IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received September 30, 2019, accepted October 22, 2019, date of publication October 30, 2019,
date of current version November 13, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2950537*

# Two Stream Deep Neural Network for Sequence-Based Urdu Ligature Recognition

**SYED YASSER ARAFAT AND MUHAMMAD JAVED IQBAL**
Department of Computer Science, University of Engineering and Technology (UET), Taxila 47080, Pakistan
Corresponding author: Muhammad Javed Iqbal (javed.iqbal@uettaxila.edu.pk)

**ABSTRACT** Urdu text is a complex cursive script and poses a challenge for recognition by OCR systems due to its large number of ligatures and cursive style. In literature, several techniques have been proposed to recognize Urdu ligatures. However, we have investigated that, suitable challenging datasets and the consequently higher recognition rate is needed for ligature recognition. In this paper, a hybrid model based on the holistic approach is adopted for the recognition of Urdu ligatures (compound characters). More than 3800 unique ligatures were used to generate 46K (38K training, 7K testing) synthetic ligatures with 9 different kinds of transformations along with the normal ligatures. Each ligature is processed through two streams of Deep Neural Networks, namely Alexnet and Vgg16 to obtain a unique set of features corresponding to each net. These features are fused and then used as an input to double layer Bidirectional Long Short Term (BLSTM) network for learning a model. The learned model maps ligature images to their corresponding sequence of individual Urdu characters. In the proposed methodology output is in the editable Urdu-script format. The proposed model was evaluated and have shown an accuracy of 97% on the training dataset and 80% on more than 7K parametrically different query ligatures (test-set).

**INDEX TERMS** BLSTM, classification, deep neural network, Nastalique, optical character recognition (OCR), synthetic Urdu text.

## I. INTRODUCTION

Urdu is the national language of Pakistan and also 6-Indian states [1], Hence covering more than 260 million people. Script recognition is an essential part of any simple/Photo OCR system. Urdu language script is a super-set of the Arabic set. OCRs are generally categorized into two categories: offline systems [1]–[3] and online systems [4], [5]. Offline means at a later stage: essentially recognizing text from printed or photo text, while online means the text is recognized as soon as it is written usually on tablets/smartphones. The paragraph in Urdu script is divided into three sublevels, firstly sentence level, secondly words and thirdly ligature which comprise of a single character or a compound character. An OCR system for the Urdu language has different writing styles for Urdu script/text, multiple size ligatures, and image degradations. Along with these variations, the presence of diacritics in Urdu script results in low recognition rates [6], [7]. Urdu has two main commonly used writing styles i.e., Naskh and Nastalique [8] besides others.

Urdu script recognition poses a challenge due to the following 5-key properties as shown in Fig. 1. These properties are not so prevalent in other European scripts. First, are the multiple shapes of primary characters due to its place in a ligature (compound character). Second is the overlapping of ligatures that makes it hard to segment individual ligatures in words. Segmentation is a key step in certain OCRs for identifying words. The third is the usage of multiple baselines, which changes the angle where different ligatures/characters can be found [9]. Fourth is the position of diacritics, where diacritic positions may change due to the use of different baselines [10]. The fifth is the bidirectional nature of Urdu script writing, i.e., Generally Urdu is written from right to left, however, when we incorporate numbers in Urdu text, these numbers are written left-to-right.

Even more beside above-mentioned properties, use of dual style numerals (Urdu and English), stretching of various words/ligatures [9], non-uniform inter and intra word spacing [11], segmenting Urdu lines and ligatures [3], [12], filled or false loops [13] poses special challenge for recognition of Urdu-script.

Researchers have incorporated various kinds of techniques to overcome the mentioned challenges of Urdu script

The associate editor coordinating the review of this manuscript and approving it for publication was Naveed Akhtar.

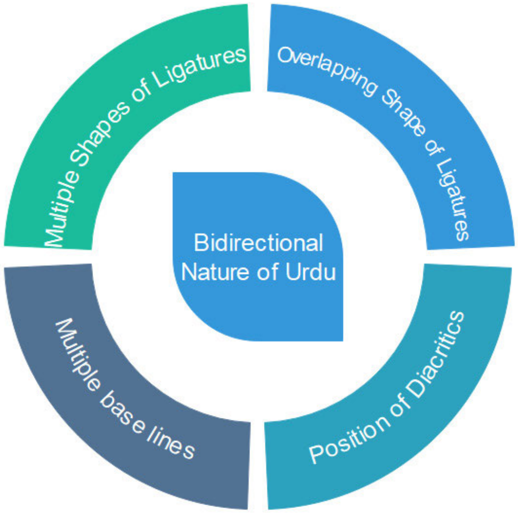

Multiple Shapes of Ligatures
Overlapping Shape of Ligatures
Bidirectional Nature of Urdu
Multiple base lines
Position of Diacritics

**FIGURE 1.** Major challenges in Urdu recognition.

recognition. Zoning features with a combination of 2DLSTM have been used to recognize Urdu text lines from UPTI dataset by Naz *et al.* [14]. In 2019, Ahmed et al. [16] have incorporated 1-d LSTM in their suggested work for the recognition of hand-written characters [15] used sliding windows on UPTI text line images and extracted a set of statistical features. The extracted features are classified using multi-dimensional long short-term memory, along with connectionist temporal classification (CTC) output layer that transcribed the given Urdu character sequences. They reported an accuracy of 96.40 %. For the English language, Huang et al. [17] used 32 × 32 based sliding window CNN features and LSTM for word recognition. A similar kind of end to end methodology using convolutional features and LSTM were adapted for English text recognition by [18]. Similarly in 2018, Cheng *et al.* [19] have developed an end to end technique with the arbitrary orientation network (AON) in their proposed solution, for capturing the character placement features in horizontal and vertical directions.

Our research work is closely related to the recent work of [20] and [21]. The proposed methodology uses more than 3800 unique ligatures generated from CLE [22] ligatures text, with certain geometric transformations giving rise to a total of 46K ligature images. The resultant 46K images are then divided into two parts of 38K synthetic images to train a model and parametrically different 7K images to test the proposed method. It is important to note that, 7K images are more challenging variations of ligature images.

This paper discusses hybrid segmentation free methodology for recognizing Urdu ligatures written in Nastalique style. The main contributions of this paper are as follow;

- The development of a dataset with more than 46K images, containing 38K+ images with 10 variations of 3867 unique ligatures for training and 7K images for testing.
- Development of a technique based on convolutional features of well-known CNNs, that is Alexnet, Vgg16 and a sequence modeling BLSTM to recognize Urdu ligatures as a sequence classification problem.
- Directly editable Urdu text generation, instead of existing roman-type text generation.

The rest of the paper is organized as follows: Section 2 describes the relevant literature and existing datasets. Section 3 discusses the proposed methodology and dataset creation process in detail. Discourse on the experiments and results are given in Section 4. Section 5 concludes the paper with suggested future directions.

## II. RELATED WORK

Ligatures mostly occur in Urdu text as an individual character or a compound character. Many approaches to ligature recognition have been used by researchers [20], [21], [23], but there are two basic methodologies to deal with ligature recognition: segmentation based [24]–[26] and holistic/segmentation-free [21], [27, [28].

Ideally, in any OCR system, script recognition is often done by certain statistical or neural network-based model. Statistical and Convolutional Neural Network (CNN) models can recognize the most of ligatures if it is trained on high quality and variant kind of data. Broadly text recognition methodologies can be divided into three categories based on feature extraction techniques and classifiers.

### A. HAND-CRAFTED FEATURES AND MACHINE LEARNING ALGORITHMS

Pathan *et al.* [29] developed an offline handwritten isolated Urdu character recognition system based on Invariant Moments and Support Vector Machine (SVM) classifier for 800 images. While Khan et al. [30] have deployed simple connected components (CC), feature extraction as an array of coordinates, and image comparison for Urdu recognition. In 2015, Tounsi *et al.* [31] have suggested use Scale Invariant Feature Transform (SIFT) features and SVM to recognize manually segmented Arabic Characters. Shabbir [32] have proposed projection profile features of connected components, Hidden Markov Model (HMM) as a classifier, and reported an accuracy of 92% on (2017 ligatures) a subset of CLE dataset [22]. Gray Level Co-occurrence Matrices (GLCM) [33], [34] based features were extracted by Jamil *et al.* [35], from video images [36] for English, Arabic, Urdu, Chinese and Hindi language images. The text was then detected by using ANN, they also employed Local Binary Pattern (LBP) [37] features for script identification. LBP feature histograms were extracted from detected text regions and finally, ANN employed for classifying the type of script.

### B. HAND-CRAFTED FEATURES AND CNN/LSTM BASED CLASSIFIERS

Naz *et al.* [14], used zoning features, along with 2DLSTM to recognize Urdu text lines on UPTI dataset and reported a recognition rate of 93.39%. Also, Naz *et al.* [15] adapted the sliding windows method on UPTI text line images and extracted a set of statistical features. The extracted features are classified using multi-dimensional long short-term memory recurrent neural network (MDLSTM RNN) along with connectionist temporal classification (CTC) output layer that transcribed the given Urdu character sequences. They reported an accuracy of 96.40%.

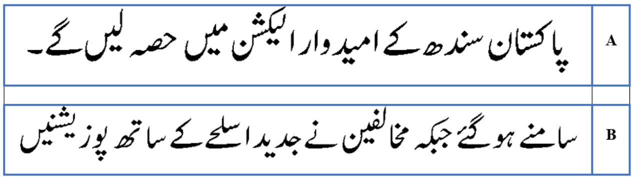
### C. HYBRID/END-TO-END APPROACHES

As early as in 2011, for text detection, an End-to-End Lexicon based system by Wang *et al.* [38], with sliding windows along with Ferns technique can be found. The authors of this study have shown promising results on ICDAR03 [39] and SVT [39] datasets. In 2013 Ul-Hasan *et al.* [40] used raw pixel windows of 30×1 to BLSTM, and CTC layer for transcription purpose and achieved an accuracy of 94.85% on UPTI dataset. In 2017 Ahmad et al. [28] proposed stacked denoising autoencoder (SDA) for automatic feature extraction from Urdu ligature images raw pixel and 2-stage SDA are trained for recognition, they reported recognition accuracy of 93-to-96%. Shi *et al.* [18] have developed an end-to-end system for English text, consisting of three layers namely novel Deep Convolutional Neural Network, Recurrent Neural Network layer and Transcription layer. The system was tested on ICDAR03 [41] dataset and reported an accuracy 95.9%. Uddin *et al.* [60] have divided a query ligature into primary and secondary ligatures and processed separately by CNN for recognition. And the results of both are later associated in a postprocessing step to predict the final complete ligature. They reported good results on both CLE and UPTI. A similar kind of study on 38K ligatures images for 98 classes scanned from the book, was done by Javed *et al.* [61], they used fixed-size ligatures of 55 × 55 pixels and reported an accuracy of 95%. Also, Rafeeq *et al.* [21] showed that a deep neural network and clustering of ligatures significantly enhances the classification accuracy on the custom generated a dataset of 17010 Urdu ligatures from CLE [22] in 2018. He et al. [7] reported an accuracy of 98.12% on UPTI dataset using CNN and MDLSTM.

### D. EXISTING DATASETS

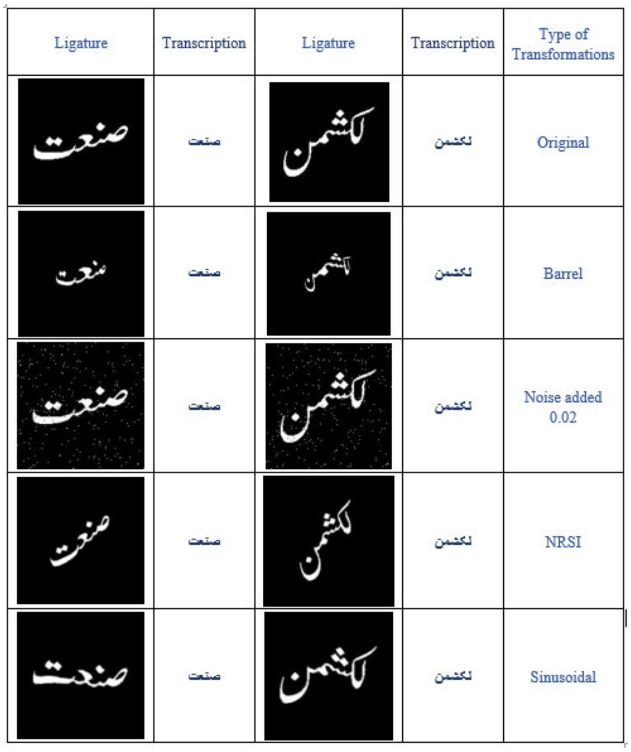
Besides custom datasets, notable datasets used for Urdu text recognition are the Urdu Printed Text Images UPTI [44] and Center of Language Engineering CLE [22]. UPTI contains printed Urdu Nastalique text of more than 10K lines, while CLE data set contains more than 18K unique ligatures. Sample text lines from UPTI dataset are shown in Fig. 2. Also, few ligatures images which were synthetically generated and normalized from CLE ligatures text are shown in Fig. 3.

Various researchers have used a subset of these datasets and their synthetic variations [1], [11], [20], [21], [23], [24], [28], [32], [42], [43]. Akram and Hussain [43] have presented a technique that took various font size document images as an input to make font-size independent OCR system. Ahmad et al. [20] determined 3604 classes of ligatures from UPTI dataset containing 10K sentences. A few recent notable contributions are summarized in Table 1. It can be seen in Table 1 that, the number of unique ligatures varies from as low as 250 at one end to high as 3867 at the other end.



| | |
|---|---|
| پاکستان سندھ کے امیدوار الیکشن میں حصہ لیں گے۔ | A |
| سامنے ہوگئے جبکہ مخالفین نے جدید اسلحے کے ساتھ پوزیشنیں | B |

**FIGURE 2.** Urdu text sample from UPTI.



| Ligature | Transcription | Ligature | Transcription | Type of Transformations |
|---|---|---|---|---|
| صنعت | صنعت | لکشمن | لکشمن | Original |
| صنعت | صنعت | لکشمن | لکشمن | Barrel |
| صنعت | صنعت | لکشمن | لکشمن | Noise added 0.02 |
| صنعت | صنعت | لکشمن | لکشمن | NRSI |
| صنعت | صنعت | لکشمن | لکشمن | Sinusoidal |

**FIGURE 3.** Synthetically generated ligature variations.

**TABLE 1.** Summary of proposed and recent contributions in Urdu OCR.

| Study | No of Unique Ligatures | Approach | Dataset | Accuracy |
|---|---|---|---|---|
| [21] | 2430 | holistic | CLE | 95.2% |
| [42] | 1525 | holistic | UPTI | 92.00% |
| [28] | 3732 | holistic | UPTI | 96.00% |
| [1] | 1845 | holistic | Custom | 90.29% |
| [11] | 2430 | holistic | Custom | 90.00% |
| [32] | 250 | Analytic | Custom | 92.00% |
| [24] | 1692 | Analytic | Custom | 92.73% |
| [23] | 2028 | holistic | CLE + Custom | 97.73% |
| [43] | 1965 | holistic | Custom | 96.66% |
| [20] | 3604 | holistic | UPTI | 96.71% |
| **Proposed** | 3867 | holistic | CLE | 97.17% |

## III. PROPOSED METHODOLOGY

Proposed research presents a holistic segmentation free methodology for ligature classification. Two existing CNN networks, Alexnet [45] and Vgg16 [46] are suggested as feature extractors from ligature images. These two CNNs are well known for their top recognition performance in imagenet ILSRC challenge. The purpose of using these two top CNNs is to extract maximum unique features from each query image. For showing the uniqueness of extracted features of a given image, the activation feature maps obtained by using the convolution-5 layer of both Alexnet and vgg16 is shown in Fig 4. Four ligatures 'بیں', 'ہے', 'ہو', and 'ند' ligatures are taken as input query image and the next 2-columns show their activations corresponding to Alexnet and Vgg16 respectively. Both nets react differently to similar input. Each network captures a unique feature-fingerprint about a ligature than the other. Pseudo colored images highlight unique features weightage.

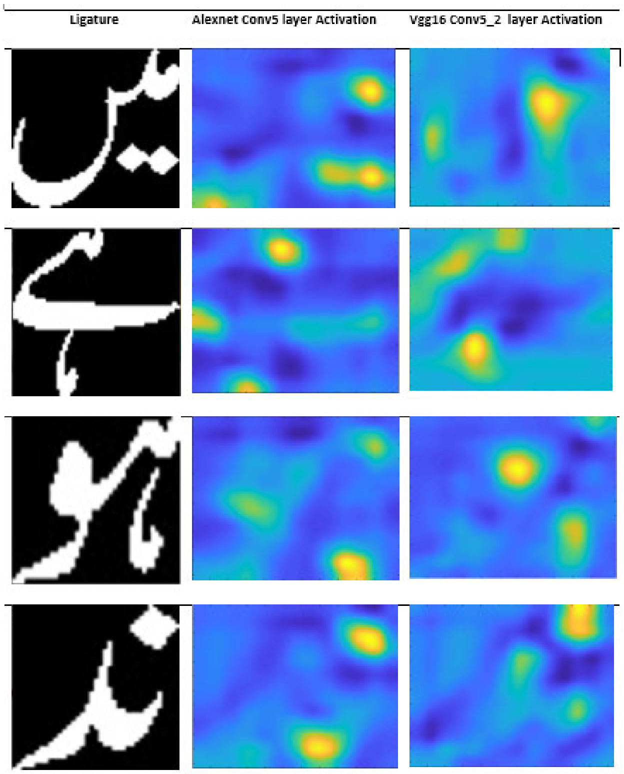


**FIGURE 4.** Ligatures and their corresponding activations at conv5/conv5_2 layer of Alexnet and Vgg16.

Each CNN gave a unique feature vector of 4096 values for the given input ligature image. Both feature vectors were fused to get 8192-size hybrid vector. In hybrid vector, 16 zeros were added to make it a vector of 8208 values. This new vector was rearranged as image-grid pattern of size 18X 456 as shown in Fig. 5. The grid serves as a mechanism for distributing the CNN-features to be explicitly arranged in the spatial dimension. Grid arrangement corresponds to different parts of the given ligature image features. Double layer BLSTM was used as a sequence classifier, that learns the correspondence between target sequence characters and spatial-features grid.

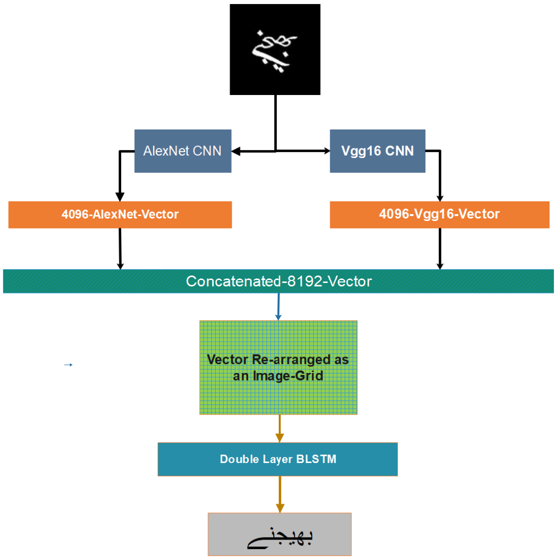


**FIGURE 5.** Proposed model for ligature recognition.

Double layer BLSTM contained a total of 1200 cells. Each ligature was considered as a maximum sequence of 18 basic characters. As ligatures can have 1-2 or more primary characters, the rest of the values in the train sequences is considered empty. Hence learning the presence of emptiness is also an implicit part of learning by double-layer BLSTM. Our focus was on improving recognition accuracy and it is well known that Alexnet and Vgg16 are the winner of imagenet challenge 2012 and 2014 respectively. The key inspiration for Feature-Fusion is from the study [47] by Zhou *et al.*, wherein the images, objects localization can be done based on the activation map of the neural network. In [48] authors have presented a 2-stream network for action recognition and Ullah et al. [49] have developed a linear combination of CNN and BLSTM for action recognition in video sequences. While in [50] Yang et al. have discussed 3-stream network structure with a 3-ConvNets i.e., spatial ConvNet, a pixel-level temporal ConvNet and a block-level temporal ConvNet for video action recognition.

As can be seen from Fig. 4, in our study, activation maps of each Alexnet and vgg16, activate on different regions of the same ligature. It means each net is prioritizing different parts of ligature images. So, by feature fusion, we are maximizing the parts which can be learned and predicted by LSTM in the next stage. Definitely, the system can be evaluated using mono-stream or single network, and time comparisons can be done, which may need a separate set of experiments and may need more time.

### A. DATASET PREPARATION

Data is the key thing to train any classifier presence. Neural networks need a lot and variant type of data to train, so we need a dataset which covers variations in input data. The most similar dataset is CLI compared to our dataset and consists of just a single image for each ligature and different aspect ratios. While UPTI has sentence-level images and others have their custom images. Our dataset contains multiple images of the same ligature with various kinds of real-world image transformations.

For the proposed model, we have generated more than 46 thousand (46K) images of more than 3800 unique ligatures/classes inspired by CLE [22]. For each Ligature 9-transformed images, plus the original ligature image is generated and saved. The proposed methodology uses 2 more transformations per ligature as compared to [21]. The 9-types of transformations that were applied to generate synthetic images were, Non-Reflective similarity (NRSI), Barrel, Sinusoidal, 2-types of noise Sigma $\sigma$ in range (0.01,0.02), 4-rotation transformations with angles in range ($-7$, $+7$, $-16$, $+16$). For the testing purpose, 7330 images with random transformations were generated with different parametric variations of angles, scales and sigma values than the originally generated synthetic images for training. Hence, the testing set is a kind of challenging nature in the proposed study.

#### 1) NON-REFLECTIVE SIMILARITY (NRSI) TRANSFORMATION

Non-reflective similarity transform is a subset of affine transformations, which depends on the scale and angle parameters. In this case, the translation along x-axis and y-axis is zero. Mathematically

$$[u\ v] = [u\ v\ 1]\,T \tag{1}$$

where T is a transformation matrix in Eq. 1, built on scale and angle values. Here u and v are two axis points need to be determined.

#### 2) BARREL TRANSFORMATION

Barrel Transformation transforms an image radially outward from its center. The distortion introduced by barrel transformation is greater at the end of the image than that is at the center. Hence resulting in convex side shapes. It is generally achieved by converting cartesian x- and y-coordinates to polar coordinates $\theta$ and r. Here $\theta$ represents the angle and r represents the radius. r changes linearly as distance from the center pixel increases. Mathematically it can be described by Eq. 2.

$$bt = r + r^3 * (a/rmax^2) \tag{2}$$

Here **a** is an amplitude of the cubic term. This parameter is adjustable. **bt** and $\theta$ are converted back using polar to cartesian transformation.

#### 3) SINUSOIDAL TRANSFORMATION

As its name represents, Sinusoidal transformation is the process of applying a sinusoidal wave pattern on image and generating a wavy pattern shape image. In sinusoidal transformation, the x-coordinate position of pixels is unchanged in the image, while the y-coordinate position of rows of pixels are shifted up or down depending on the wave pattern. Mathematically sinusoidal transformation can be written as shown in Eq. 3.

$$st = y + a^* \sin\left(2^* \pi^* x/\alpha\right) \tag{3}$$

Here y is a column, x is a row value and a is an amplitude, $\alpha$ is the number of rows. While sin represents the sine function.

#### 4) TYPES OF NOISE

2-types of noise introduce in generated ligature images. Sigma's ($\sigma$) value of 0.01 and 0.02 are used for adding 'salt and pepper' noise to images. Here 0.02 represents the density of noise and in this case, it affects approximately 2% of pixels of ligature.

### B. FEATURE EXTRACTION AND CONVERSION

Convolutional Neural Networks (CNNs) or Deep Neural Networks (DNNs) are a special class of neural networks, that are designed to identify, locate and recognize visual features directly from image pixels. Two high performance deep neural networks, Alexnet [45] and Vgg16 [46] were used as a feature extractor. Each network gave a feature vector of 4096 values, which were later fused to get 8192 feature-vector and expanded to get 8208 size vector.

#### 1) ALEXNET

Alexnet was designed by Alex Krizhevsky. This net competed in the ImageNet Large Scale Visual Recognition (ILSVRC) Challenge, in 2012. The Alexnet has a very similar architecture, as LeNet [51] but is with more layers, with more filters per layer and filters of 11 × 11, 5 × 5, 3 × 3, were used. It also improved the usage of a particular type of layers with different functionality such as CONV, ReLU activations, Max pooling, Dropout. The network achieved a top-5 error from 26% to 15.3%. The Alexnet, has been cited over 30K times. The architecture of Alexnet can be found in the study [45].

#### 2) VGG16

Vgg16 is developed by Visual Geometry Group (VGG) from University of Oxford, and was the 1st runner-up, in ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2014 in the classification task, and it beats the GoogLeNet in the localization task of ILSVRC 2014. Vgg16 consists of 16 convolutional layers. Its architecture is like Alexnet, but it uses only 3 × 3 convolutions and a large number of filters. Vgg16 consists of 138 million parameters to be learned. Hence computationally expensive to train. The architecture of Vgg16 can be found in the study [52].

#### 3) LSTM

Long short-term memory (LSTM) cells are units of a recurrent neural network (RNN). Few common variations of the LSTM are Bidirectional LSTM (BLSTM), multi-dimensional long short-term memory (MDLSTM). LSTM [53] networks

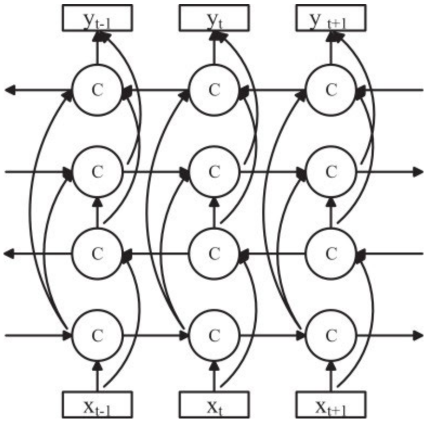
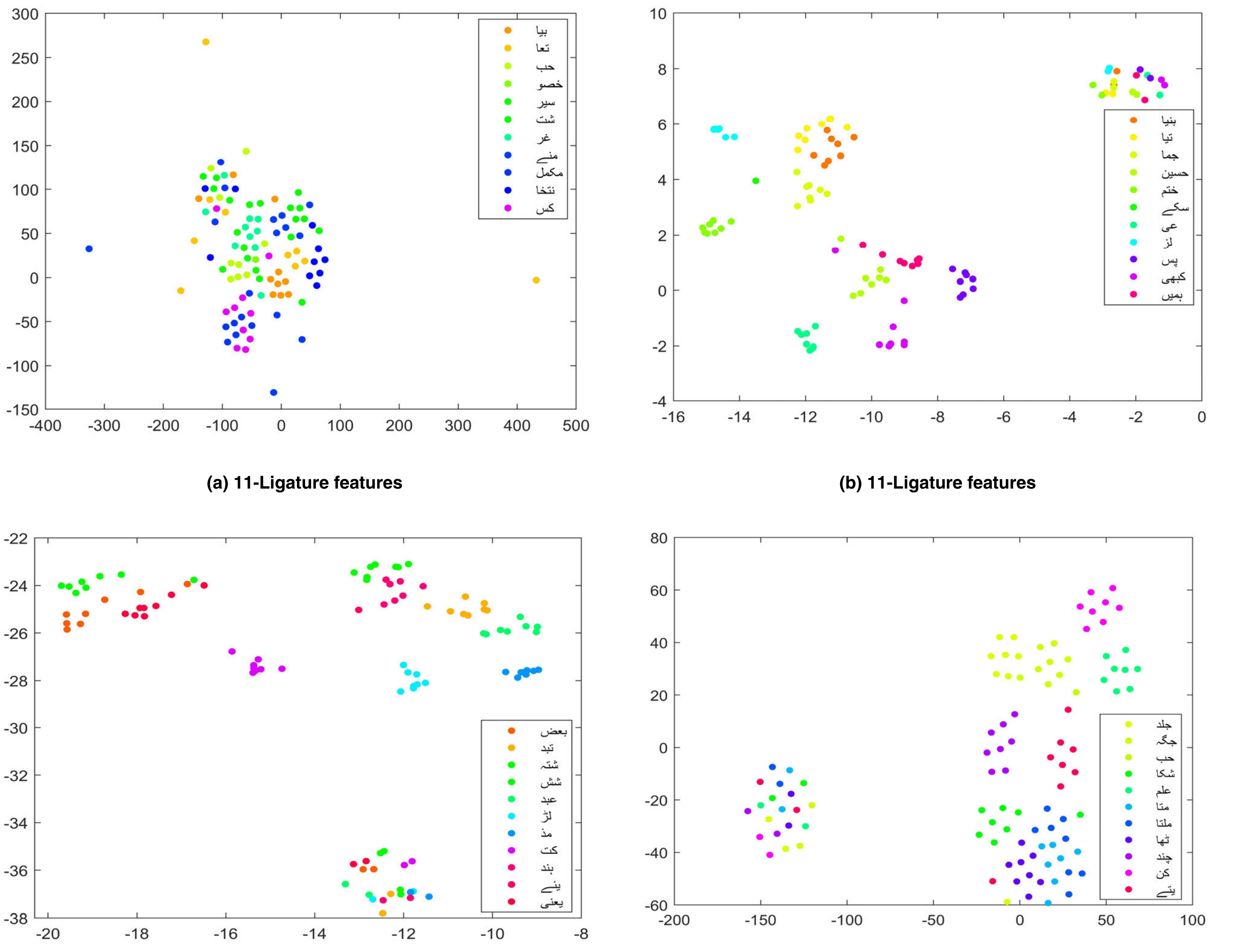


**FIGURE 6.** Internal architecture of double-layer BLSTM [57].

are well-known to classifying, processing and making predictions based on time series data. Fig. 6 shows the double-layer LSTM, which have been used in our proposed methodology.

### C. VISUALIZATION OF FEATURES

The feature for each ligature image is of length 8192, which was obtained after applying Alexnet and Vgg16. For understanding the overlapping of features or non-linearity of obtained features vectors t-SNE [54] was applied. t-SNE is a well-known technique for dimensionality reduction and is often used for the visualization of high-dimensional data [55], [56]. Fig. 7 (a) shows only 11 ligatures-images features-mapping. It shows that feature vectors namely of ‘ بیا’ and ‘ تعا’ features are difficult to separate along with features of ‘سیر’ and ‘غر’. Fig. 7 (b) shows that ligatures ‘ بنیا’, ‘ تیا’, and ‘ جما’ features are dispersed and difficult to separate. Fig. 7 (c) shows ligatures ‘ شتہ’ and ‘شش’ features are not easily linearly separable beside other ligature features. Similarly, Fig. 7 (d), shows ligatures ‘متا’ and ‘ ملتا’ are difficult to separate while ‘چند’ and ‘کن ’ ligatures have easily discernable features. In these 4-sub figures of Fig. 7, a total of 44-ligatures features in 2D space are shown. Overlapping of features show



(a) 11-Ligature features

(b) 11-Ligature features

(c) 11-Ligature features

(d) 11-Ligature features

**FIGURE 7.** Non-linearity of ligatures features in 2D.

nonlinearity of data and consequently a challenging classification problem.

### D. CLASSIFICATION

The proposed methodology considers features extracted in earlier steps, as a sequence of inputs Xi, and image labels as a sequence of required target vectors as Yi. In our case, Yi has a maximum length of 18 primary Urdu characters inclusive of spaces. If a ligature length is less than 18 characters, then the remaining space is treated as a space character. Table 2 and Table 3 succinctly describe the working of the proposed methodology in the form of pseudocode, where steps in both training and testing phases are described. In pseudocode ƛ, ϖ and $\xi$ show Alexnet features, Vgg16 features, and image labels respectively.

**TABLE 2.** System training pseudocode.

```
Pseudocode for Training
I)  For each training image ii in → NN(TrainSet)
        a)  Extract AlexNet features → $\lambda ii$
        b)  Extract Vgg16 features → $\varpi ii$
        c)  Extract image label ligatures → $\xi ii$
        d)  Concatenate step-a & step-b features → $[\lambda ii \ \varpi ii]$
        e)  Standardize length of step-c features → $\eta([\lambda ii \ \varpi ii])$
        f)  Convert step-e features to a Model compatible
            format → $\Psi_1(\eta([\lambda ii \ \varpi ii]))$
        g)  Convert the label of the image to a Model compatible
            format → $\Psi 2(\xi ii)$
II) Train the Model on NN images, using $\Psi_1$ $\Psi_2$
```

**TABLE 3.** System testing pseudocode.

```
Pseudocode for Testing
For each query image I → TS (TestSet)
        a)  Extract AlexNet features → $\lambda I$
        b)  Extract Vgg16 features → $\varpi I$
        c)  Concatenate step-a & step-b features → $[\lambda I \ \varpi I]$
        d)  Standardize length of step-c features → $\eta([\lambda I \ \varpi I])$
        e)  Convert step-d features to a Model compatible format
            → $\Psi_1(\eta([\lambda I \ \varpi I]))$
        f)  Predict the sequence (decode) using Trained Model
                $PredSeq \leftarrow \rho(Net, \Psi_1(\eta([\lambda I \ \varpi I])))$
        g)  Convert a predicted sequence of step-f to Unicode →
            $\chi(PredSeq)$
        h)  Editable Ligature Generation
```

Where $\psi 1$ represents a function that takes the input features and converts them to model consistent format, while $\psi 2$ is function similar to $\psi 1$ in functionality, which deals with image labels compatibility. $\eta$ represents functions for standardizing the input features. The model was trained using the piece-wise learning rate, with an initial value of 0.01. The drop factor for decreasing the learning rate was defined for every 20 epochs. The model was trained for 100 epochs.

## IV. RESULTS AND DISCUSSION

The proposed method was implemented using Matlab and GPU based hardware. The system was trained on more than 38K ligatures images. In the experiments, we have used a large number of unique ligatures as compared to other similar studies, as shown in Table 1. Also Fig. 7's different subfigures, shows that just 11-ligatures can have great overlapping of features which is difficult to classify using traditional classifiers.

Two types of experiments were performed to evaluate the accuracy of the learned model. One is on trained-set, while another one with similar transformations (test-set) but with different parametric values. Ligatures are considered as a sequence of primary characters in Urdu script, recognition rate based on exact or whole sequence matching was performed. Sometimes sequence may match partially or differ by only one character, they are not considered a positive match for exact or one-to-one match. So, the partial sequences were matched by Levenshtein-distance [58], also sometimes known as edit-distance [59]. Table 4 shows the difference between actual and partial matching mechanism and how it affects accuracy. The exact match happens if each character or characters appear in the correct order and are similar.

**TABLE 4.** Comparison of two accuracy measures for ligature recognition.

| Type of Text | Ligature | Exact Match Accuracy | Ligature | Partial Match Accuracy |
|---|---|---|---|---|
| True Text ---------→ | جی | - | جی | - |
| Predicted text ----→ | کی | 0 % | کی | 50% |
| True Text ---------→ | لکشمن | - | لکشمن | - |
| Predicted text ----→ | لکسمن<br>(لکسمن) | 0% | لکسمن | 80% |

While in partial matching, a wrongly recognized character is characterized as a minor mistake or percentage of all given ligatures.

Correctly recognized characters contribute towards overall accuracy in partial matching, hence results in higher overall recognition rate, than exact match.

The proposed method is unique as compared to [21] because they have used clustering to reduce the search space. Deciding the final cluster size is a kind of hit and trial parameter, that may depend on certain intuitiveness and observations. While in our approach there is no need to do clustering.

Table 1 shows, that various researchers have used different kinds of datasets for evaluation of their proposed methodologies. These studies have used a different dataset and even different number of subset ligatures from the same dataset, so fair comparison is difficult as found in other studies for English or Chinese languages due to standard datasets. The common approach to handle ligature recognition is holistic, as it implicitly extracts relevant features without

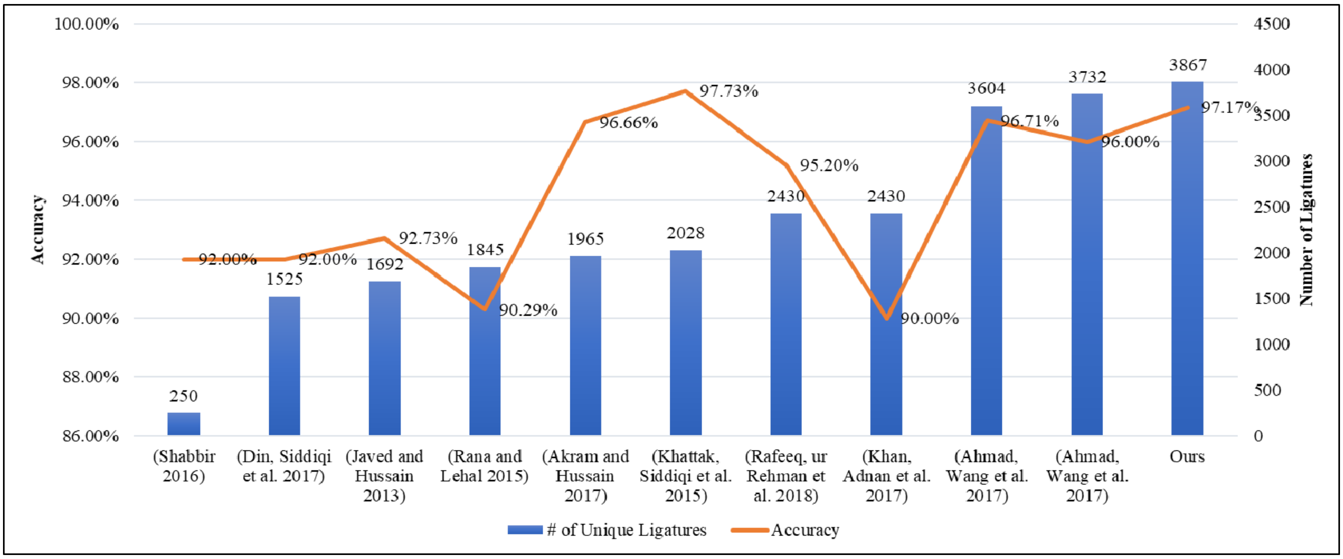


**FIGURE 8.** Accuracy comparison of proposed and previous methods.

decomposing the original ligature. To date, to the best of our knowledge, we have used the maximum number of 3867 unique ligatures for experimentation. To evaluate our methodology, both kinds of accuracies/evaluation, i.e., on train-set and test-set, are necessary to know whether any overfitting in the learned model is there or not.

Although, Khattak *et al.* [23] have achieved an accuracy of 97.73% on CLE but using only 2028 unique ligatures. The proposed methodology achieves comparable accuracy on more than 3800 unique ligatures. A total of 46000 ligature images were processed by the system. The 97.17% partial accuracy for trainset indicates that the system is well-tuned on trained images as shown in Table 5. While an exact sequence recognition rate of 66.33% is achieved. Similarly, for challenging test-set of 7K images were having partial-sequence accuracy of 80.46% and an exact match of 3.65% achieved. Test-set images generated are more variant in terms of parametric variation than 9-type of variations including original images in the train-set. Test-set (as it contains parametric transformations, which vary from train-set) highlights the importance of more image variations. Overfitting may be perceived, or it appears so, but it is due to the challenging test-set images. Fig. 8 compares our results with other existing studies, shows the best results achieved in terms of accuracy by using the large number of unique ligatures.

**TABLE 5.** Comparison of recognition rate for proposed dataset.

| Ligatures Type | No of Total Ligatures | Exact-Sequence Recognition rate | Partial-Sequence Recognition Rate |
|---|---|---|---|
| Trained-Set | 38670 (3.8K) | 66.33% | **97.17%** |
| Test-Set (with random similar transformations) | 7330 (7.3K) | 3.65% | **80.46%** |

It shows further, that large size dataset(s) is needed to improve the learned model to handle more variations in images. Hence more robust datasets will consequently improve recognition rate and can make a robust recognition system. In the future, large size datasets with other types of transformations, such as more angles, font colors, background variations are vehemently required for any successful Urdu OCR. These improvements are necessary for datasets for any real-world useful Urdu OCR applications.

## V. CONCLUSION

In this paper, a hybrid holistic ligature-based recognition system using ligatures as a sequence of characters for learning and classifying images has been presented. Two kinds of synthetic images were generated from the ligatures text of CLE dataset, one for training and another for testing. A total of 46K images with at least 9 variations of each ligature were generated. Image features were extracted using two famous CNNs, namely Alexnet, Vgg16 and transformed into an image like a grid pattern. We have performed t-SNE visualization to understand the complexity of the dataset. Finally, double-layer BLSTM was used as a sequence classifier, which was trained on fused features of two CNNs. Results indicate that the system learned well on the trained set, but on challenging test-set it shows comparably low performance. In the future, this work may be extended to word-level script recognition and can be extended to different writing styles. Other neural network models, such as Resnet, Inception, mobilenet, and Squeezenet can be used in various combinations for Urdu text feature extraction. Subsequently, a comparative analysis of accuracies can be performed.

## ACKNOWLEDGMENT

The authors would like to thank UET, Taxila, for supporting this work.

## REFERENCES

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

[2] D. A. Satti and K. Saleem, "Complexities and implementation challenges in offline urdu Nastaliq OCR," in *Proc. Conf. Lang. Technol.*, 2012, pp. 85–91.

[3] S. A. Malik, M. Maqsood, F. Aadil, and M. F. Khan, "An efficient segmentation technique for urdu optical character recognizer (OCR)," in *Proc. Future Inf. Commun. Conf.* Cham, Switzerland: Springer, 2019, pp. 131–141.

[4] F. Anwar, M. A. Aftab, S. A. Hussain, and A. Hussain, "Preprocessing of online Urdu handwriting for mobile devices," *Int. J. Comput. Sci. Netw. Secur.*, vol. 17, no. 10, pp. 173–178, 2017.

[5] R. Kaur, "Text recognition applications for mobile devices," *J. Global Res. Comput. Sci.*, vol. 9, no. 4, pp. 20–24, 2018.

[6] G. S. Lehal and A. Rana, "Recognition of Nastalique Urdu ligatures," in *Proc. 4th Int. Workshop Multilingual OCR*, Aug. 2013, p. 7.

[7] S. Naz, A. I. Umar, R. Ahmad, I. Siddiqi, S. B. Ahmed, M. I. Razzak, and F. Shafait, "Urdu Nastaliq recognition using convolutional–recursive deep learning," *Neurocomputing*, vol. 243, pp. 80–87, Jun. 2017.

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

[9] G. Kaur, S. Singh, and A. Kumar, "Urdu ligature recognition techniques-A review," in *Proc. Int. Conf. Intell. Commun. Comput. Techn. (ICCT)*, Dec. 2017, pp. 285–291.

[10] A. F. Ganai and F. R. Lone, "Character segmentation for Nastaleeq URDU OCR: A review," in *Proc. Int. Conf. Elect., Electron., Optim. Techn. (ICEEOT)*, Mar. 2016, pp. 1489–1493.

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

[12] I. U. Din, Z. Malik, I. Siddiqi, and S. Khalid, "Line and ligature segmentation in printed Urdu document images," *J. Appl. Environ. Biol. Sci.*, vol. 6, no. 3S, pp. 114–120, 2016.

[13] C. Boufenar, M. Batouche, and M. Schoenauer, "An artificial immune system for offline isolated handwritten arabic character recognition," *Evolving Syst.*, vol. 9, no. 1, pp. 25–41, 2018.

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

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

[16] S. B. Ahmed, S. Naz, S. Swati, and M. I. Razzak, "Handwritten Urdu character recognition using one-dimensional BLSTM classifier," *Neural Comput. Appl.*, vol. 31, no. 4, pp. 1143–1151, 2017.

[17] P. He, W. Huang, Y. Qiao, C. C. Loy, and X. Tang, "Reading scene text in deep convolutional sequences," in *Proc. 13th AAAI Conf. Artif. Intell.*, Feb. 2016, pp. 3501–3508.

[18] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2298–2304, Nov. 2017.

[19] Z. Cheng, Y. Xu, F. Bai, Y. Niu, S. Pu, and S. Zhou, "AON: Towards arbitrarily-oriented text recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5571–5579.

[20] I. Ahmad, X. Wang, Y. H. Mao, G. Liu, H. Ahmad, and R. Ullah, "Ligature based Urdu Nastaleeq sentence recognition using gated bidirectional long short term memory," *Cluster Comput.*, vol. 21, no. 1, pp. 703–714, 2018.

[21] M. J. Rafeeq, Z. ur Rehman, A. Khan, I. A. Khan, and W. Jadoon, "Ligature categorization based Nastaliq Urdu recognition using deep neural networks," *Comput. Math. Org. Theory*, vol. 25, no. 2, pp. 184–195, 2019.

[22] (2010). *Valid Ligatures of Urdu (CLE)*. Accessed: Aug. 10, 2018. [Online]. Available: http://www.cle.org.pk/software/ling_resources/UrduLigatures.htm

[23] I. U. Khattak, I. Siddiqi, S. Khalid, and C. Djeddi, "Recognition of Urdu ligatures–A holistic approach," in *Proc. 13th Int. Conf. Document Anal. Recognit.*, Aug. 2015, pp. 71–75.

[24] S. T. Javed and S. Hussain, "Segmentation based Urdu Nastalique OCR," in *Iberoamerican Congress on Pattern Recognition*. Berlin, Germany: Springer, 2013, pp. 41–49.

[25] S. Hussain, S. Ali, and Q. ul Ain Akram, "Nastalique segmentation-based approach for Urdu OCR," *Int. J. Document Anal. Recognit.*, vol. 18, no. 4, pp. 357–374, 2015.

[26] A. Mahmood and A. Srivastava, "A novel segmentation technique for Urdu type-written text," in *Proc. Recent Adv. Eng., Technol. Comput. Sci.*, Feb. 2018, pp. 1–5.

[27] I. Uddin, I. Siddiqi, and S. Khalid, "A holistic approach for recognition of complete Urdu ligatures using hidden Markov models," in *Proc. Int. Conf. Frontiers Inf. Technol.*, Dec. 2017, pp. 155–160.

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

[29] I. K. Pathan, A. A. Ali, and R. Ramteke, "Recognition of offline handwritten isolated Urdu character," *Adv. Comput. Res.*, vol. 4, no. 1, pp. 117–121, 2012.

[30] E. R. Q. Khan and E. W. Q. Khan, "Urdu optical character recognition technique for Jameel Noori Nastaleeq script," *J. Independ. Stud. Res.*, vol. 13, no. 1, pp. 81–86, 2015.

[31] M. Tounsi, I. Moalla, A. M. Alimi, and F. Lebouregois, "Arabic characters recognition in natural scenes using sparse coding for feature representations," in *Proc. 13th Int. Conf. Document Anal. Recognit.*, Aug. 2015, pp. 1036–1040.

[32] S. Shabbir, "Optical character recognition system for Urdu words in Nastaliq font," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 5, pp. 567–576, 2016.

[33] R. M. Haralick and K. Shanmugam, "Textural features for image classification," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973.

[34] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*. Reading, MA, USA: Addison-Wesley, 1992.

[35] A. J. Jamil, A. Batool, Z. Malik, A. Mirza, and I. Siddiqi, "Multilingual artificial text extraction and script identification from video images," Bahria Univ., Islamabad, Pakistan, Tech. Rep., 2016.

[36] N. Image Processing Center IPC (2012). *IPC-Artificial Text Data Set*. [Online]. Available: https://sites.google.com/ site/artificialtextdataset

[37] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[38] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1457–1464.

[39] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, "ICDAR 2003 robust reading competitions," in *Proc. 7th Int. Conf. Document Anal. Recognit.*, Aug. 2003, pp. 682–687.

[40] 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. 12th Int. Conf. Document Anal. Recognit.*, vol. 2013, pp. 1061–1065.

[41] P. S. Lucas. *ICDAR 2003 Robust Reading Competitions*. Accessed: Jan. 20, 2019. [Online]. Available: http://www.iapr-tc11.org/mediawiki/index.php/ICDAR_2003_Robust_Reading_Competitions

[42] I. U. Din, I. Siddiqi, S. Khalid, and T. Azam, "Segmentation-free optical character recognition for printed Urdu text," *EURASIP J. Image Video Process.*, vol. 1, no. 1, p. 62, 2017.

[43] Q. U. A. Akram and S. Hussain, "Ligature-based font size independent OCR for Noori Nastalique writing style," in *Proc. 1st Int. Workshop Arabic Script Anal. Recognit.*, Apr. 2017, pp. 129–133.

[44] N. Sabbour and F. Shafait, "A segmentation-free approach to Arabic and Urdu OCR," *Proc. SPIE, Document Recognit. Retr. XX*, vol. 8658, 2013, Art. no. 86580N.

[45] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[46] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[47] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2921–2929.

[48] Y. Zhu, Z. Lan, S. Newsam, and A. Hauptmann, "Hidden two-stream convolutional networks for action recognition," in *Proc. Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2018, pp. 363–378.

[49] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, "Action recognition in video sequences using deep bi-directional LSTM with CNN features," *IEEE Access*, vol. 6, pp. 1155–1166, 2017.

[50] W. Yang, S. Gao, W. Liu, and X. Ji, "3-Stream convolutional networks for video action recognition with hybrid motion field," in *Proc. IEEE 20th Int. Workshop Multimedia Signal Process.*, Aug. 2018, pp. 1–6.

[51] Y. Bengio and Y. LeCun, "Scaling learning algorithms towards AI," *Large-Scale Kernel Mach.*, vol. 34, no. 5, pp. 1–41, 2007.

[52] S. Das. (Jan. 30, 2019). *CNN Architectures: LeNet, AlexNet, VGG, GoogLeNet, ResNet and MoreâĂę*. https://medium.com/@sidereal/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5

[53] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," in *Proc. 9th Int. Conf. Artif. Neural Netw.*, 1999, pp. 850–855.

[54] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[55] J. Tang, J. Liu, M. Zhang, and Q. Mei, "Visualizing large-scale and high-dimensional data," in *Proc. 25th Int. Conf. World Wide Web*, 2016, pp. 287–297.

[56] E. Becht, "Dimensionality reduction for visualizing single-cell data using UMAP," *Nature Biotechnol.*, vol. 37, no. 1, p. 38, 2019.

[57] Z.-C. Liu, Z.-H. Ling, and L.-R. Dai, "Articulatory-to-acoustic conversion using BLSTM-RNNs with augmented input representation," *Speech Commun.*, vol. 99, pp. 161–172, May 2018.

[58] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet Phys. doklady*, vol. 10, no. 8, pp. 707–710, 1966.

[59] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 26, no. 1, pp. 43–49, Feb. 1978.

[60] I. Uddin, N. Javed, I. A. Siddiqi, S. Khalid, and K. Khurshid, "Recognition of printed Urdu ligatures using convolutional neural networks," *J. Electron. Imag.*, vol. 28, no. 3, 2019, Art. no. 033004.

[61] N. Javed, S. Shabbir, I. Siddiqi, and K. Khurshid, "Classification of Urdu ligatures using convolutional neural networks–A novel approach," in *Proc. Int. Conf. Frontiers Inf. Technol.*, Dec. 2017, pp. 93–97.

**SYED YASSER ARAFAT** received the MSCS degree from the International Islamic University (IIU), Islamabad, in 2007. He is currently pursuing the Ph.D. degree with the Department of Computer Science, UET Taxila, with research on outdoor Urdu-Text detection and recognition. He is also working as an Assistant Professor with the Department of Computer Science and Information Technology (CS&IT), Mirpur University of Science and Technology (MUST). He has more than 14 years of teaching experience at various National Universities. His research interests include NLP, computer-vision, deep learning, and robotics.

**MUHAMMAD JAVED IQBAL** received the M.Sc. degree in computer science from the University of Agriculture, Faisalabad, Pakistan, in 2001, the M.S./M.Phil. degrees in computer science from International Islamic University Islamabad, Pakistan, in 2008, and the Ph.D. degree in computer science/information technology from Universiti Teknologi PETRONAS, Malaysia, in February 2015. He is currently a HEC approved Ph.D. Supervisor and also an Assistant Professor with the Computer Science Department, University of Engineering and Technology Taxila, Pakistan. After completion of his doctoral studies, he has been actively involved in research. He has more than 20 international publications which includes four ISI indexed impact factor journals, one book chapter Springer LNEE, and ten Scopus-indexed conferences. He is also a Guest Editor of the *Journal of Internet Technology* (Indexed by SCI-E) and a Reviewer of renowned national and international journals and conferences. His research interests include machine learning, data science, pattern recognition, computational intelligence algorithms for biological data classification, bioinformatics, and big data mining.