), (2004). ISBN 0521813972
64. Chang, C.C. and Lin, C.J. “LIBSVM: A library for support vector machines”, *ACM Transactions on Intelligent Systems and Technology*, **2**, (2011).
65. Lipton, Z.C. “A critical review of recurrent neural networks for sequence learning”, *CoRR*, (2015). URL: http://arxiv.org/abs/1506.00019
66. Graves, A. “Rnnlib: A recurrent neural network library for sequence learning problems”, http://sourceforge.net/projects/rnnl/.
67. Jannoud, I.A. “Automatic Arabic handwritten text recognition system”, *American journal of applied sciences*, **4**, (2007).
68. Asiri, A. and Khorsheed, M.S. “Automatic processing of handwritten Arabic forms using neural networks”, *World academy of science, engineering and technology*, (2005). ISSN 1307-6884
69. Broumandnia, A., Shanbehzadeh, J., and Varnoosfaderani, M.R. “Persian/Arabic handwritten word recognition using M-band packet wavelet transform”, *Image Vision Computing*, **26**, (2008).

Biographies

Quara-tul-Ain Safdar is a PhD scholar at PIEAS, Islamabad, IR Pakistan. She has received her MS. degree in Computer Science from University of Central Punjab, Lahore, IR, Pakistan in 2005. Her research interests include Pattern recognition, and Urdu handwriting recognition.

Kamran Ullah Khan is with Deptt. of Elect. Engg., Pakistan Inst. of Engg. and Applied Sciences (PIEAS), Islamabad, IR Pakistan. He got his PhD from Tsinghua University, Beijing, in 2008, MS from Quaid-e-Azam University, Islamabad, in 2000, and B.Sc. Elect. Engg. from UET Peshawar in 1998.

Liangrui PENG is currently an associate professor at the Department of Electronic Engineering, Tsinghua University, Beijing, China. Her research interests include multilingual document image recognition and understanding, machine learning, pattern recognition. She is a member of the IEEE and of the ACM.

Tables Figures and Captions

List of Figures

List of Tables

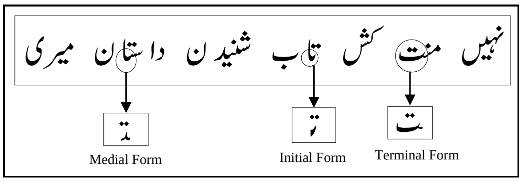


Figure 1: Urdu character ‘ت’ in half forms.

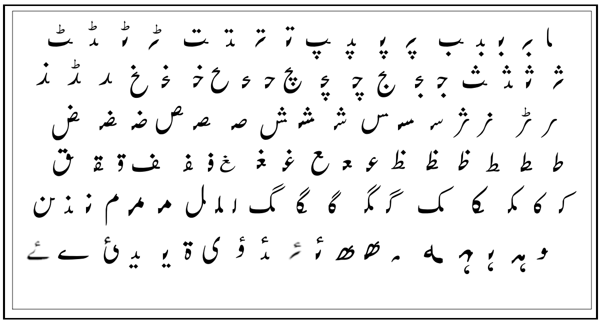


Figure 2: All Urdu characters in all half forms.

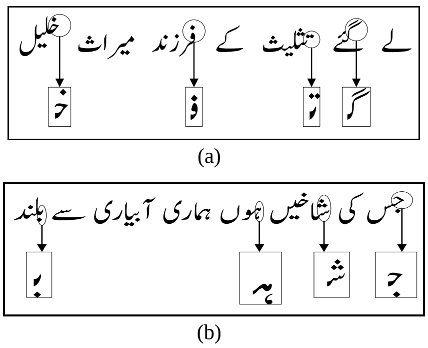


Figure 3: Examples: use of initial half form multistroke characters.

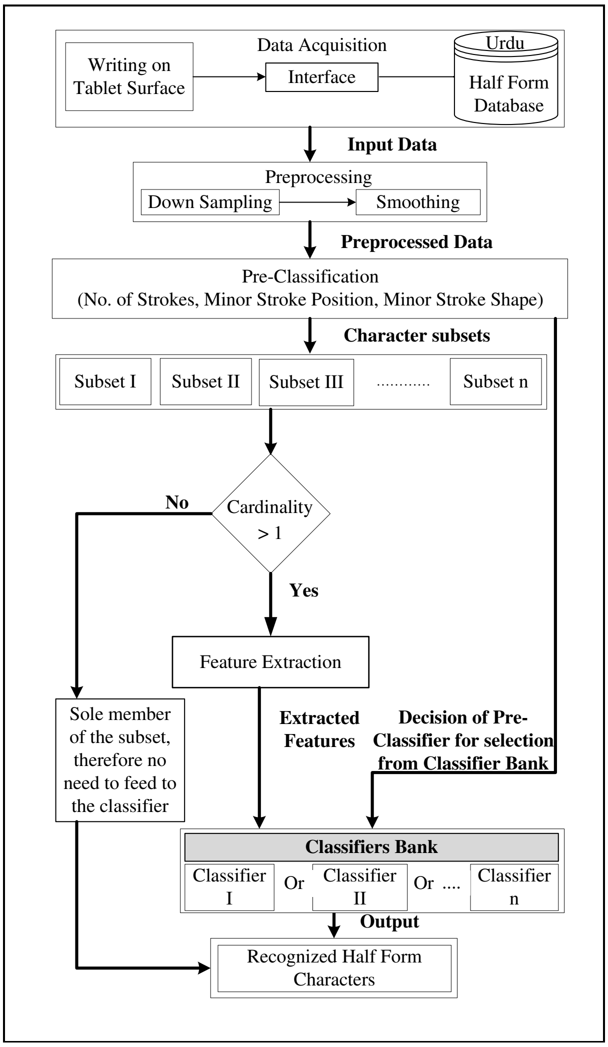


Figure 4: Block diagram of the proposed Online Urdu Character Recognition System: from data acquisition to preprocessing to pre-classification to feature extraction to final classification.

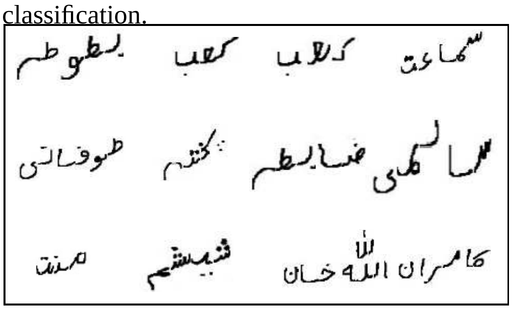


Figure 5: Examples of words composed from (segmented) handwritten half form characters.

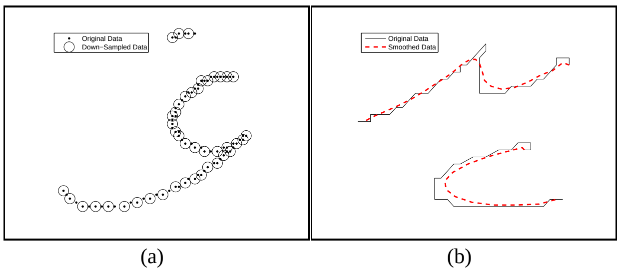


Figure 6: Preprocessing: (a) Resampling and Down-sampling of character ‘ﻐ’(b) Smoothing of character ‘ﺠ’.

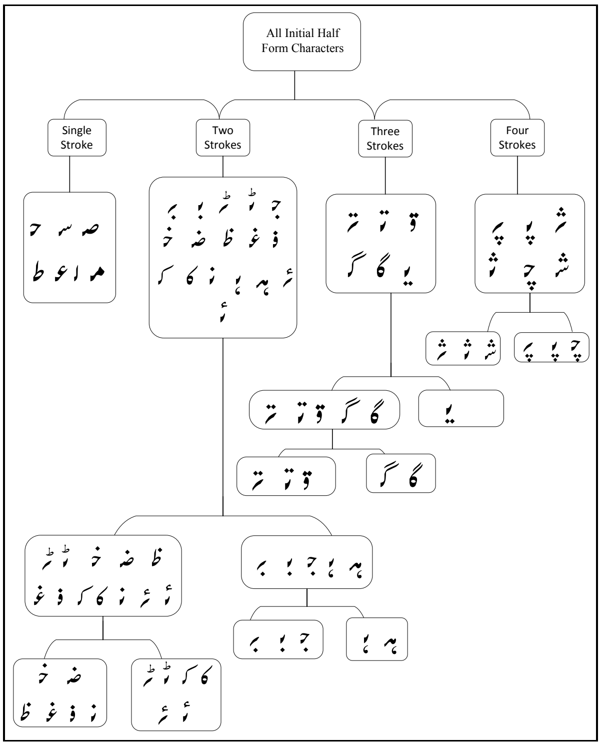
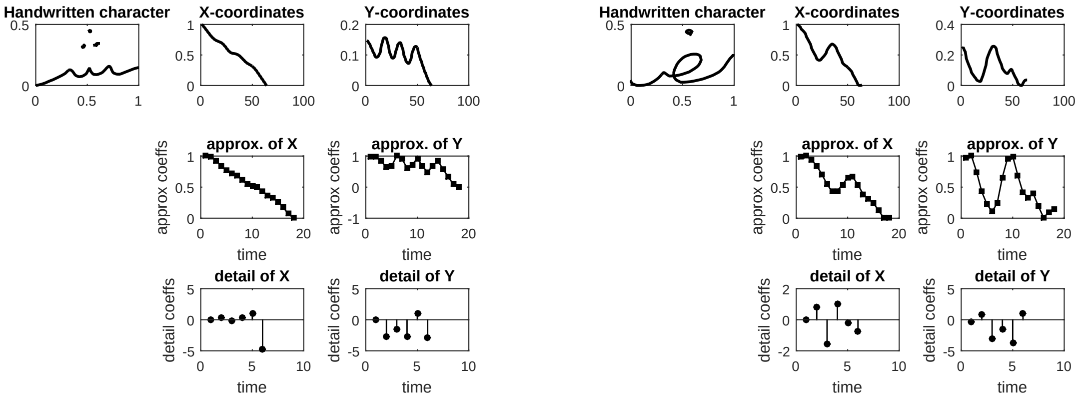
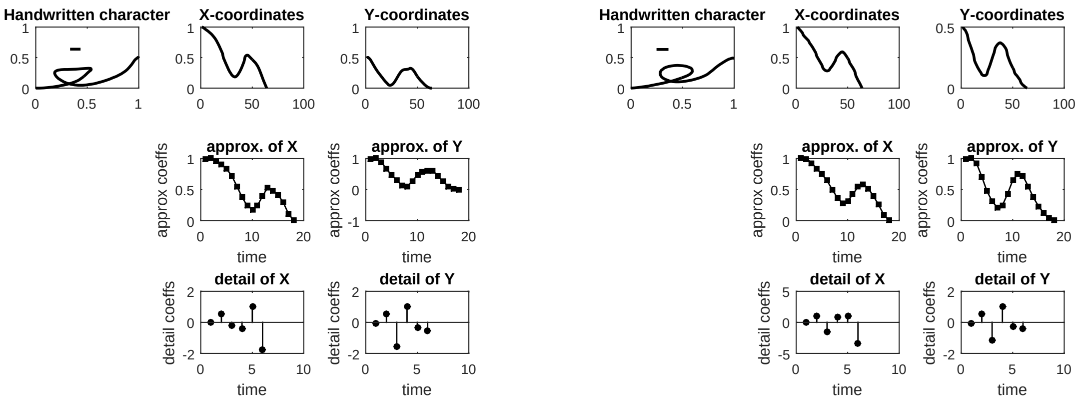


Figure 7: Pre-classification of initial half forms on the basis of stroke count, position and shape of minor strokes.



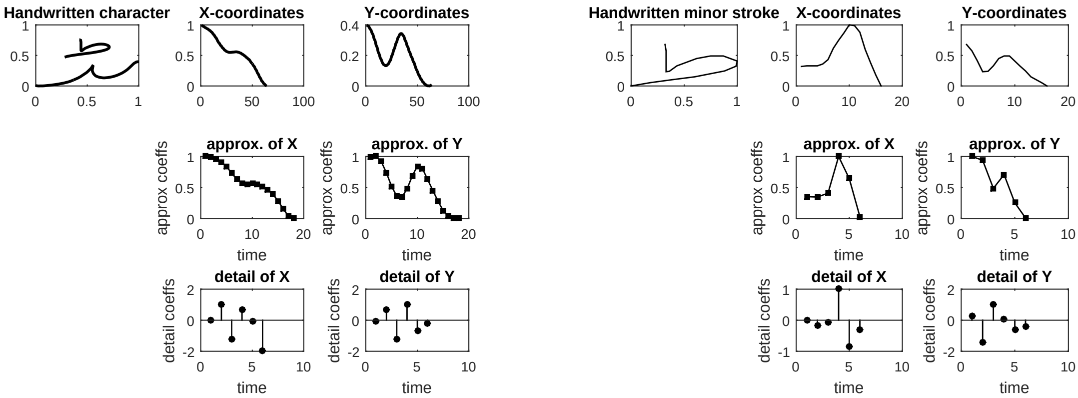
(a) (b)

Figure 8: (a) Top row shows character *sheen* in medial form, and $x(t)$ and $y(t)$ of its major stroke. Second and third rows show level-2 db2 wavelet approximation, and level-4 db2 wavelet detail coefficients of $x(t)$ and $y(t)$ respectively.
(b) Top row shows character *zwad* in medial form, and $x(t)$ and $y(t)$ of its major stroke. Second and third rows show level-2 db2 wavelet approximation, and level-4 db2 wavelet detail coefficients of $x(t)$ and $y(t)$ respectively.



(a) (b)

Figure 9: (a) Top row shows character *ghain* in medial form, and $x(t)$ and $y(t)$ of its major stroke. Second and third rows show level-2 db2 wavelet approximation, and level-4 db2 wavelet detail coefficients of $x(t)$ and $y(t)$ respectively.
(b) Top row shows character *fay* in medial form, and $x(t)$ and $y(t)$ of its major stroke. Second and third rows show level-2 db2 wavelet approximation, and level-4 db2 wavelet detail coefficients of $x(t)$ and $y(t)$ respectively.



(a) (b)

Figure 10: (a) Top row shows character *Tay* in medial form, and $x(t)$ and $y(t)$ of its major stroke. Second and third rows show level-2 db2 wavelet approximation, and level-4 db2 wavelet detail coefficients of $x(t)$ and $y(t)$ respectively.
(b) Top row shows the minor stroke of *Tay* and its $x(t)$ and $y(t)$ coordinates. Second and third rows show the level-2 db2 wavelet approximation and level-2 db2 wavelet detail coefficients of $x(t)$ and $y(t)$ of minor stroke respectively.

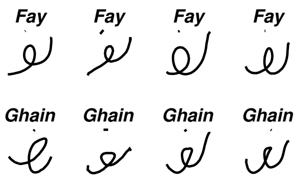


Figure 11: Handwritten samples of ـفـ (*Fay*) and ـغـ (*Ghain*).
The ـغـ (*Ghain*s) are confusingly similar to the ـفـ (*Fay*s).

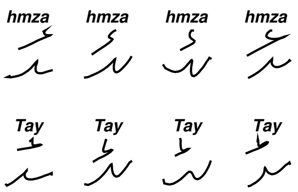


Figure 12: Handwritten samples of ـئـ (*hamza*) and ـٹـ (*Tay*).
The ـٹـ (*Tay*s) are confusingly similar to the ـئـ (*hamza*s).

Table 1: Comparison of online Urdu handwritten character recognition (proposed) with Arabic & Persian work.

Authors	Type	Character Set x Samples	Language	Features	Classification	Participants	Accuracy (%)
Proposed work	Multistroke characters *(IHF,MHF, THF)	77x100	Urdu	Structural, Wavelet Coefficients	BPNN, SVM	100	87.5% to 100%
I. A. Jannoud [67]	Isolated,*IHF, MHF, THF	not reported	Arabic	Discrete Wavelet Transformation	MLE	Not reported	99% for isolated, more than 90% for *IHF and THF, 91% for MHF
Asiri and Khorsheed [68]	Isolated,*IHF, MHF, THF	30x500	Arabic	Haar Wavelet Transform	ANN	Not reported	for 3 different sets of wavelet coefficients: 74%, 82%, and 88%
A. Mowlaei et al. [43]	Isolated	32x190	Persian	Haar Wavelets	MLP	200	92.3
Aburas and Rehiel [42]	Isolated	28x48	Arabic	Wavelet coeffocients	Codebook serach & Euclidean distance measure	48	45.8% to 97.9%
Broumandnia et al. [69]	Words	100x8 rotations of each word	Persian	2D M-band wavelet packets	Mahalanobis classifier	12	65% to 96%
M. R. Jenabzade et al. [44]	Isolated	33x200	Persian	Haar Wavelets	MLP	not reported	86.3

*Initial Half Form, Medial Half Form, Terminal Half Form

Table 2: Pre-classification of Urdu character set. The encircled numbers indicate the cardinality of final stage subsets that could be obtained with the help of the proposed pre-classifier

	subset	Number of Characters in subset	Division on Minor stroke position w.r.t major stroke (Above/Below)	Number of characters in sub- subset	Division on Diacritic Type (*dot*/*other-than-dot*)	Number of characters in sub-sub-subset
Initial half forms (36 characters)	Single-Stroke	**7**	×	×	×	×
	Two-Stroke	17	Above	12	*dot*	**6**
					other-than-dot	**6**
			Below	5	*dot*	**3**
					other-than-dot	**2**
	Three-Stroke	6	Above	5	*dot*	**3**
					other-than-dot	**2**
			Below	**1**	×	×
	Four-Stroke	6	Above	3	*dot*	**3**
					other-than-dot	×
			Below	3	*dot*	**3**
					other-than-dot	×
Medial half forms (30 characters)	Single-Stroke	**8**	×	×	×	×
	Two-Stroke	13	Above	10	*dot*	**6**
					other-than-dot	**4**
			Below	3	*dot*	**2**
					other-than-dot	**1**
	Three-Stroke	5	Above	4	*dot*	**2**
					other-than-dot	**2**
			Below	**1**	×	×
	Four-Stroke	4	Above	2	*dot*	**2**
					other-than-dot	×
			Below	2	*dot*	**2**
					other-than-dot	×
Terminal half forms (42 characters)	Single-Stroke	**16**	×	×	×	×
	Two-Stroke	17	Above	16	*dot*	**9**
					other-than-dot	**7**
			Below	**1**	×	×
	Three-Stroke	4	Above	4	*dot*	**3**
					other-than-dot	**1**
			Below	×	×	×
	Four-Stroke	5	Above	4	*dot*	**4**
					other-than-dot	×
			Below	**1**	×	×

Table 3: Characters recognized at pre-classification stage and don't require any further classification

Target Group	Subset	Character	Recognition Rate (%)
Initial half forms	3-stroke dot below	[illegible]	100
Medial half forms	2-stroke other below	ـجـ	100
Medial half forms	3-stroke dot below	[illegible]	100
Terminal half form	2-stroke dot below	ـب	100
Terminal half form	3-stroke other above	ـگ	100
Terminal half form	4-stroke dot below	ـپ	100

Table 4: ANN Configurations (trained using wavelet *db2* approximation and detailed coefficients).

Target Group	No. of hidden layers	Neurons in hidden layer 1	Neurons in hidden layer 2	Recognition rate (%)
ANN Configuration: Initial Half Forms				
2-stroke *dot* Above	2	9	6	90.2
2-stroke *other*- Above	2	9	6	87.7
2-stroke *dot* Below	1	1	-	94.4
2-stroke *other*- Below	1	1	-	97.5
3-stroke *dot* Above	2	2	3	97.7
3-stroke *other*- Above	2	2	3	98.3
4-stroke *dot* Above	2	6	3	89.4
4-stroke *dot* Below	2	4	3	89
ANN Configuration: Medial Half Forms				
2-stroke *dot* Above	2	9	9	81.6
2-stroke *other*- Above	2	8	6	91.6
2-stroke *dot* Below	1	1	-	99.1
3-stroke *dot* Above	2	3	3	98.3
3-stroke *other*- Above	1	2	-	97.5
4-stroke *dot* Above	2	4	2	97.5
4-stroke *dot* Below	2	4	2	100
ANN Configuration: Terminal Half Forms				
2-stroke *dot* Above	2	7	9	93.3
2-stroke *other*- Above	2	7	7	95.7
3-stroke *dot* Above	1	2	-	95.5
4-stroke *dot* Above	2	4	2	97.9

Table 5: Recognition rates for each subset (cardinality >1) of multistroke half form Urdu characters obtained from the pre-classifier. Results obtained using ANN, SVM, and RNN using different features are presented for comparison.

Half Form	Character Subset	Number of characters in subset	Recognition Rate (%) using **ANN**		Recognition Rate (%) using **SVM**		Recognition Rate (%) using **RNN**	
			Structural Features	Wavelet (db2)	Wavelet (db2)	Wavelet (bior1.3)	Single LSTM hidden layer of size 100	Multi LSTM hidden layers of varying sizes
Initial half forms (8 subsets, 28 chars)	2-Stroke *dot* Above	6	81.3	90.2	99.1	98	78	84.7
	2-Stroke *other-than-dot* Above	6	76.3	87.7	91.9	87.2	72	73.3
	2-Stroke *dot* Below	3	92.2	94.4	97.2	97.7	88.7	94
	2-Stroke *other-than-dot* Below	2	90	97.5	98.3	96.6	79	90
	3-Stroke *dot* Above	3	88.8	97.7	94.4	95.5	88.7	91.3
	3-Stroke *other-than-dot* Above	2	99.1	98.3	100	100	96	97
	4-Stroke *dot* Above	3	77.7	89.4	88.8	87.7	83.3	85.3
	4-Stroke *dot* Below	3	88.8	89	92.7	93.3	84.7	88.7
Medial half forms (7 subsets, 20 chars)	2-Stroke *dot* Above	6	58	81.6	93.6	91.3	74	74
	2-Stroke *other-than-dot* Above	4	80.4	91.6	93.3	94.1	73	73.5
	2-Stroke *dot* Below	2	99	99.1	98.3	100	94	96
	3-Stroke *dot* Above	2	95	98.3	95	98.3	81	90
	3-Stroke *other-than-dot* Above	2	94.1	97.5	95.8	97.5	84	94
	4-Stroke *dot* Above	2	87.5	97.5	95.8	95.8	86	86
	4-Stroke *dot* Below	2	97.5	100	100	100	89	99
Terminal half forms (4 subsets, 23 chars)	2-Stroke *dot* Above	9	66.6	93.3	96.7	97.2	92	92
	2-Stroke *other-than-dot* Above	7	82.6	95.7	99	99.2	89.3	91.8
	3-Stroke *dot* Above	3	93.3	95.5	99.4	100	96	99.3
	4-Stroke *dot* Above	4	94.1	97.9	99.6	99.1	96.5	97.5
		Overall Accuracy	81.9	92.9	96.1	95.8	84.7	87.2

Table 6: Confusion Matrix for initial half forms 2-stroke characters with *other-than-dot* diacritic above the major stroke. Overall accuracy for this subset is 91.9%

	ئـ	ئـ	ک	ھ	ٹ	ٹـ	Unknown	
ئـ	57	2	0	0	1	0	0	60
ئـ	3	52	1	0	4	0	0	60
ک	0	0	60	0	0	0	0	60
ھ	0	0	1	59	0	0	0	60
ٹ	0	4	0	0	49	7	0	60
ٹـ	2	2	0	0	2	54	0	60
	62	60	62	59	56	61	0	

Table 7: Confusion Matrix for medial half form 2-stroke characters with *dot* diacritic above the major stroke. Overall accuracy for this subset is 93.6%

	ـفـ	ـغـ	ـنـ	ـذ	ـظـ	ـضـ	Unknown	
ـفـ	51	5	0	1	0	3	0	60
ـغـ	7	51	0	1	0	1	0	60
ـنـ	0	0	60	0	0	0	0	60
ـذ	1	1	0	57	1	0	0	60
ـظـ	0	0	1	0	59	0	0	60
ـضـ	0	0	0	0	1	59	0	60
	59	57	61	59	61	63	0	

Table 8: Confusion Matrix for medial half forms 2-stroke characters with *other-than-dot* diacritic above the major stroke. Overall accuracy for this subset is 93.3%

	ـئـ	ـکـ	ـھـ	ـٹـ	Unknown	
ـئـ	56	1	0	3	0	60
ـکـ	0	60	0	0	0	60
ـھـ	0	3	57	0	0	60
ـٹـ	9	0	0	51	0	60
	65	64	57	54	0	

Table 9: Confusion Matrix for terminal half forms 2-stroke characters with *dot* diacritic above the major stroke. Overall accuracy for this subset is 96.7%

	ف	غ	ج	خ	ن	ذ	ز	ظ	ض	Unknown	
ف	58	0	0	0	0	0	0	0	2	0	60
غ	0	54	0	6	0	0	0	0	0	0	60
ج	0	1	59	0	0	0	0	0	0	0	60
خ	0	4	0	56	0	0	0	0	0	0	60
ن	0	0	0	0	60	0	0	0	0	0	60
ذ	0	0	0	0	0	58	2	0	0	0	60
ز	0	0	0	0	0	0	60	0	0	0	60
ظ	0	0	0	1	0	1	0	58	0	0	60
ض	1	0	0	0	0	0	0	0	59	0	60
	59	59	59	63	60	59	62	58	61	0	

Table 10: Confusion Matrix for terminal half forms 4-stroke characters with *dot* diacritic above the major stroke. Overall accuracy for this subset is 99.6%

		چ	ث	ش	ژ	Unknown
چ	60	0	0	0	0	60
ث	0	60	0	0	0	60
ش	0	0	59	1	0	60
ژ	0	0	0	60	0	60
	60	60	59	61	0	