Computers, Materials & Continua
DOI:10.32604/cmc.2020.013715
Article

Tech Science Press

Urdu Ligature Recognition System: An Evolutionary Approach

Naila Habib Khan[1,*], Awais Adnan[1], Abdul Waheed[2,3], Mahdi Zareei[4], Abdallah Aldosary[5] and Ehab Mahmoud Mohamed[6,7]

[1]Department of Computer Science, Institute of Management Sciences, Peshawar, 25000, Pakistan
[2]Department of Information Technology, Hazara University, Mansehra, 21120, Pakistan
[3]School of Electrical and Computer Engineering, Seoul National University, Seoul, 08826, South Korea
[4]Tecnologico de Monterrey, School of Engineering and Sciences, Zapopan, 45201, Mexico
[5]Department of Computer Science, Prince Sattam Bin Abdulaziz University, As Sulayyil, 11991, Saudi Arabia
[6]Electrical Engineering Department, College of Engineering, Prince Sattam Bin Abdulaziz University, Wadi Addwasir, 11991, Saudi Arabia
[7]Electrical Engineering Department, Faculty of Engineering, Aswan University, Aswan, 81542, Egypt
[*]Corresponding Author: Naila Habib Khan. Email: naila.khancs@yahoo.com
Received: 10 August 2020; Accepted: 24 August 2020

Abstract: Cursive text recognition of Arabic script-based languages like Urdu is extremely complicated due to its diverse and complex characteristics. Evolutionary approaches like genetic algorithms have been used in the past for various optimization as well as pattern recognition tasks, reporting exceptional results. The proposed Urdu ligature recognition system uses a genetic algorithm for optimization and recognition. Overall the proposed recognition system observes the processes of pre-processing, segmentation, feature extraction, hierarchical clustering, classification rules and genetic algorithm optimization and recognition. The pre-processing stage removes noise from the sentence images, whereas, in segmentation, the sentences are segmented into ligature components. Fifteen features are extracted from each of the segmented ligature images. Intra-feature hierarchical clustering is observed that results in clustered data. Next, classification rules are used for the representation of the clustered data. The genetic algorithm performs an optimization mechanism using multi-level sorting of the clustered data for improving the classification rules used for recognition of Urdu ligatures. Experiments conducted on the benchmark UPTI dataset for the proposed Urdu ligature recognition system yields promising results, achieving a recognition rate of 96.72%.

Keywords: Classification rules; genetic algorithm; intra-feature hierarchical clustering; ligature recognition; Urdu script

1 Introduction

Urdu is the national language of Pakistan and is written in the Nastalique calligraphic style [1,2]. Urdu script is extremely context-sensitive and cursive [3]. Generally, a sentence is composed of two textual components, i.e., characters and words. However, in the case of Urdu script, there is an added component, sub-word, called a ligature [3,4]. The Urdu informatics, particularly the Optical Character

Recognition (OCR), has been suffering in research and development due to the complexities and segmentation errors associated with its cursive script [5]. This cursive nature of the script leads to numerous challenges such as context-sensitivity, overlapping, nuqtas placement, thickness variation, positioning and diagonality [6–9]. Recently, several efforts have been designated towards the development of an OCR system for Urdu script. However, most of the studies have focused on character-based recognition; the analytical approach requires intensive procedures for character level segmentation. Due to the calligraphic nature of Urdu script, the character-based recognition systems are more complex, challenging and highly prone to errors. During the process of character segmentation, the shape of the characters might be deteriorated by segmenting at wrong segmentation points, leading to lower recognition accuracies. One solution to avoid the overhead of character segmentation in Urdu script recognition systems is to use ligatures for recognition. In ligature recognition systems the whole words/ligatures are recognized instead of individual characters. Consequently, ligature recognition systems omit the intensive steps required for character level segmentation. Similarly, with ligature recognition systems, instead of extracting structural features from the segmented characters, simple geometrical, as well as statistical features, can be extracted from the ligature images. However, most of the existing studies for ligature recognition using the holistic approach have worked with limited datasets, extracted a large number of features or features that are extremely difficult to process, and have low recognition rates. The proposed system uses a Genetic Algorithm (GA) based approach for the recognition of printed Urdu ligatures. GA is a metaheuristic algorithm belonging to a larger class of the Evolutionary Algorithm (EA), based on the idea of the evolution theory given by Holland in 1975 [10]. It is commonly used to provide solutions to various optimization and search problems, in machine learning and in the research field for modeling different phenomena [11].

This research study observes the following objectives for developing of the ligature recognition system for cursive Urdu script. (1) Use the processes of pre-processing, segmentation and feature extraction to generate a representation for each ligature against its ground-truth classes. (2) Use intra-feature hierarchical clustering to enable the efficient organization, and generate cluster limits that can be used as threshold values for data representation of ligature observations using classification rules. (3) Use a genetic algorithm for optimization and recognition. The primary contribution of the proposed research is using a genetic algorithm-based approach for ligature recognition, providing the ability to process a large number of ligature classes (3645) for a large dataset (189003) ligatures in a small amount of time and achieving high recognition accuracy. The rest of the research article is organized as follows: Section 2 provides a comprehensive literature review about the works dedicated to Urdu script recognition. Moreover, several studies are reviewed that have used genetic algorithm-based approaches for optimization and/or recognition of text. Next, Section 3 provides a thorough description of the proposed methodology. The methodology is supported by various figures, tables and equations. Following, in Section 4 the dataset is discussed, experiments and results for using the proposed approach are also provided. Finally, Section 5 provides a discussion and Section 6 concludes the research article and additionally suggests several future directions.

2 Related Work

The recognition systems for Urdu text can be broadly divided into two categories based on its segmentation unit i.e., a character or word/ligature, also known as analytical or holistic recognition systems respectively. The analytical recognition systems are further subdivided into two types based on the segmentation process it observes i.e., implicit segmentation and explicit segmentation [3,12,13]. In the past years, efforts have been made for developing Urdu text recognition systems by numerous researchers for analytical recognition systems [14–16] as well as holistic recognition systems [4,17–21].

Similarly, prominent efforts have been made towards general Urdu informatics and natural language processing [1,22–30].

Extensive research has been reported for isolated and cursive Urdu character recognition systems. The complex machine learning architecture of Neural Network was used for training and testing an Urdu OCR system by [31,32] achieving an accuracy of 98.3% and 91.3% respectively. Whereas, no Neural Network was used by [33] and still obtained an accuracy of 97.43%. Akram et al. [34] recognized single character Urdu ligatures, achieving an accuracy of 96% for scanned images and 98% for manual data. An accuracy of 96.2% was achieved using a principal component analysis for recognizing Urdu text [35]. Whereas, in [36], a decision tree was used for the classification of handwritten and printed Urdu characters, reporting a recognition rate of 92.06%. A two-step approach was used for recognition of characters, first, a Kohonen Self-Organizing Map was used to group varying shapes for the identical characters into 33 classes [37]. In the second step, 25 unique features were extracted. For the final recognition, 104 segmented characters were tested on the system. Similarly, traditional k-NN, SVM and HMM were used for classification of about 9262 Urdu ligatures achieving an accuracy of about 98% [38]. An accuracy of 89% was reported for an OCR system developed by Nawaz et al. [39] using a pattern matching technique. Likewise, template matching was used for Urdu script sentences having a 72 font size, a comparison was carried out between the saved templates and the identified objects [40]. SURF was used for the Urdu text recognition system by Khan et al. [41] and the proposed system was tested for a total of 20 newspaper clippings. In [42], a modified tesseract system was reported for the Nastalique Urdu script that outperformed the original tesseract system. The modified system reported accuracy of 97.87% and 97.71% for 14 font size and 16 font size respectively. In [43], an accuracy of 93.4% was reported using a neural network architecture for training various character forms, testing was performed using the real world as well as synthetic images. Similarly, Ahmad et al. [44] used a feed-forward neural network for training 41 characters, 100 samples each, divided into 56 classes. On average, a 70% recognition rate was reported for the suggested Urdu text recognition system. In a research study performed by Ahmed et al. [45], recurrent neural network (RNN) was evaluated for both cursive (Urdu) and the non-cursive (Latin) text using a Bidirectional Long Short-Term Memory (BLSTM). Sequence alignment was observed using a Connectionist Temporal Classification layer. A recognition rate of 88.79% and 88.94% was reported for cursive Urdu text with position information and without position information respectively. An accuracy of 99.1% was reported for Latin script on the UNLV-ISRI dataset. Likewise, in [46], a Bi-directional LSTM was used for the recognition of Urdu text using two approaches. In the first case, shape information was considered and an error rate of 13.6% was observed. Whereas, for the second case, ignoring the shape information an error rate of 5.15% was reported accordingly. A combination of BLSTM and ANFIS method was proposed by Patel et al. [47], reporting an error rate of 5.4%. Similarly, in [16], an accuracy of 96.40% was achieved for Urdu Nastalique text recognition using statistical features and a multi-dimensional long short-term memory recurrent neural network (MDLSTM RNN) with Connectionist Temporal Classification (CTC) layer. A system was developed by [14] that used a Convolution Neural Network (CNN) for automatic feature extraction, an MDLSTM was used for classification and recognition. The recommended system achieved a recognition rate of 98.12% for the benchmark UPTI dataset. In another study, MDLSTM and statistical features were used for the recognition of printed Urdu script achieving a recognition rate of 94.97% [15]. Likewise, a model using zoning features and a 2DLSTM was evaluated for recognition of Urdu script, obtaining a recognition accuracy of 93.39% [48]. In [49], a 98% recognition rate was reported for unconstrained printed Urdu Nastalique text using a Multi-Dimensional Long Short Term Memory (MDLSTM) model and CTC as the output layer.

Bio-inspired, genetic algorithms are widely used for feature optimization problems, finding an optimal solution in the least time from the problem space [50,51]. Recently, the genetic algorithm has gained popularity for pattern recognition tasks [52]. The genetic algorithm also has other numerous applications

[53–57], successfully used for license plate recognition [53], intrusion detection system [54,55] and segmentation [56,57]. Genetic algorithms have also been widely used for optical character recognition systems. Summary of the notable contributions for genetic algorithm-based OCR systems is given in Tab. 1. The authors in [58], proposed a system for recognizing online cursive Arabic handwriting. A fuzzy neural network was used for recognition, whereas the best combination of characters was selected using a genetic algorithm. A feature subsets selection method using an enhanced genetic algorithm was proposed by Abandah et al. [59] for Arabic text recognition. It was observed that selecting a subset of features from the character's main body and its secondaries significantly improved the accuracy using SVM, improvement of 15% and 10% for normalized central moments and zernike moments for a 20-feature subset. Genetic algorithm and visual coding were also used for online handwriting recognition of Arabic script by Kherallah et al. [60]. The main contribution of this research paper was based on the conception of an encoding system and a fitness function. The evaluation function was developed using the visual indices similarity. Words from Arabic dataset "LMCA" developed by 24 participants from the same laboratory was used for the evaluation purposes. The final results obtained were very promising, proving that the suggested hybrid method was extremely powerful. Similarly [61], obtained phenomenal recognition rate of 95% for isolated Arabic characters. The unknown character was read from a file, numerous operations were performed on it to make the final recognition effective. Likewise, in [62], a genetic algorithm was used to select the best combination of characters for recognition of online handwritten Arabic text and Beta neuro-fuzzy was used for recognition of the characters. In [63], an approach was presented for an offline Arabic writer identification using a combination of structural and global features. A genetic algorithm was used for feature subset selection, whereas, Support vector machines and multilayer perceptron (MLP) were used as classifiers. The experiments were carried out on a database of 120 samples, achieving about 94% accuracy for MLP. Dealing with a large number of features in OCR systems is not unusual and may increase the computational load of the recognition process [64]. For reducing the unnecessary and redundant features from the recognition process for Farsi script, a genetic algorithm was employed [64]. Lower computational complexity and enhanced recognition rates were reported for the optical recognition system. Kala et al. [65] suggested a method where graphs were generated for each of the 26 capital alphabets of the English language. These graphs were then intermixed to generate new styles using the genetic algorithm. The final recognition was carried out by matching the graph generated by an unknown character image to the graphs generated by the mixing process. A recognition rate of 98.44% was achieved for the recommended method.

3 Proposed Methodology

This is one of the first studies that propose the use of an evolutionary approach for the recognition of offline printed Urdu script. In the initial stages, a text line image is pre-processed. A holistic segmentation approach is then used to segment Urdu text lines into individual ligatures. After segmentation, 15 hand-engineered geometric and statistical features are extracted from the segmented ligature images. These features are then concatenated to form the final feature vector for each ligature image. Later after feature extraction, data points for each of the features are clustered using an intra-feature hierarchical clustering algorithm. This clustered data results in classification rules. Classification rules are used for the representation of the clustered data points against the available ground-truth classes for each of the ligatures. The classification rules generated are encoded in the form of conditional statements. Finally, a Genetic Algorithm (GA) is used for optimization and recognition. The clustered data is optimized, henceforth, the classification rules are optimized. The recognition accuracy for the proposed Urdu ligature recognition system is calculated using the predicted information against the known label information i.e., ground-truth. The overview of the proposed system is given in Fig. 1.

Table 1: Summary of contributions for genetic algorithm-based OCR systems

Study	Text	Methods	Dataset	Accuracy
Alimi [58]	Handwritten Arabic	GA and fuzzy neural network	Training: 2000 characters testing: 100 replications	89%
Abandah et al. [59]	Handwritten Arabic	GA and SVM	28 Arabic characters: 48 samples from 48 different people	Less error
Kherallah et al. [60]	Handwritten Arabic	Visual encoding and genetic algorithm	LMCA	–
Abed et al. [61]	Isolated Arabic	Genetic algorithm	Isolated characters	95%
Alimi [62]	Handwritten Arabic	GA and hierarchical beta neuro-fuzzy system (BNFS)	1000 words = 3000 characters	89.5%
Gazzah et al. [63]	Handwritten Arabic	Genetic algorithm, SVM and MLP	120 samples	SVM: 93.76% MLP: 94.7%
Soryani et al. [64]	Printed Farsi	Genetic algorithm and euclidean distance	1080 images	Enhanced
Kala et al. [65]	Handwritten English	Genetic algorithm	Training: 69 characters test data: 385	98.44%

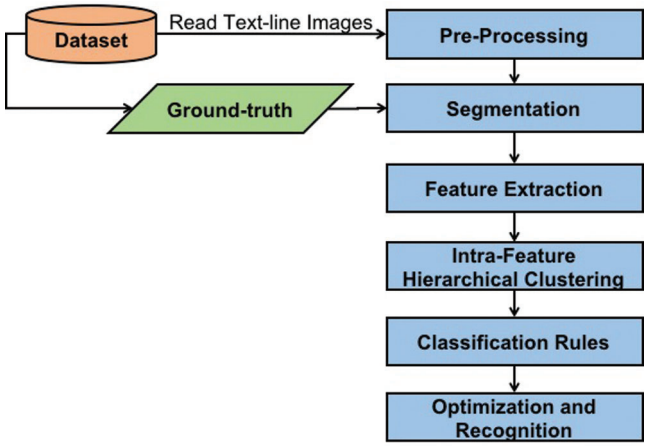

Figure 1: Overview of proposed Urdu ligature recognition system

3.1 Pre-Processing, Segmentation and Feature Extraction

This section observes the settings for the first three stages of the proposed recognition system, namely, pre-processing, segmentation and feature extraction.

- **Pre-processing:** The text line images are pre-processed and connected components having less than 4 pixels are regarded as noise and removed. The text line images are then converted into a bi-level binary image using image thresholding. Global image thresholding is applied to the text line images using Otsu's method as given in [66]. The Otsu method finds a single threshold value for the entire

image and then divides the image into the foreground and the background [67]. The Otsu method is used because it considers optimal inter-class variance, henceforth, the overall process of thresholding is more reliable in handling noise issues. Similarly, the Otsu method is extremely fast and gives good results for scanned documents or images that have uniform illumination over the document [67]. The original line image and its thresholded version are shown in Fig. 2.

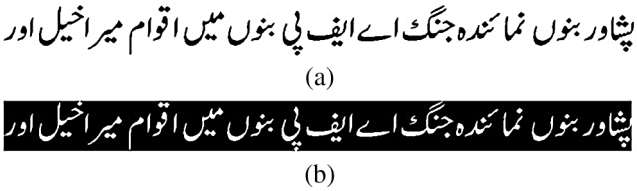

Figure 2: (a) Original image taken from [68]. (b) Thresholded image using Otsu's method

- **Holistic Segmentation:** After the pre-processing, the text line images are segmented into constituent ligatures using a holistic segmentation approach. This results in well-separated segmented ligature images. The segmented ligature images are composed of both the base ligatures and the associated diacritics.
- **Feature Extraction:** A total of 15 hand-engineered features are extracted from the ligature images. Geometric, first-order and second-order statistical features are extracted from the segmented ligature images. Statistical features have been used in the past and have achieved exceptional results for analytical recognition systems [69]. The geometric features (F1 and F2), considered are Aspect Ratio and Compactness. The first-order statistical features are extracted using the VPP (Vertical Projection Profile) and HPP (Horizontal Projection Profile) histogram distribution for the ligature images. First-order features (F3 to F11) include the distributions of Density (F3), Horizontal and Vertical; Edge (F4 and F5), Mean (F6 and F7), Variance (F8 and F9) and Kurtosis (F10 and F11) distribution. The second-order statistical features are more advanced and also observes the relationship among the pixels in the ligature images. The Gray Level Co-occurrence Matrix (GLCM) features (F12 to F15), contrast, correlation, energy and homogeneity are considered. The total number of features (15) before and after the clustering process is the same since in this study no feature selection is observed.

3.2 Intra-Feature Hierarchical Clustering

In raw data, all the ligature observations appear to be dissimilar in so many ways, that it prevents efficient searching and organization. The default intra-feature clusters may be extremely large in number and ligatures in each may vary widely, which is problematic for any machine learning and classification method. The abundant number of clusters for each feature introduces challenges during machine learning tasks, specifically, the objective functions might not work properly. Likewise, it is challenging to train the learning algorithms in a feasible amount of time. The classification accuracy of the model may also be degraded. Hence, having an optimal number of intra-feature clusters becomes an essential task for developing an efficient and robust learning model. Here, a hierarchical clustering algorithm is utilized to generate an optimal number of clusters for each feature (F1 to F15).

The working of the hierarchical clustering algorithm is extremely simple. For each of the features (F1 to F15), the following steps are observed for clustering. First, all the data points for a feature are taken into consideration. The data points are then sorted incrementally, smallest to the largest. Next, the first-order derivative is calculated to find the rate of change between the data points. The change greater than zero reflects data points that may belong to a different cluster. If the change is zero, it means that the data points are similar and may belong to the same

cluster. The first-order derivative is the difference between its adjacent elements and is given as, $[\mathrm{dp}\ (2) - \mathrm{dp}\ (1),\ \mathrm{dp}\ (3) - \mathrm{dp}\ (2),\ \mathrm{dp}\ (4) - \mathrm{dp}\ (3) \ldots\ldots\ldots\ldots,\ \mathrm{dp(N)} - \mathrm{dp(N-1)}]$. Where dp stands for a data point and N is the total number of data points. The change for a single data point is calculated using Eq. (1),

$$\Delta dp_i = (dp_{i+1} - dp_i) \tag{1}$$

The total number of data points are reduced to N − 1 after the first-order derivative. Following, the mean is calculated for only the positive (greater than 0) first-order derivative elements. Subsequently, all first-order derivatives having a value greater than mean are taken into consideration. Next, the second-order derivative is computed on the resultant elements of the first-order derivative to find the segmentation points for clustering.

$$\Delta dp_{i+1} - \Delta dp_i < Threshold \tag{2}$$

A focal data point is selected, $\Delta \mathrm{dp_i}$, and all those data points that fall within the threshold are assigned to a cluster. When the threshold limit is crossed, a new segmentation point is identified for the next collection of the data points. Likewise, a new focal data point is selected, $\Delta \mathrm{dp_{i+1+1}}$. The above steps are repeated until all the first-order derivatives satisfying the mean condition are processed. The proposed system observes a minimum threshold of 30 as the total number of data points in each cluster for a feature. Meaning that a cluster for a feature will have at least 30 ligatures assigned to it. For each feature, the total number of clusters as well as the minimum and maximum data point value per cluster is taken into consideration for the final clustering.

As stated, the process of hierarchical clustering is observed for each of the Features (F1 to F15). Due to space limitations, here, illustrations have been provided for the Feature F7 (Vertical Mean). Fig. 3a shows the initial distribution of data points for feature F7 from the segmented ligature images of the UPTI dataset. It clearly shows that the feature data points are widely dispersed, giving a high distribution of data. The distribution of the feature data points after incremental sorting can be seen in Fig. 3b. This figure shows the data points for feature (F7) for all the 189003 ligatures that have been sorted in ascending order. The graph shows curves sloping upwards when observed from left to right. Suggesting that the data distribution varies widely which may lead the fitness function of optimization problems to fail drastically. The graphical distribution for change, i.e., first-order derivative for feature (F7) extracted from the ligatures images is shown in Fig. 3c. The mean of the positive first-order derivative for feature (F7) is shown in Fig. 3d. The red line, shown in the graph of Fig. 3d represents the mean value. All the resultant elements having a value greater than the mean are considered for further processing.

3.3 Data Representation Using Classification Rules

Using the clustered data points and the given ground-truth classes, the upper and lower threshold limit of a feature for a given class label can be found. The cluster composition of the boundaries represents certain rules that can be used for decision making. Since these rules can be used for decision making, they are said to be classification rules. However, these classification rules are not hand-coded but comprehended from the hierarchical clustering results. There are numerous techniques to represent the cluster composition of the boundaries (upper and lower limits) such as using the propositional logic, conditional rules, trees and networks. Regardless of the representation technique, the output is dependent on the total number of ligatures, the total number of features, upper/lower threshold limits of the data points and the total number of ground-truth classes.

Trees, linear list and network representations have their limitations and are unfeasible to be used here to represent all the clustered output data due to the limitations of page size and numbers. Therefore, the classification rules can be best represented using conditional expressions. Each rule is encoded as a

conditional statement. Each feature corresponds to a condition and the class corresponds to the conclusion. Both the condition and the conclusion parts are dynamic and adaptive and subject to change for different types of datasets, features and the classes under consideration. The classification rules will keep varying based on the optimality of the clustered data using intra-feature hierarchical clustering.

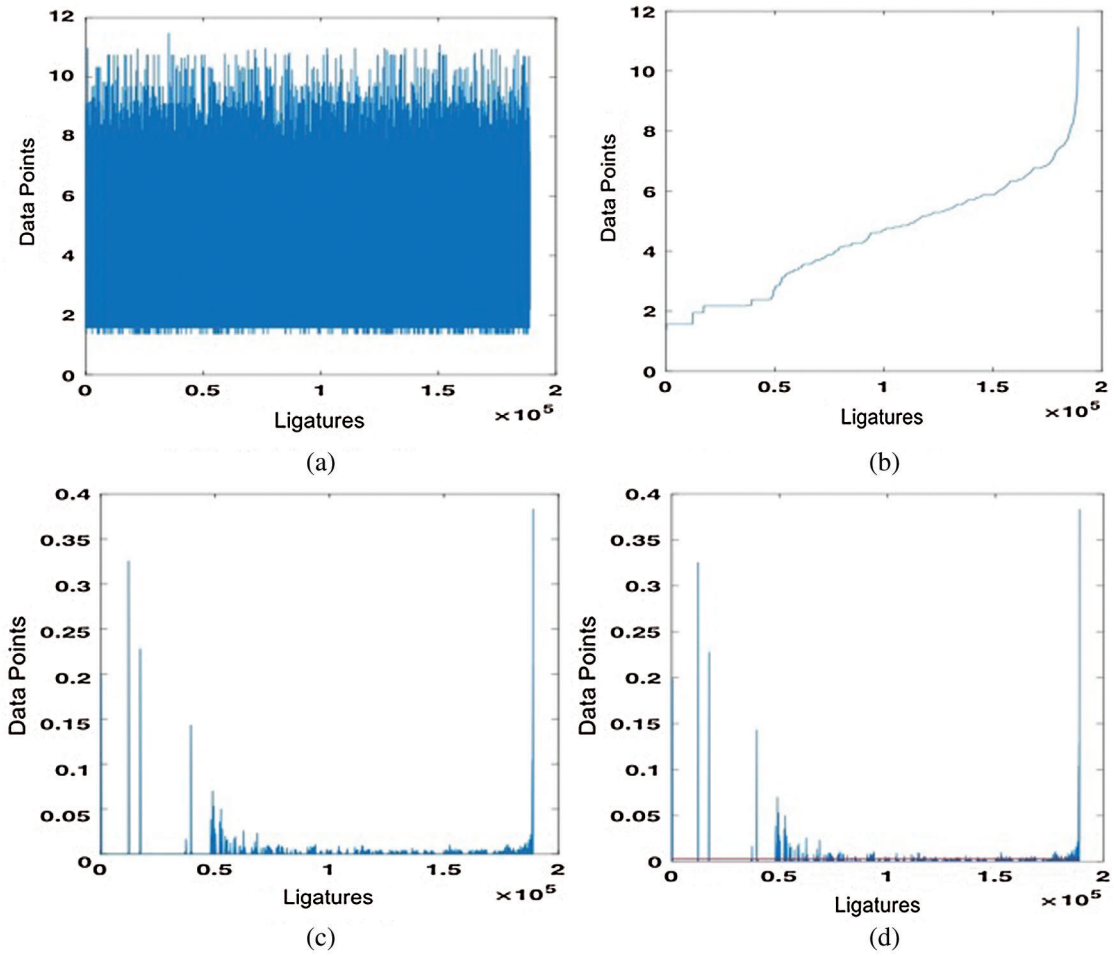

Figure 3: Hierarchical clustering steps observed for feature F7

3.4 Optimization and Recognition

Genetic Algorithm is usually used for various optimization tasks. A genetic algorithm is comprised of standard stages of the initial population, chromosome encoding, parent selection, crossover, mutation, fitness function, survivor selection and the termination process. A genetic algorithm usually begins with an initial population [11,70]. Each of the population provides a subset of solutions in the current generation. These subsets of solutions are generally termed as chromosomes [11]. Usually, the population size is not kept very large since it could slow down the entire GA process. Extremely small population size is also avoided to ensure a good mating pool. The population is usually given as a two-dimensional array of population size and chromosome size. Next, the chromosomes within the population are expressed using a representation. Subsequently, in parent selection, a pair of chromosomes from the population are selected for the further process of reproduction. The reproduction consists of two operations, crossover and mutation. The mutation operation is usually performed to maintain and introduce diversity in the new

population [11]. The mutation operation alters the genes within the same offsprings generated after the crossover operation. The fitness of the chromosomes is found using fitness criteria. In a GA, the survivor selection process determines which chromosomes will survive as a new population to be used for the next generation. During the survivor selection process, it's made sure that the best solutions survive to be used in the next generation. There are numerous techniques for survivor selection. A genetic algorithm terminates when a termination criterion is met or the maximum number of generations is reached. In this paper genetic algorithm is used for the optimization as well as the recognition of Urdu ligatures. Genetic Algorithm is used for optimization of the hierarchical clustering and hence improving the classification rules. The basic architecture of the proposed Urdu ligature recognition system using GA optimization and recognition is shown in Fig. 4. The clustered data is taken and later passed through the standard stages of a GA, observing optimization of rules and recognition of ligatures.

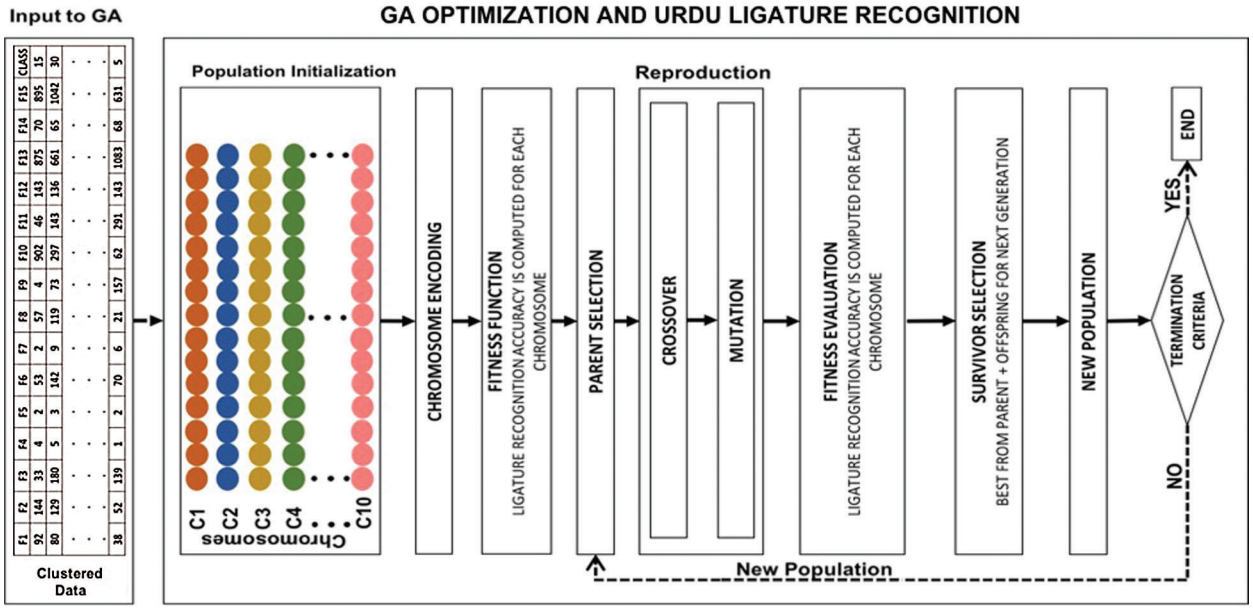

Figure 4: Overview of GA process and input data observed for proposed ligature recognition system

Usually, a GA is used to find an optimal solution(s) to a given computation problem [10,11]. In this paper, the computation problem is finding the best sequence for processing the features for improving the hierarchical clustering and henceforth, generating a set of classification rules for recognition of the ligatures while achieving the maximum accuracy. Therefore, the genetic algorithm in the proposed system deals with the permutation problem of the features. The permutation problem helps in deciding the order in which the features should be processed in each generation. Based on the sequence of features, the clusters are optimized, henceforth, the classification rules are optimized and recognition is improved. The parameter settings used for the Genetic algorithm in the proposed ligature recognition system for optimization and recognition is given in Tab. 2. The details of various parameter settings for each stage of the Genetic Algorithm is discussed in the list below.

Table 2: Parameters for proposed genetic algorithm

Parameter	Value(s)
Number of generations	101
Population per generation	1
Population size (chromosomes)	10
Chromosome size (genes)	15
Allele scale	1 to 15
Chromosome encoding	Permutation
Parent selection	Sequential order
Crossover type	One point
Mutation rate	0.5
Fitness measure	Recognition accuracy
Survivor selection	Elitism (parents + offsprings)
Termination criteria	Maximum limit of generations

1. **Population initialization:** In the proposed recognition system, the genetic algorithm's population size is decided optimally using trial and error. The ligature recognition accuracy is tested for 101 generations. Each generation has 1 population and each population has a total of 10 chromosomes. Each chromosome has 15 genes (equivalent to the number of features). Hence, each population matrix has a size of 10×15 in each generation. Random initialization is used to populate the initial population with completely random solutions.
2. **Chromosome Encoding:** The encoding for each chromosome in the proposed ligature recognition system is the permutation order to access the features for multi-level sorting. For each population, the chromosome stores the information about the access order of the features. Every chromosome is represented as a string of numbers (1 to 15). Each number represents a feature from the feature vector. This representation is then used for mappings from the representation space to the phenotype space in the fitness function, where the hierarchical clustering results are optimized and the ligature recognition accuracy is computed.
3. **Parent Selection:** In the proposed recognition system, the parent chromosome pair is selected sequentially from the population for maintaining good diversity in the future generations.
4. **Crossover:** A single point crossover is applied during the crossover operation. A single crossover point i.e., gene 8 is selected as the cutoff. Initially, up to gene 7, the parents are kept as it is for the offsprings. Genes 8 to 15 are scanned one by one for both the parents and their alleles are exchanged with each other to generate the offspring. If the allele is not repeated in the offspring, it is directly added to the offspring. If any repetition is observed, the offspring are scanned for the first instance of the repeated alleles and they are swapped accordingly. This crossover operation is repeated for all the parents of a given population in a generation. The process is repeated for all the genes from the crossover point (8) until the end of the chromosome.
5. **Mutation:** The genetic algorithm uses the normal random mutation for each chromosome in the proposed recognition system. A mutation rate of 0.5% is considered during the mutation operation. For each crossover offspring in the population, in the mutation operation, first, two random genes are selected from an offspring. Next, their allele is exchanged with each other. The

same process is repeated for all the offsprings in the population. The fitness function is evaluated against the mutated population. Similarly, this mutated population is also used during the survivor selection process.

6. **Fitness Function:** The fitness of each chromosome is calculated as the ligature recognition accuracy of the proposed system. Each chromosome when transformed into the phenotype space, the features are accessed sequentially as per the order of the alleles in the chromosome. Fig. 5 shows a single chromosome sequence as a solution. This chromosome encoding representation is then translated into the phenotype space. The sequence of alleles in the chromosome is taken into consideration. According to the chromosome sequence (6, 10, 15, 7, 14, 13, 12, 2, 1, 11, 4, 8, 3, 9, 5), the features in the phenotype space are selected in the same sequence and are multi-level sorted, ordered in ascending order, during the fitness function optimization process. For example, as given in Fig. 5, the features are multi-level sorted such that feature 6 i.e., the horizontal variance is selected first and sorted, feature 10 i.e., contrast is sorted second, feature 15 i.e., homogeneity is sorted third and so on.

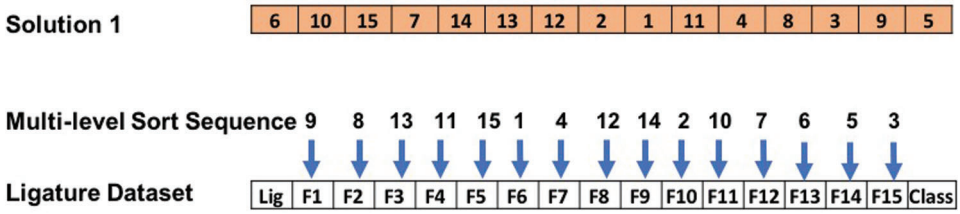

Figure 5: Multi-level column sorting process for a solution

Sorting the features, optimizes the clustered data and generates an optimized set of classification rules for recognition. The fitness of a chromosome is assessed by its ligature recognition accuracy for a sample test data. The fitness for each chromosome is found using Eq. (3).

$$Chr_{Acc} = \frac{\sum_{i=1}^{n-1} Clus_i + \frac{\sum(tclass_i == clsmode_i)}{\sum (tclass_i > -1)}}{Total\ Test\ Data} \tag{3}$$

In order to find the recognition accuracy in the fitness function, the dataset in the phenotype space is sorted according to the gene sequence for a chromosome from the population. For sample test data, the last allele in the sequence is taken and its feature is located in the phenotype space. Next, for this feature, the difference between the adjacent elements is found. All those elements for which the feature has a difference greater than zero are found. The number of observations within two adjacent difference elements is iteratively scanned for its accuracy. The ground-truth, the class information is assigned to all those observations and the most repeated class is found. The accuracy of each chromosome sequence in the population is found by evaluating and summing all the cluster accuracies.

7. **Survivor Selection:** In the proposed recognition system, the genetic algorithm uses the elitism. For a generation g, if P is the parents and O are the offsprings, the elitism results in generating a new population by replacing the current population pol. Elitism ensures that the solutions obtained by a GA will not decrease from one generation to the next generation. Best 10 chromosomes from the parent and the offsprings are selected based on its fitness as a new population for the next generation.
8. **Termination:** In the proposed ligature recognition system, the Genetic Algorithm's termination condition is met when the maximum limit of the generations i.e. 101 is reached. At the end of

each generation, a set of optimized rules are generated, each chromosome represents a set of optimized classification rules for ligature recognition.

4 Experiments and Results

This section explores the results for the intra-feature hierarchical clustering, classification rules and the genetic algorithm-based optimization and Urdu ligature recognition. The dataset used for evaluation is also addressed.

4.1 Dataset and Ground-Truth

The performance of the proposed method is evaluated on the benchmark dataset, named, UPTI, developed by Sabbour et al. [68]. UPTI is one of the most renowned datasets used for comparison and evaluation of Urdu recognition systems. It contains a total of 10,063 text line image, having different versions. The undegraded text line images from the UPTI dataset are taken into consideration here. The proposed recognition system uses the ground-truth data from the UPTI dataset, corresponding to each text line image and its subsequent successfully segmented ligature image. A total of 189003 successfully segmented ligatures are considered that are divided into 3645 unique classes.

4.2 Intra-Feature Hierarchical Clustering and Classification Rules Results

The intra-feature hierarchical feature clustering divides categorizes feature's data point for all ligatures into groups i.e., clusters for improved understanding and summarization. The hierarchical clustering approach generates optimal clusters for each of the features (F1 to F15). The detailed results for intra-feature hierarchical clustering are shown in Fig. 6. The blue line in the chart indicates the total number of unique data points (clusters) for each feature initially or by default without any clustering algorithm. The orange line in the chart indicates the clusters for each feature after the hierarchical clustering algorithm. The clustering algorithm doesn't require the total number of clusters to be known in advance. However, the minimum threshold for a cluster needs to be set. The minimum number of data points for each cluster is set to 30.

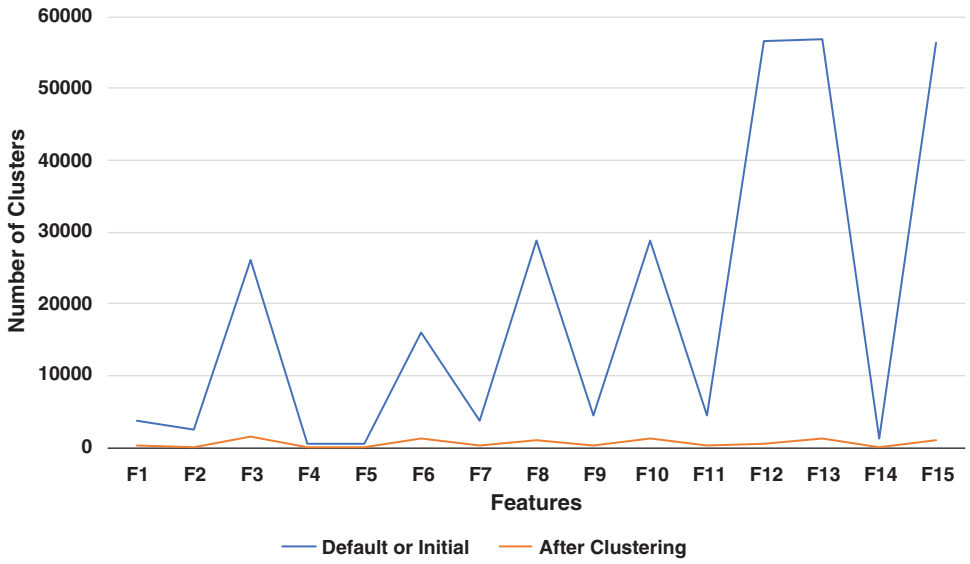

Figure 6: Intra-feature clustering results

Once, all the features are clustered, classification rules are used for the representation of the clustered data. Using the classification rules, each ligature is associated to a class (1 to 3645). A total of 3645 rules are generated for associating each ligature to a known class. The rules for some of the classes (first four classes and last four classes) have been provided in a tabular form in Tab. 3.

Table 3: Classification rules

Rule	Conditions (Cond1:Cond15)	Class C1:C3645	Class Ligature
1	$(15 \leq F1 \leq 121) \wedge (16 \leq F2 \leq 162) \wedge (1 \leq F3 \leq 58) \wedge (3 \leq F4 \leq 5) \wedge (F5 = 2) \wedge (1 \leq F6 \leq 64) \wedge (F7 = 1) \wedge (34 \leq F8 \leq 91) \wedge (F9 = 6) \wedge (1153 \leq F10 \leq 1338) \wedge (F11 = 120) \wedge (41 \leq F12 \leq 168) \wedge (3 \leq F13 \leq 1110) \wedge (F14 = 70) \wedge (183 \leq F15 \leq 1022)$	C1	ء
2	$(13 \leq F1 \leq 217) \wedge (14 \leq F2 \leq 164) \wedge (71 \leq F3 \leq 1239) \wedge (1 \leq F4 \leq 8) \wedge (2 \leq F5 \leq 10) \wedge (32 \leq F6 \leq 963) \wedge (6 \leq F7 \leq 309) \wedge (11 \leq F8 \leq 317) \wedge (142 \leq F9 \leq 355) \wedge (26 \leq F10 \leq 1175) \wedge (80 \leq F11 \leq 333) \wedge (26 \leq F12 \leq 229) \wedge (30 \leq F13 \leq 1167) \wedge (2 \leq F14 \leq 68) \wedge (77 \leq F15 \leq 1059)$	C2	آ
3	$(50 \leq F1 \leq 66) \wedge (72 \leq F2 \leq 105) \wedge (173 \leq F3 \leq 208) \wedge (F4 = 6) \wedge (F5 = 10) \wedge (136 \leq F6 \leq 154) \wedge (F7 = 16) \wedge (F8 = 34) \wedge (F9 = 161) \wedge (48 \leq F10 \leq 67) \wedge (F11 = 310) \wedge (F12 = 146) \wedge (419 \leq F13 \leq 805) \wedge (F14 = 64) \wedge (436 \leq F15 \leq 978)$	C3	أ
4	$(19 \leq F1 \leq 117) \wedge (22 \leq F2 \leq 160) \wedge (64 \leq F3 \leq 760) \wedge (5 \leq F4 \leq 8) \wedge (2 \leq F5 \leq 10) \wedge (69 \leq F6 \leq 375) \wedge (9 \leq F7 \leq 59) \wedge (95 \leq F8 \leq 131) \wedge (73 \leq F9 \leq 146) \wedge (68 \leq F10 \leq 708) \wedge (94 \leq F11 \leq 143) \wedge (60 \leq F12 \leq 161) \wedge (180 \leq F13 \leq 1102) \wedge (49 \leq F14 \leq 65) \wedge (249 \leq F15 \leq 1022)$	C4	ؤ
3642	$(340 \leq F1 \leq 346) \wedge (17 \leq F2 \leq 68) \wedge (1375 \leq F3 \leq 1572) \wedge (F4 = 17) \wedge (F5 = 20) \wedge (1275 \leq F6 \leq 1279) \wedge (F7 = 339) \wedge (1065 \leq F8 \leq 1067) \wedge (F9 = 127) \wedge (75 \leq F10 \leq 82) \wedge (F11 = 59) \wedge (590 \leq F12 \leq 592) \wedge (648 \leq F13 \leq 666) \wedge (F14 = 2) \wedge (84 \leq F15 \leq 210)$	C3642	لیفٹیننٹ
3643	$(F1 = 329) \wedge (F2 = 97) \wedge (F3 = 1156) \wedge (F4 = 17) \wedge (F5 = 20) \wedge (F6 = 1275) \wedge (F7 = 372) \wedge (F8 = 638) \wedge (F9 = 122) \wedge (F10 = 72) \wedge (F11 = 27) \wedge (F12 = 592) \wedge (F13 = 664) \wedge (F14 = 1) \wedge (F15 = 84)$	C3643	نیفیکیشن
3644	$(337 \leq F1 \leq 339) \wedge (70 \leq F2 \leq 74) \wedge (1240 \leq F3 \leq 1254) \wedge (F4 = 17) \wedge (F5 = 20) \wedge (F6 = 1278) \wedge (F7 = 374) \wedge (587 \leq F8 \leq 591) \wedge (F9 = 122) \wedge (79 \leq F10 \leq 80) \wedge (F11 = 27) \wedge (F12 = 592) \wedge (666 \leq F13 \leq 668) \wedge (F14 = 1) \wedge (89 \leq F15 \leq 159)$	C3644	ٹیفیکیشن
3645	$(F1 = 346) \wedge (3 \leq F2 \leq 16) \wedge (1095 \leq F3 \leq 1336) \wedge (F4 = 17) \wedge (F5 = 20) \wedge (1277 \leq F6 \leq 1279) \wedge (F7 = 284) \wedge (970 \leq F8 \leq 1006) \wedge (F9 = 90) \wedge (83 \leq F10 \leq 86) \wedge (F11 = 109) \wedge (F12 = 592) \wedge (665 \leq F13 \leq 669) \wedge (F14 = 2) \wedge (72 \leq F15 \leq 92)$	C3645	سٹیبلشمنٹ

4.3 Genetic Algorithm Results

The proposed Urdu ligature recognition system uses a genetic algorithm for optimization and recognition. Each chromosome represents a sequence to access the features in the feature vector for processing (multi-level sorting) within the hierarchically clustered ligature dataset. Each allele represents a feature of the ligature and the gene location represents the sequence in which the features need to be accessed. At the end of each generation, the clustered data is optimized, henceforth, the classification rules are optimized. The ligature recognition accuracy is of the system is computed for each chromosome in the population, over several generations. Since the classification rules are modified at the end of each generation for the entire dataset 189003 ligatures, hence the entire dataset is used for training. While for each chromosome the dataset 189003 is partitioned, a random sized test data is taken from within the 189003 ligatures and is used for evaluating the fitness function of a chromosome. The fitness of a chromosome is assessed by its ligature recognition accuracy for a given test data.

4.3.1 Ligature Recognition Accuracy

For each chromosome, the ligature recognition is calculated after multi-level sorting of the entire ligature dataset and hence optimization of the classification rules. Fitness evaluation i.e., the recognition accuracy of each chromosome is tested for all the ligatures for which the difference between consecutive cluster data points for the lowest-level feature is greater than zero. Over the generations, for each chromosome, the test data taken are of varying sizes from within the original dataset. Therefore, as reported earlier the training size is 189003 ligatures. While the test data is taken by partitioning the original dataset randomly, with varying size of test data for each chromosome to optimize the recognition results. Hence, a total of 1010 random test samples are generated for each of the 1010 chromosomes over the 101 generations. The size of test data keeps increasing to check the robustness of the proposed method. The smallest test data sample reported is of 162 ligatures that are taken by randomly partitioning the 189003 ligatures, whereas the maximum test data sample is of 30755 ligatures. The ligature test data sizes used for all the generations is shown in Fig. 7.

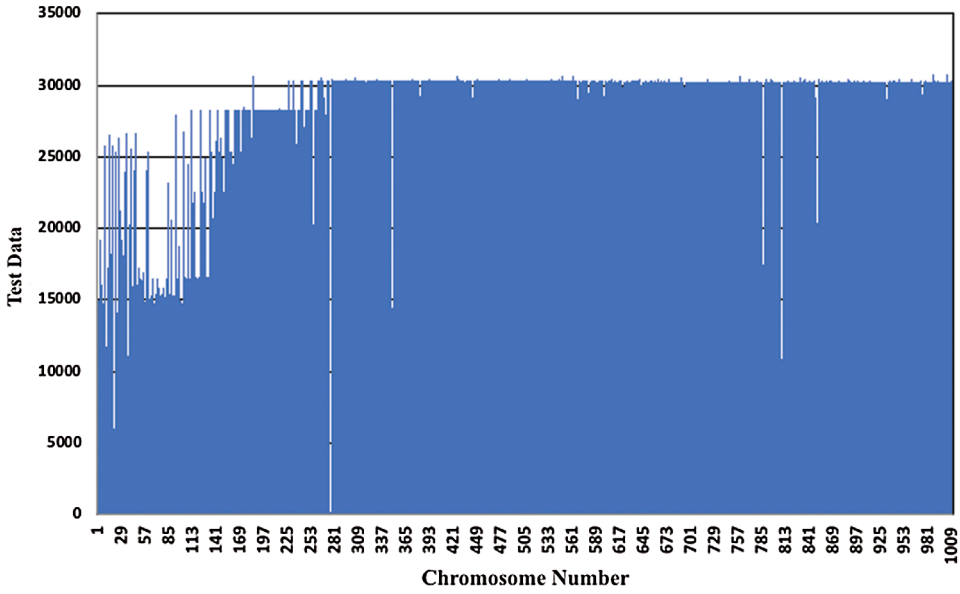

Figure 7: Test data sizes to evaluate ligature recognition accuracy for each chromosome

The recognition results for the test ligature samples across various generations (101) for all chromosomes in the population are evaluated. Survivors are selected at the end of each generation using the fitness (accuracy in %) of the chromosomes, parents as well as the offsprings. A total of 10 survivors are selected at the end of each generation. It should be noted that in the proposed recognition system, the survivors are generated for a total of 100 generations, since, the first generation doesn't go through the survivor selection process. The results for survivor selection are promising since the fitter individuals have not been kicked out and are kept for the next generation. The maximum ligature recognition accuracy achieved at the end of each generation is given in Fig. 8. For the first generation, the maximum recognition accuracy of 87.95% is achieved for a sample test data. Throughout the remaining 100 generations, the hierarchical clustering is optimized and hence, the recognition accuracy is improved. The maximum ligature recognition rate of 96.72% is obtained for the 66th generation from the highest survivor. The accuracy at the end of each generation shows that the best chromosomes are selected and therefore the recognition rate never decreases.

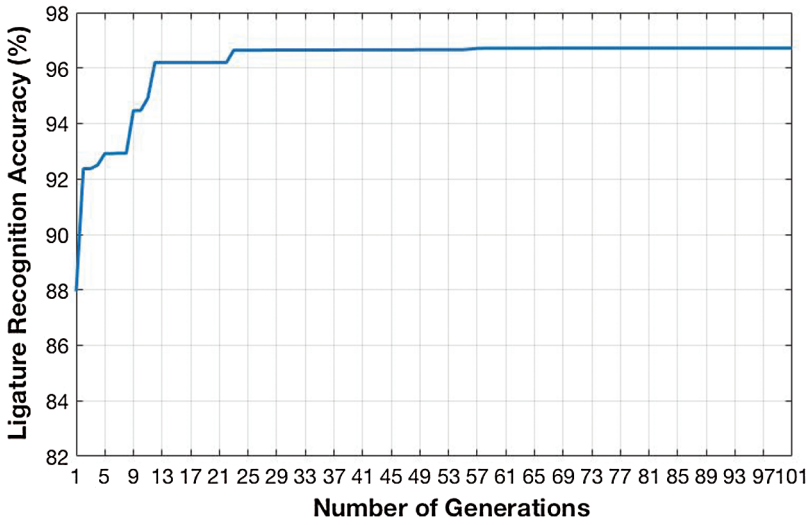

Figure 8: Maximum ligature recognition accuracy (%) achieved for each generation

5 Discussion

This research article proposes the use of a genetic algorithm for Urdu Nastalique ligature recognition using hand-engineered features and a holistic approach. The proposed system leads to high performance for ligature recognition due to its GA consistent optimization using multi-level sorting of the clustered data and the classification rules. For any GA the best solutions are those that have a common ancestor and their fitness is very identical both to each other and to that of high fitness solution from the previous generations. The proposed solutions generated by the genetic algorithm used for Urdu ligature recognition also stabilizes after a time and converges towards a common fitness value (see Fig. 9). The chromosomes in a population become increasingly similar after each generation and hence converges to a common solution. The 77th generation and onwards, the solutions (chromosomes) converges towards a fitness value of 96.72%. Also, the algorithm doesn't have premature convergence, variability is maintained in the population over several generations.

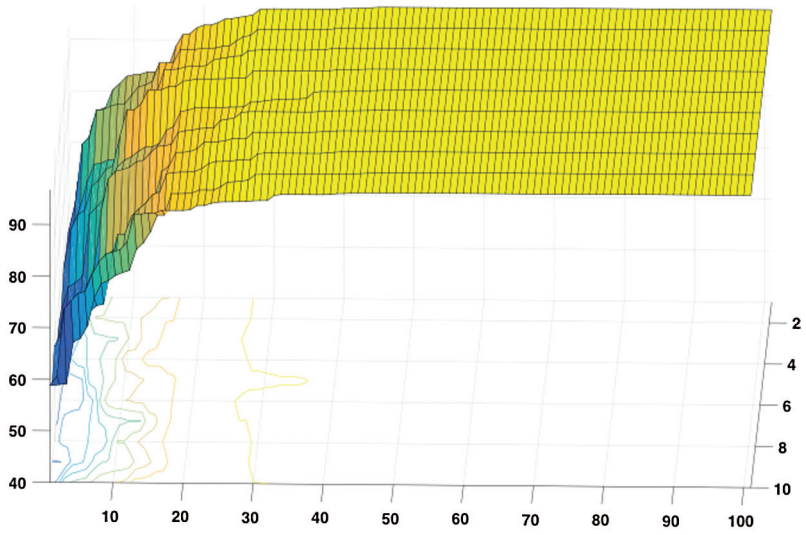

Figure 9: Convergence towards a common solution

The comparison of the proposed Urdu text recognition system to other systems can be based on the textual unit, type of methods and the dataset. Therefore, it is best to compare the proposed system with other ligature-based recognition systems (see Tab. 4). The table contains two parts traditional approach based recognition systems [17,20,21,68,71–77] and deep learning-based recognition systems [18,19,78]. Few of the studies have used deep learning methods for recognition; Ahmad et al. [18] achieved an accuracy of 96.71% for 187039 ligature images, divided into 3732 unique classes. A total of 127180 ligatures were used for training and 29935 ligatures were used for testing. Whereas, in another study, Ahmad et al. [19] extracted stacked autoencoder features from raw pixels of 178573 segmented ligature images, adjusted in 3732 classes. The system used 60,000 each from the UPTI dataset's degraded version (jitter and sensitivity) for validation and testing. In a study by Javed et al. [78], Convolution Neural Networks were used leading to 95% accuracy on 38000 training images and 17000 query images.

Table 4: Comparison to existing Urdu ligature recognition systems

Approach Type: Deep Learning			
Study	Method	Ligatures	Accuracy
Ahmad et al. [18]	Gated bidirectional LSTM	Training: 127180 testing: 29935	96.71%
Ahmad et al. [19]	Stacked denoising autoencoder and SoftMax	178573	96%
Javed et al. [78]	Convolution neural network	Training: 38000 testing: 17000	95%
Approach Type: Traditional Learning			
Study	Method	Ligatures	Accuracy
Rana et al. [20]	SVM, k-NN	11,000	90.29%
Khattak et al. [21]	HMM	2,028	97.93%

Table 4 (continued).

Chanda et al. [71]	Binary tree classifier	3210 Urdu words	98.09%
Javed et al. [74]	HMM	1692	92.73%
Nazir et al. [72]	Correlation method	6728	97.40%
Husain [73]	Feed forward back propagation neural network	200	100%
Javed et al. [75]	HMM	Training: 1282 testing: 3655	92%
Razzak et al. [76]	HMM, fuzzy logic	1800	Nastalique: 87.6% Naskh: 74.1%
Husain et al. [77]	BPNN	850 (1, 2 and 3 char ligature): 18000 for recognition	Base ligatures: 93%, Secondary ligatures: 98%
Sabbour et al. [68]	K-NN	10,000	91%
Din et al. [17]	HMM	Training: 1525 HFL clusters, query ligatures: 6187	92.26%
Proposed ligature recognition system	**Evolutionary approach**	**Training: 189, 003 testing: Variable**	**96.72%**

The best comparison of the proposed system is possible with other systems that used traditional learning methods for recognition. Although the accuracy of [21,71,72,73] is higher than the accuracy of the proposed system, the authors have used extremely small datasets for training and testing the classifiers. The best comparison of the proposed system is possible with the work of Sabbour and Shafait [68], and Din et al. [17] that have used the same UPTI dataset. Sabbour et al. [68] extracted simple contour-based features from the segmented ligature images and used K-NN for classification, a dataset of 10,000 ligatures was used to acquire accuracy of 91%. Similarly, Din et al. [17] extracted simple statistical features and used HMM for classification, 1525 ligature clusters were used and a recognition rate of 92.26% was reported for complete ligatures i.e., 5708 ligatures were correctly recognized from a total of 6187 ligatures. In comparison to the presented studies, the accuracy of the proposed GA based ligature recognition system is 96.72% that is one of the highest accuracies reported for Urdu ligature recognition systems using traditional approaches and a large dataset. Additionaly, in comparison, to the deep learning architectures, the proposed system provides the possibility to be executed on a lower-end computing device for a large dataset of ligatures.

6 Conclusion and Future Recommendation

A holistic recognition system observes the ligatures as a basic unit for the recognition. This paper proposes a ligature recognition system for printed Urdu script. The overall recognition system comprises the use of a genetic algorithm-based approach for optimization and recognition. Initially, intra-feature hierarchical clustering is applied for clustering the data points for each of the features. After the hierarchical clustering classification rules are generated, where each ligature belongs to a unique class (1 to 3645). The clustered data, along with the ground-truth information is then given to a genetic

algorithm for optimization of the hierarchical clustering and recognition of the ligatures. The proposed method is evaluated on the segmented 189003 ligatures. The overall recognition rate for the proposed system is 96.72%.

Currently, the system has been used for ligature recognition, in the future, the same approach can be modified to be used for analytical recognition of Urdu script. Potential researchers may also investigate the use of other hand-engineered or automated features. Handwriting recognition is extremely complex due to the variations in writing styles. The recognition system can be modified to be used for recognition of complex handwritten Urdu text. The stage of post-processing can be added to deal with complex situations like grammar correction, spelling detection and correction. In the future, the proposed system can be extended and modified to be used for other cursive handwritten and printed scripts such as Arabic, Pashto, Panjabi, Seraiki, Kurdish, Persian, Sindhi, Kashmiri, Ottoman Turkish and Malay.

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Hussain, "Resources for Urdu language processing," in *Proc. of the 6th Workshop on Asian Language Resources*, Hyderabad, India, 2008.

[2] S. T. Javed and S. Hussain, "Improving Nastalique specific pre-recognition process for Urdu OCR," in *Proc. of 13th Int. Multitopic Conf. INMIC*, Islamabad, Pakistan, pp. 1–6, 2009.

[3] N. H. Khan and A. Adnan, "Urdu optical character recognition systems: Present contributions and future directions," *IEEE Access*, vol. 6, pp. 46019– 46046, 2018.

[4] N. H. Khan, A. Adnan and S. Basar, "Urdu ligature recognition using multi-level agglomerative hierarchical clustering," *Cluster Computing*, vol. 21, no. 1, pp. 503–514, 2018.

[5] S. A. Sattar, S. U Haque and M. K. Pathan, "A finite state model for Urdu Nastalique optical character recognition," *International Journal of Computer Science And Network Security*, vol. 9, no. 9, 116, 2009.

[6] S. Naz, K. Hayat, M. Imran Razzak, M. Waqas Anwar, S. A. Madani *et al.,* "The optical character recognition of Urdu-like cursive scripts," *Pattern Recognition*, vol. 47, no. 3, pp. 1229–1248, 2014.

[7] S. Naz, K. Hayat, M. I. Razzak, M. W. Anwar and H. Akbar, "Arabic script based language character recognition: Nasta'liq *vs.* Naskh analysis," in *World Congress on Computer and Information Technology*, Sousse, Tunisia, pp. 1–7, 2013.

[8] S. B. Ahmed, S. Naz, S. Swati, M. I. Razzak, A. I. Umar *et al.,* "UCOM offline dataset—An Urdu handwritten dataset generation," *International Arab Journal of Information Technology*, vol. 14, no. 2, pp. 239–245, 2017.

[9] S. Naz, K. Hayat, M. W. Anwar, H. Akbar and M. I. Razzak, "Challenges in baseline detection of cursive script languages," in *Science and Information Conf. (SAI)*, London, UK, pp. 551–556, 2013.

[10] J. H. Holland, "Genetic algorithms," *Scientific American*, vol. 267, no. 1, pp. 66–73, 1992.

[11] J. Carr, "An introduction to genetic algorithms," *Senior Project*, vol. 1, 40, 2014.

[12] S. Naz, A. I. Umar, S. H. Shirazi, S. B. Ahmed, M. I. Razzak *et al.,* "Segmentation techniques for recognition of Arabic-like scripts: A comprehensive survey," *Education and Information Technologies*, vol. 21, no. 5, pp. 1225–1241, 2016.

[13] S. Naz, A. I. Umar, S. B. Ahmed, S. H. Shirazi, M. I. Razzak *et al.,* "An OCR system for printed Nasta'liq script: A segmentation based approach," in *Proc. of 17th Int. Multi-Topic Conf. (INMIC)*, Karachi, Pakistan, pp. 255–259, 2014.

[14] S. Naz, A. I. Umar, R. Ahmad, I. Siddiqi, S. B. Ahmed *et al.,* "Urdu Nastaliq recognition using convolutional-recursive deep learning," *Neurocomputing*, vol. 243, pp. 80–87, 2017.

[15] S. Naz, A. I. Umar, R. Ahmad, S. B. Ahmed, S. H. Shirazi *et al.,* "Urdu Nasta'liq text recognition system based on multi-dimensional recurrent neural network and statistical features," *Neural Computing and Applications*, vol. 28, no. 2, pp. 219–231, 2017.

[16] S. Naz, A. I. Umar, R. Ahmad, S. B. Ahmed, S. H. Shirazi *et al.,* "Offline cursive Urdu-Nastaliq script recognition using multidimensional recurrent neural networks," *Neurocomputing*, vol. 177, pp. 228–241, 2016.

[17] I. U. Din, I. Siddiqi, S. Khalid and T. Azam, "Segmentation-free optical character recognition for printed Urdu text," *EURASIP Journal on Image and Video Processing*, vol. 2017, no. 1, pp. 62, 2017.

[18] I. Ahmad, X. Wang, Yhao Mao, G. Liu, H. Ahmad *et al.,* "Ligature based Urdu Nastaleeq sentence recognition using gated bidirectional long short term memory," *Cluster Computing*, vol. 21, no. 1, pp. 703–714, 2018.

[19] I. Ahmad, X. Wang, R. Li and S. Rasheed, "Offline Urdu Nastaleeq optical character recognition based on stacked denoising autoencoder," *China Communications*, vol. 14, no. 1, pp. 146–157, 2017.

[20] A. Rana and G. S. Lehal, "Offline Urdu OCR using ligature based segmentation for Nastaliq Script," *Indian Journal of Science and Technology*, vol. 8, no. 35, pp. 1–9, 2015.

[21] I. U. Khattak, I. Siddiqi, S. Khalid and C. Djeddi, "Recognition of Urdu ligatures—A holistic approach," in *13th Int. Conf. on Document Analysis and Recognition*, Tunis, Tunisia, pp. 71–75, 2015.

[22] A. Abidi, I. Siddiqi and K. Khurshid, "Towards searchable digital Urdu libraries—a word spotting based retrieval approach," in *Int. Conf. on Document Analysis and Recognition (ICDAR)*, Beijing, China, pp. 1344–1348, 2011.

[23] A. Abidi, A. Jamil, I. Siddiqi and K. Khurshid, "Word spotting based retrieval of Urdu handwritten documents," in *Int. Conf. on Frontiers in Handwriting Recognition (ICFHR)*, Bari, Italy, pp. 331–336, 2012.

[24] R. Hussain, H. A. Khan, I. Siddiqi, K. Khurshid and A. Masood, "Keyword based information retrieval system for Urdu document images," in *11th Int. Conf. on Signal-Image Technology & Internet-Based Systems*, Bangkok, Thailand, pp. 27–33, 2015.

[25] M. Ijaz and S. Hussain, "*Corpus* based Urdu lexicon development," in *Proc. of Conf. on Language Technology*, Pakistan: University of Peshawar, vol. 73, 2007.

[26] S. Hussain and M. Afzal, "Urdu computing standards: Urdu Zabta Takhti UZT 1.01," in *Proc. of the IEEE Int. Multi-Topic Conf.*, LUMS, Lahore, pp. 223–228, 2001.

[27] S. Naz, A. Iqbal Umar, S. Hamad Shirazi, S. Ahmad Khan, I. Ahmed *et al.,* "Challenges of Urdu named entity recognition: A scarce resourced language," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 8, no. 10, pp. 1272–1278, 2014.

[28] S. Naz, A. I. Umar and M. I. Razzak, "A hybrid approach for NER system for scarce resourced language-URDU: Integrating n-gram with rules and gazetteers," *Mehran University Research Journal of Engineering & Technology*, vol. 34, no. 4, pp. 349–358, 2015.

[29] N. Durrani and S. Hussain, "Urdu word segmentation," in *Proc. of Annual Conf. of the North American Chapter of the Association for Computational Linguistics*, Los Angeles, California, pp. 528–536, 2010.

[30] S. Hussain, "Letter-to-sound conversion for Urdu text-to-speech system," in *Proc. of the Workshop on Computational Approaches to Arabic Script-Based Languages*, Geneva, Switzerland, pp. 74–79, 2004.

[31] I. Shamsher, Z. Ahmad, J. K. Orakzai and A. Adnan, "OCR for printed Urdu script using feed forward neural network," in *Proc. of World Academy of Science, Engineering and Technology*, vol. 23, pp. 172–175, 2007.

[32] Q. Safdar and K. U. Khan, "Online Urdu handwritten character recognition: Initial half form single stroke characters," in *12th Int. Conf. on Frontiers of Information Technology*, Islamabad, Pakistan, pp. 292–297, 2014.

[33] J. Tariq, U. Nauman and M. U. Naru, "Softconverter: A novel approach to construct OCR for printed Urdu isolated characters," in *2nd Int. Conf. on Computer Engineering and Technology*, Chengdu, China, vol. 3, pp. 495–498, 2010.

[34] Q. U. A. Akram, S. Hussain and Z. Habib, "Font size independent OCR for Noori Nastaleeq," in *Proc. of Graduate Colloquium on Computer Sciences*, Lahore, Pakistan, vol. 1, 2010.

[35] K. Khan, R. Ullah, N. A. Khan and K. Naveed, "Urdu character recognition using principal component analysis," *International Journal of Computer Applications*, vol. 60, no. 11, pp. 1–4, 2012.

[36] K. Khan, R. U. Khan, A. Alkhalifah and N. Ahmad, "Urdu text classification using decision trees," in *12th Int. Conf. on High-Capacity Optical Networks and Enabling/Emerging Technologies*, Islamabad, Pakistan, pp. 1–4, 2015.

[37] S. A. Hussain, S. Zaman and M. Ayub, "A self organizing map based Urdu Nasakh character recognition," in *Int. Conf. on Emerging Technologies*, Islamabad, Pakistan, pp. 267–273, 2009.

[38] G. S. Lehal and A. Rana, "Recognition of Nastalique Urdu ligatures," in *Proc. of the 4th Int. Workshop on Multilingual OCR*, Washington, D.C., USA, pp. 1–5, 2013.

[39] T. Nawaz, S. A. H. S. Naqvi, H. U. Rehman and A. Faiz, "Optical character recognition system for Urdu (Naskh font) using pattern matching technique," *International Journal of Image Processing*, vol. 3, no. 3, pp. 92–104, 2009.

[40] E. R. Q. Khan and E. W. Q. Khan, "Urdu optical character recognition technique for Jameel Noori Nastaleeq script," *Journal of Independent Studies and Research*, vol. 13, no. 1, pp. 81–86, 2015.

[41] W. Q. Khan and R. Q. Khan, "Urdu optical character recognition technique using point feature matching; a generic approach," in *Int. Conf. on Information and Communication Technologies*, Karachi, Pakistan, pp. 1–7, 2015.

[42] Q. u. A. Akram, S. Hussain, A. Niazi, U. Anjum and F. Irfan, "Adapting tesseract for complex scripts: An example for Urdu Nastalique," in *11th IAPR Int. Workshop on Document Analysis Systems*, Tours, France, pp. 191–195, 2014.

[43] Z. Ahmad, J. K. Orakzai, I. Shamsher and A. Adnan, "Urdu Nastaleeq optical character recognition," in *Proc. of World Academy of Science, Engineering and Technology*, vol. 26, pp. 249–252, 2007.

[44] Z. Ahmad, J. K. Orakzai and I. Shamsher, "Urdu compound character recognition using feed forward neural networks," in *Proc. of the 2nd Int. Conf. on Computer Science and Information Technology*, Beijing, China, pp. 457–462, 2009.

[45] S. B. Ahmed, S. Naz, M. I. Razzak, S. F. Rashid, M. Z. Afzal *et al.,* "Evaluation of cursive and non-cursive scripts using recurrent neural networks," *Neural Computing and Applications*, vol. 27, no. 3, pp. 603–613, 2016.

[46] A. Ul-Hasan, S. B. Ahmed, F. Rashid, F. Shafait and T. M. Breuel, "Offline printed Urdu Nastaleeq script recognition with bidirectional LSTM networks," in *Proc. of 12th Int. Conf. on Document Analysis and Recognition*, Washington, DC, USA, pp. 1061–1065, 2013.

[47] R. Patel and M. Thakkar, "Handwritten Nastaleeq script recognition with BLSTM-CTC and ANFIS method," *International Journal of Computer Trends and Technology*, vol. 11, no. 3, pp. 131–136, 2014.

[48] S. Naz, S. B. Ahmed, R. Ahmad and M. I. Razzak, "Zoning features and 2DLSTM for Urdu text-line recognition," *Procedia Computer Science*, vol. 96, pp. 16–22, 2016.

[49] S. Naz, A. I. Umar, R. Ahmed, M. I. Razzak, S. F. Rashid *et al.,* "Urdu Nasta'liq text recognition using implicit segmentation based on multi-dimensional long short term memory neural networks," *SpringerPlus*, vol. 5, no. 1, pp. 2010, 2016.

[50] A. Onan and S. Korukoğlu, "A feature selection model based on genetic rank aggregation for text sentiment classification," *Journal of Information Science*, vol. 43, no. 1, pp. 25–38, 2017.

[51] S. Sural and P. Das, "A genetic algorithm for feature selection in a neuro-fuzzy OCR system," in *Proc. of Sixth Int. Conf. on Document Analysis and Recognition*, Seattle, WA, USA, pp. 987–991, 2001.

[52] S. K. Pal and P. P. Wang, *Genetic Algorithms for Pattern Recognition*. 1st ed. Boca Raton: CRC Press, 2017.

[53] J. Tarigan, R. Diedan and Y. Suryana, "Plate recognition using backpropagation neural network and genetic algorithm," *Procedia Computer Science*, vol. 116, pp. 365–372, 2017.

[54] M. Middlemiss and G. Dick, "Feature selection of intrusion detection data using a hybrid genetic algorithm/KNN approach, " in *Design and Application of Hybrid Intelligent Systems*. Amsterdam, The Netherlands: IOS Press, pp. 519–527, 2003.

[55] M. S. Hoque, "An implementation of intrusion detection system using genetic algorithm," *International Journal of Network Security & Its Applications*, vol. 4, no. 2, pp. 109–120, 2012.

[56] T. Saba, A. Rehman and G. Sulong, "Non-linear segmentation of touched roman characters based on genetic algorithm," *International Journal on Computer Science and Engineering*, vol. 2, no. 6, pp. 2167–2172, 2010.

[57] X. Wei, S. Ma and Y. Jin, "Segmentation of connected Chinese characters based on genetic algorithm," in *Proc. of Eight Int. Conf. on Document Analysis and Recognition*, Seoul, South Korea, pp. 645–649, 2005.

[58] A. M. Alimi, "An evolutionary neuro-fuzzy approach to recognize on-line Arabic handwriting," in *Proc. of the Fourth Int. Conf. on Document Analysis and Recognition*, Ulm, Germany, 1, pp. 382–386, 1997.

[59] G. Abandah and N. Anssari, "Novel moment features extraction for recognizing handwritten Arabic letters," *Journal of Computer Science*, vol. 5, no. 3, pp. 226–232, 2009.

[60] M. Kherallah, F. Bouri and A. M. Alimi, "On-line Arabic handwriting recognition system based on visual encoding and genetic algorithm," *Engineering Applications of Artificial Intelligence*, vol. 22, no. 1, pp. 153–170, 2009.

[61] M. A. Abed, A. N. Ismail and Z. M. Hazi, "Pattern recognition using genetic algorithm," *International Journal of Computer and Electrical Engineering*, vol. 2, no. 3, pp. 583–588, 2010.

[62] A. M. Alimi, "Evolutionary computation for the recognition of on-line cursive handwriting," *IETE Journal of Research*, vol. 48, no. 5, pp. 385–396, 2002.

[63] S. Gazzah and N. B. Amara, "Neural networks and support vector machines classifiers for writer identification using Arabic script," *International Arab Journal of Information Technology*, vol. 5, no. 1, pp. 92–101, 2008.

[64] M. Soryani and N. Rafat, "Application of genetic algorithms to feature subset selection in a Farsi OCR," *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, vol. 2, no. 6, pp. 2167–2170, 2008.

[65] R. Kala, H. Vazirani, A. Shukla and R. Tiwari, "Offline handwriting recognition using genetic algorithm," *International Journal of Computer Science Issues*, vol. 7, no. 2, pp. 16–25, 2010.

[66] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[67] F. Shafait, D. Keysers and T. M. Breuel, "Efficient implementation of local adaptive thresholding techniques using integral images," in *Proc. Document recognition and retrieval XV*, San Jose, California, USA, vol. 6815, pp. 681510, 2008.

[68] N. Sabbour and F. Shafait, "A segmentation-free approach to Arabic and Urdu OCR," in *Document Recognition and Retrieval XX*. vol. 8658, pp. 86580N, 2013.

[69] S. Naz, A. I. Umar, S. B. Ahmed, R. Ahmad, S. H. Shirazi *et al.,* "Statistical features extraction for character recognition using recurrent neural network," *Pakistan Journal of Statistics*, vol. 34, no. 1, pp. 47–53, 2018.

[70] D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, no. 2, pp. 65–85, 1994.

[71] S. Chanda and U. Pal, "English, Devanagari and Urdu text identification," in *Proc. of the Int. Conf. on Cognition and Recognition*, Mandya, pp. 538–545, 2005.

[72] S. Nazir and A. Javed, "Diacritics recognition based Urdu Nastalique OCR system," *Nucleus*, vol. 51, no. 3, pp. 361–367, 2014.

[73] S. A. Husain, "A multi-tier holistic approach for Urdu Nastaliq recognition," in *Proc. of the 6th Int. Multitopic Conf.*, Karachi, Pakistan, pp. 528–532, 2002.

[74] S. T. Javed and S. Hussain, "Segmentation based Urdu Nastalique OCR," in *Iberoamerican Congress on Pattern Recognition*, Berlin, Heidelberg, vol. 8259, pp. 41–49, 2013.

[75] S. T. Javed, S. Hussain, A. Maqbool, S. Asloob, S. Jamil *et al.,* "Segmentation free Nastalique Urdu OCR," *World Academy of Science, Engineering and Technology*, vol. 46, pp. 456–461, 2010.

[76] M. I. Razzak, F. Anwar, S. A. Husain, A. Belaid and M. Sher, "HMM and fuzzy logic: A hybrid approach for online Urdu script-based languages' character recognition," *Knowledge-Based Systems*, vol. 23, no. 8, pp. 914–923, 2010.

[77] S. A. Husain, A. Sajjad and F. Anwar, "Online Urdu character recognition system," in *Proc. of IAPR Conf. on Machine Vision Applications*, Tokyo, Japan, pp. 98–101, 2007.

[78] N. Javed, S. Shabbir, I. Siddiqi and K. Khurshid, "Classification of Urdu ligatures using convolutional neural networks—A novel approach," in *Int. Conf. on Frontiers of Information Technology*, Islamabad, Pakistan, pp. 93–97, 2017.