IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 8, 2022, accepted May 25, 2022, date of publication May 30, 2022, date of current version June 8, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3179366

Recognition of Visual Arabic Scripting News Ticker From Broadcast Stream

MOEEN TAYYAB[1], AYYAZ HUSSAIN[2], MOHAMMED ALI ALSHARA[3], SHAKIR KHAN[3], REEMIAH MUNEER ALOTAIBI[3], AND ABDUL RAUF BAIG[3]

[1]Department of Computer Science and Software Engineering, International Islamic University, Islamabad 44000, Pakistan
[2]Department of Computer Science, Quaid-i-Azam University, Islamabad 44000, Pakistan
[3]Department of Computer Science, College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University, Riyadh 11564, Saudi Arabia

Corresponding author: Abdul Rauf Baig (abbaig@imamu.edu.sa)

This work was supported by the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University through Research Group under Grant RG-21-07-08.

ABSTRACT News ticker recognition is a vital area of research due to its applications such as information analysis, opinion mining and language translation for media regulatory authorities. Without automated systems, manual anatomizing is difficult. In this paper, we focus on the automatic Arabic and Urdu news ticker recognition system. It mainly consists of ticker segmentation and text recognition to generate textual data for various online services. Our work investigates character-wise explicit segmentation and syntactical models with Kufi and Nastaleeq fonts. Various network models anticipate learning of deep representations by homogenizing the classes regardless of inter-symbol correlations and linguistic taxonomy. The proposed learning model incorporates fairness by maximizing the balance among sensitive features of characters in a unified manner. Furthermore, we demonstrate the efficiency of the proposed model by carrying out experiments using customized news tickers datasets with accurate character-level and component-level labeling. Moreover, our method is evaluated on a challenging Urdu Printed Text Images (UPTI) dataset that only provides ligature based annotations. The proposed method attains 98.36%, outperforms the current state of the art method. Ablation investigations show that our technique enhances the performance of character classes with low symbol frequencies.

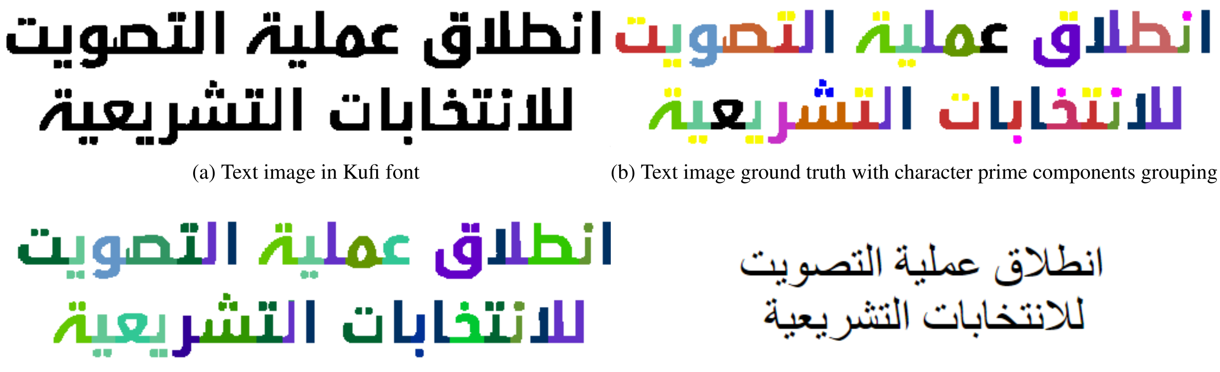
INDEX TERMS Arabic, Urdu, optical character recognition, SegNet, principal component analysis, deep convolutional network.

I. INTRODUCTION

Automated visual text reading is a vital research area on the basis of its applications. Big data analysis of news tickers is one of these applications from the daily broadcast streams. Recent developments are mostly focused on ticker recognition in the English language. However, recognition of tickers in other languages still requires considerable deliberation. In particular, recognition of news tickers in Arabic or Urdu languages is a challenging task due to a limited availability of labeled datasets and techniques specifically for cursive scripts. As of late, a few strategies [1] are presented to recognize the cursive text in a specially ad-hoc manner. These methods are utmost focusing on scan printed text in a controlled way. Nonetheless, the data from real world broadcast streams is noisy and comparatively troublesome to recognize by transfer learning approaches. Video distortions could be caused as a consequence of various factors like transmission, resolution, signal-to-noise ratio and other compression artifacts. Contemporary datasets come up short on these antiquities and accordingly a deterrent in the improvement of a learning model. This research confers a novel approach for ticker segmentation and recognition from video streams. By proposing the character-wise text segmentation techniques, this study forbids holistic approaches due to the huge number of ligature classes. To be font specific and standardization of news tickers, Saudi TV channel Al Arabiya (Kufi font) and few Pakistani (Nastaleeq font) broadcast streams are selected. We evaluate our model scheme on UPTI [2] dataset, which is the famous Urdu Nastaleeq dataset available, but it incorporates ligature-level labels.

Fully Connected Network (FCN) and SegNet segmentation architectures are explored followed by a proposed syntactical model. Deep learning models, especially

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



(a) Text image in Kufi font

(b) Text image ground truth with character prime components grouping

(c) Recognized text image result

(d) Recognized text result

FIGURE 1. Top row: Visualization of cursive text and its corresponding ground truth. Bottom row: recognized text image and recognized text result using the proposed technique. Best viewed in color.

Convolutional Neural Network (CNN), is specifically used for two-dimensional image data analytic problems. However, recently single dimensional CNN has also demonstrated tremendous performance, mainly for time series data analysis [3]. In the area of video/image processing, CNN has set up opportunities in numerous tasks including person re-identification, tracking, video analysis, text recognition and object detection etc. [4]. The latest methods extricate deep representation from images through different CNN models that have an incredible success in computer vision domains among others. In the current study, CNN is employed for extraction of pixel-wise features. These features are upsampled to form characters or character components. Syntactical model is used to conclude the recognized string into the sentence of words. Effectiveness of the models are analyzed by conducting experiments using a novel Arabic and Urdu news ticker datasets with character-level as well as component-level labeling. Fig. 1 illustrates visualization of cursive Arabic text and its corresponding ground truth. Evaluation on the UPTI dataset shows comparatively better performance from the previous results. The proposed method outperforms the current state of the art techniques.

The contributions of this study are elaborated in the following points.

1) Introduction of novel Arabic and Urdu news ticker datasets incorporating character-level as well as component-level labeling with Kufi and Nastaleeq fonts respectively.
2) Addresses the challenge of a huge number of ligature classes and proposes an explicit approach of character-wise segmentation.
3) Extends FCN and SegNet architectures for effective character-wise Urdu text recognition that provide significant improvements over traditional feature learning.
4) Proposes model by grouping of similar shaped classes for more effective segmentation.
5) Experimental evaluation of proposed technique by applying it on UPTI dataset and custom built dataset with cursive Arabic and Urdu script.
6) Momentous improvement in overall performance of symbolic classes that have homogeneous shapes with low frequencies.

II. RELATED WORK

Few strategies are proposed to deal with the challenge of recognizing Arabic and Urdu text. The techniques can be primarily categorized into two categories i.e. 1) analytical and 2) holistic approaches [5]. An analytical approach relies on segmentation of text into characters, explicitly or implicitly. Holistic approaches employ segmentation of ligatures, as units of recognition. Structural and statistical features are the main studies of attention in numerous researches. Structural features rely on structure of the patterns, under study characters or ligatures i-e. information of loops, curves, joining or ending points, distribution of horizontal and vertical lines etc. Statistical features consider statistics calculated from the arrangement of pixels in the pattern of images. These features normally comprise moments, profiling, projections, pixel densities etc.

Most of the focus has been on different strategies for cursive text recognition with scanned images. Hussain *et al.* [6] has done adequate work on Urdu OCR with Hidden Markov Model (HMM) and a manually designed rule based post processing methods. They mostly used the Nastaleeq font. They investigate both holistic [7] and analytical [8] techniques using the Center of Language Engineering (CLE) dataset. Hussain *et al.* [8] propose a hybrid approach of Bag-of-Feature framework and HMMs for sequence recognition. In the evolutionary contributions of Arabic and Urdu OCR tasks, advanced machine learning techniques have been introduced. Variants of Recurrent Neural Networks (RNNs) are extensively used in text recognition research for multiple languages. They compare performances of RNN based text recognizers. Rahal *et al.* [9] used a very similar concept by using improved Long Short Term Memory (LSTM) for Arabic text recognition in videos. References [10] and [11] proposed Multidimensional Long Short Term Memory (MDLSTM) with Connectionist Temporal Classification (CTC) output layer. Ji Gan *et al.* [12] introduces an effective architecture for Chinese text recognition, which concurrently takes advantage of CNN and LSTM networks. Mirza *et al.* [13] highlighted text image problems from videos with complex backgrounds and proposed a deep learning model with the composition of CNN and LSTM. Experiments are carried out on 12,000 text lines extracted from 4,000 video frames from Pakistani news channels. Mirza *et al.* [14] proposed a similar UrduNet model with the composition of CNN and LSTM. A comprehensive series of experiments are carried out on a self-generated dataset from more than 13,000 video frames.

In this study, CNN networks are explored as well, but in a different manner. The proposed scheme is applied on news tickers of popular Saudi TV channel Al Arabiya and few Pakistani broadcast streams. From the method evaluation perspective, an extensively used UPTI dataset is selected. It is a publicly available dataset developed by Sabbour *et al.* [2] for the research community. This dataset includes various degraded variants of text lines with ligature based annotations. It includes ten thousand text images having printed Urdu Nastaleeq font scripts. Further details of holistic and analytical approaches are categorized as follows.

A. HOLISTIC APPROACH

As mentioned prior, the holistic approach deals with whole ligatures. The models are trained to recognize ligatures directly. Akram *et al.* [15] investigated and modified the Tesseract OCR to support Urdu Nastaleeq style with a holistic approach. For this, they used a self-generated dataset extracted from 17,453 unique words. They target a large set of ligature problems and reduce them to 1475 main body types without diacritics. Results on font sizes 14 and 16 are declared as 97.87% and 97.71% respectively. Images used were cleaned and segmented. They also developed Tesseract based Urdu OCR with four different recognizers. By this, they support multi-font sizes from 14 to 44 and accuracy up to 86.15% per ligature on the dataset of 224 document images. Farhan *et al.* [16] indicates the challenge of computation complexity and introduces a computationally efficient holistic approach for Arabic text recognition. Their technique is based on similar shaped word clustering. They suggested Discrete Cosine Transform (DCT) for Arabic OCR, with a word recognition rate of 84.8% on printed text. Based on the performance of deep learning in computer vision tasks, Rehman *et al.* [17] introduce a holistic approach with CNN based classification model. They claim accuracy up to 84.2% on custom built Urdu dataset. Ahmad *et al.* [18] presents a dissimilar technique from the previous by using robust gated contextual information among ligatures. They developed a model incorporating raw pixel value as a feature. They used a Gated Bi-directional Long Short Term Memory (GB LSTM) learning model on UPTI dataset having 43 classes and aligned input, with a declared 96.71% recognition rate. Architecture is trained on undegraded and tested on unseen image data. Javed *et al.* [19] also proposed CNN model, evaluated 18,000 Urdu ligatures and 98 different classes, and realized a recognition rate of 95%. They rely on fine-tuning and use of pre-trained CNN to avoid local optima problems. Moreover, images for the network are dynamically resized rather than fixed resizing by maintaining aspect ratio without significant distortion and placed at the top left corner of fixed size image.

In this approach, each ligature is examined as a separate class for models. Lehal *et al.* [20] specifies more than 25,000 ligatures in Urdu. This indicates the large number of ligature classes with a substantial quantity of samples per class. These enormous numbers of classes cause strain in training models. Such an approach is inappropriate for real time applications such as news ticker recognition.

B. ANALYTICAL APPROACH

Analytical method relies on segmentation of text into characters. Implicit and explicit segmentation of analytical approaches are discussed as below.

1) IMPLICIT SEGMENTATION

Ahmed *et al.* [21] highlighted issues related to bidirectional writing of Urdu script, words are written from right to left whereas numbers from left to right. Associating dots with base characters is a more challenging task. They presented an implicit approach by using Bidirectional Long Short Term Memory (BLSTM) and CTC output layer with UPTI scripts, claiming 88.94% accuracy. Hasan *et al.* [22] also used BLSTM architecture with a CTC output layer to recognize UPTI dataset. They analyze that normalizing input images to some height is necessary for uniform information. For this, they use textual content baseline information and claim 94.85% accuracy. Naz *et al.* [23] also used MDLSTM. They extract features from more relevant data from the characters sequences for the recognition engine. They achieved higher accuracy rates with statistical features and MDLSTM. They presented feature extraction using right-to-left sliding windows. Their methodology on the UPTI database reports a recognition rate of 94.97%. Naz *et al.* [24] observed that MDLSTM technique using raw pixels has not been explored before. They investigate MDLSTM using raw pixels for Urdu Nastaleeq font recognition. Experiments show that MDLSTM attained a recognition accuracy of 98% using UPTI dataset, significantly outperforming the state of the art techniques. Naz *et al.* [25] proposed an approach of zoning features because of its efficiency, low complexity and high speed with significant information. They used zoning features with a combination of Two Dimensional Long Short Term Memory networks (2DLSTM) as learning classifiers. Their methodology on the UPTI database reports a recognition rate of 93.39%. Akram *et al.* [26] presented an implicit Urdu character recognition technique with Nastaleeq font, based on recognition of characters and joiners. They identify that the connected stroke of a ligature image has sequential pairs of characters and their joiners. The joiner maintains the connecting stroke shape of a character with the next. The detailed analysis is carried out to extract the artistic features of characters and their joiners. The system is tested on 1600 text lines of UPTI dataset with HMMs classifier, results 98.37% recognition rate.

Analytical approach has been used efficiently in numerous studies. References [23]–[25] uses a sliding window feature extraction technique, followed by the recognizer. This is also referred to as recognition based segmentation as both processes are carried-out in parallel. The approach has a deciding factor of the aggregated number of segments. Less segments leads to less computation but underperforms on widely written words. Increasing the segments leads to more computation with more junk segments, which needs to be handled by the model [1]. That may cause under and over segmentation.

2) EXPLICIT SEGMENTATION

Naz *et al.* [27] and Ahmed *et al.* [28] presented an explicit scheme of hybrid approach by integrating convolutional and RNN for effective features learning of Urdu Nastaleeq cursive scripts with a large number of classes. The CNN extracts low level translational invariant features followed by MDLSTM for contextual feature extraction. Naz *et al.* [27] addresses the challenge of a large number of ligature classes by proposing novel learning mechanisms to learn from a small set of classes. Evaluation on UPTI dataset attains a recognition rate of 98.12% with a number of 44 classes. Ahmed *et al.* [28] point out segment challenges due to variant shape of characters and space between words. They claimed that CNN is suitable for visual images patterns learning. In the view of cursive scripted language, such explicit segmentation and features extraction approach is appropriate where extracting features is challenging towards desirable accuracy. They used Arabic images from the English-Arabic Scene Text (EAST) dataset. The top performance reported by using filter size of 3 x 3 and 0.005 learning rate is 14.57% error rate.

Explicit segmentation approach divides text into characters. References [27], [28], [30] work on explicit approach for character segmentation. However, to isolate characters, it needs extensive and complicated knowledge of characters shapes, their starting and ending points within ligature or words.

Naz *et al.* [27] extracts features with CNN by the usage of MNIST dataset and does not exactly contemplate Urdu text features. In addition, as mentioned prior [14] proposes an implicit method by introducing the UrduNet model in composition with CNN and LSTM for new tickers from videos. It can be analyzed that the UrduNet model is troublesome in training with low accuracy even on its self training dataset.

This research focuses on challenging explicit approaches for Arabic and Urdu news tickers. Robust semantic segmentation model is proposed to trounce complications in character/component-wise segmentation for the focused language scripts. In the context of semantic segmentation, pixel level segmentation refers to segmentation of text pixels from background whereas character level segmentation refers to segmenting characters within a ligature

The rest of this paper is organized as follows. Section III details the proposed methodology of segmentation model architecture and syntactical model for textual recognition. Experiments and results along with comparisons with existing systems for text recognition are presented in Section IV, whereas section V concludes the study.

III. NEWS TICKER ARCHITECTURE

This section is about architectures used in current research. FCN and SegNet are deep convolutional neural network models that are explored for text segmentation. Syntax generation model is employed as a post segmentation process to detect spaces in order to form complete sentences of words from the sequence of recognized/segmented string of letters. Fig. 2 demonstrates proposed architectural functionality, which will be further discussed in the coming sections. Section *A* and *B* details the segmentation and syntactical models respectively for text recognition and are as follows.

A. CHARACTER SEGMENTATION MODEL

As described earlier, current study investigates an explicit approach of cursive text image segmentation techniques. Character-wise label image dataset is vital to instigate the idea. No such data is available for the research purposes of Arabic and Urdu text recognition. Datasets generation process of the novel Arabic and Urdu news tickers are described in the following subsection 1.

1) DATASET GENERATION

For datasets preparation, broadcast streams are collected by the World Call Digital TV box, model GK7601E-HDCA. Video data are captured from popular Saudi news channel Al Arabiya and Pakistani news channels ARY News, GNN, 24 News etc. for Arabic and Urdu news tickers collection respectively. All videos are of 1080p resolution. One third of the bottom portion of video frames are considered as ticker regions as shown in Fig. 3 from Pakistani (Urdu Nastaleeq) news streams. Based on assumption, this bottom portion is good enough as particularly news tickers lie on this region. To be more focused, it is also assumed that textual content is bright on a dark background. Selected region image I is clamped with threshold α, obtaining result T where α is 75 pixel value as shown by Equation. 1. Using sobel operator fx, vertical gradient G_x is calculated which at each point contains the vertical derivative approximations and clamped with α as given in Equation. 2 and Equation. 3. G_x image is shown in Fig. 4. Vertical representation of normalized moving average of M matrix locates the text area along with approximate font size. Substantial font size ticker lines are extracted. In this study text image data with font sizes from 25pt to 42pt are selected. The region and objects aside from ticker text are excluded by the help of distance and parallel lines. Extracted tickers are shown in Fig. 5 and Fig. 6. Similarly Arabic news streams are processed as shown in the Fig. 7 and Fig. 8. For Al Arabiya news channel, it is assumed that text is dark

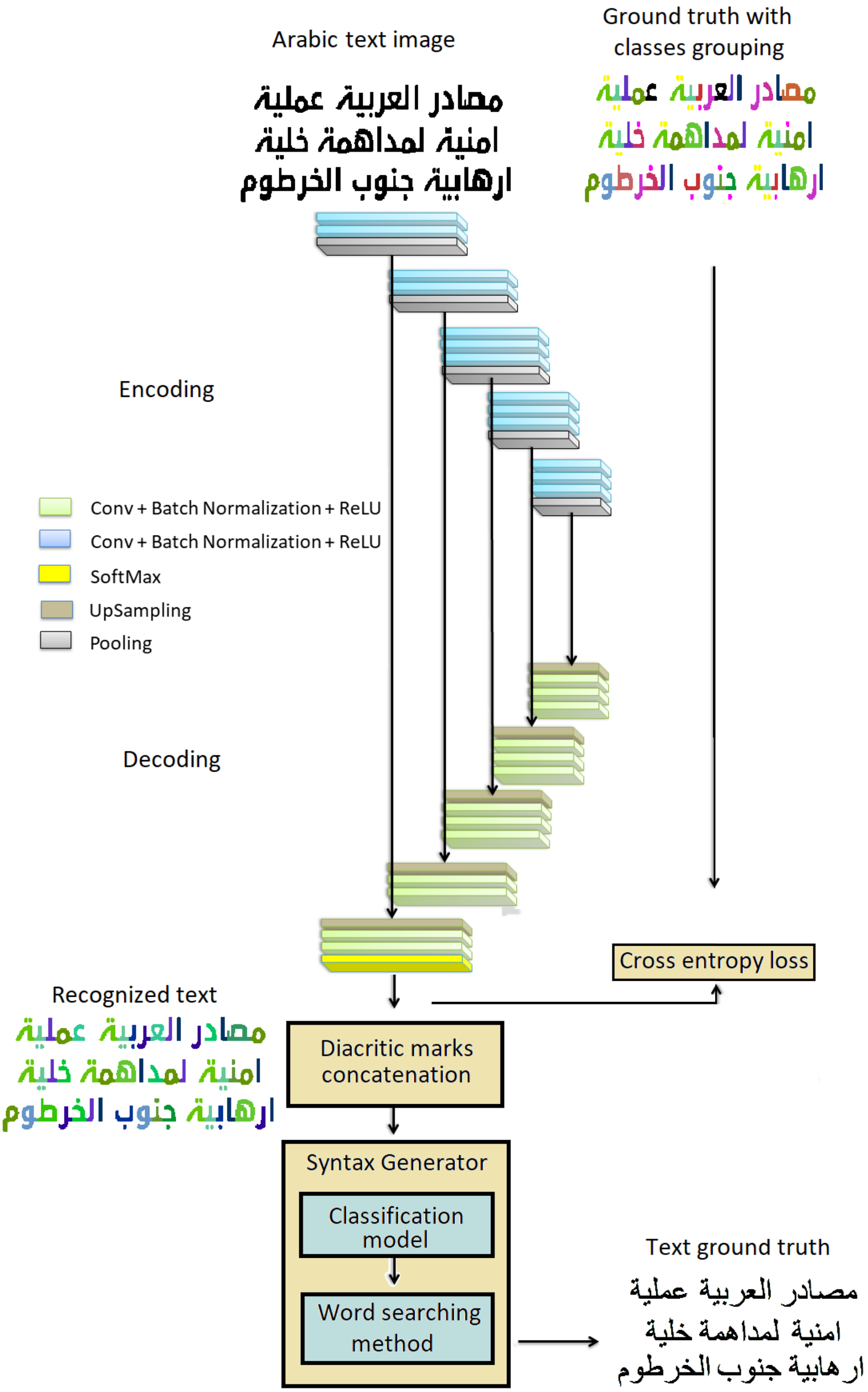


FIGURE 2. **Proposed model for news ticker recognition: SegNet model presenting downsample and upsampling architecture for characters semantic segmentation, syntax generation model for arabic (Kufi font) textual recognition. Best viewed in color.**

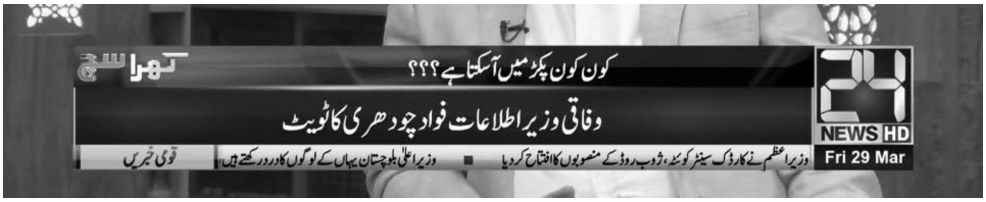

کون کون پکڑ میں آسکتا ہے؟؟؟
وفاقی وزیر اطلاعات فواد چودھری کا ٹویٹ
24
NEWS HD
Fri 29 Mar

FIGURE 3. Gray-scale image of urdu news ticker in nastaleeq font from the video stream.

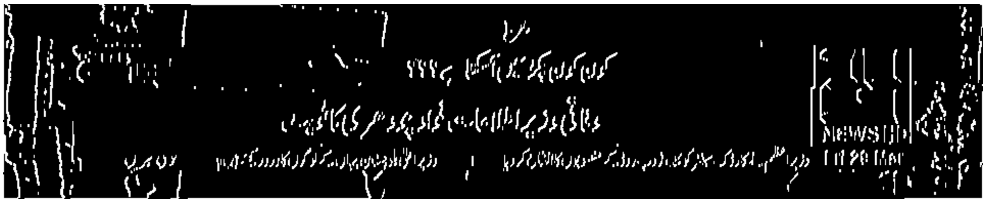


FIGURE 4. Vertical gradient G_x representation of ticker image.

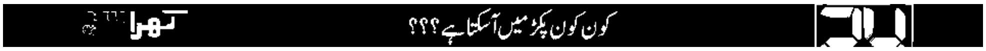


FIGURE 5. Extracted text line image of urdu news ticker in nastaleeq font (1).

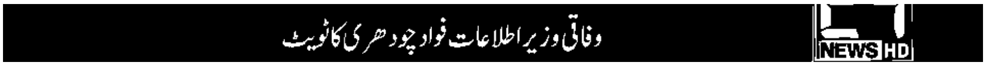


FIGURE 6. Extracted text line image of urdu news ticker in nastaleeq font (2).

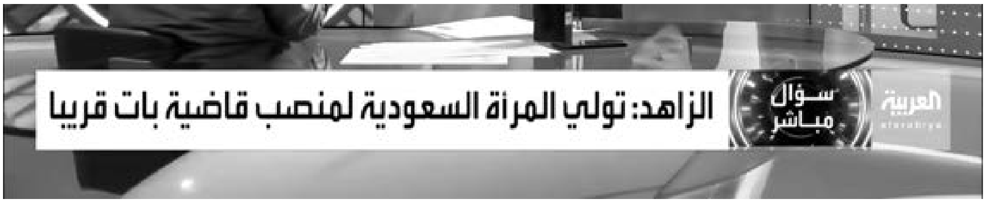


FIGURE 7. Gray-scale image of arabic news ticker in kufi font from the video stream.

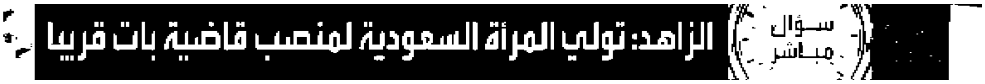


FIGURE 8. Extracted text line image of arabic news ticker in kufi font.

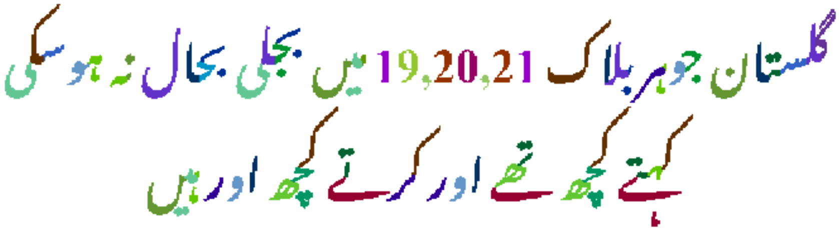

گلستان جوہر بلاک 19,20,21 میں بجلی بحال نہ ہو سکی
کہتے کچھ تھے اور کرتے کچھ اور ہیں

FIGURE 9. Visualization of character-wise labeled urdu news tickers lines in nastaleeq font. Best viewed in color.

on a bright background.

$$T = \begin{cases} 0 & I \leq \alpha \\ I & I > \alpha \end{cases} \tag{1}$$

$$G_x = f_x * T \tag{2}$$

$$M = \begin{cases} 0 & G_x \leq \alpha \\ G_x & G_x > \alpha \end{cases} \tag{3}$$

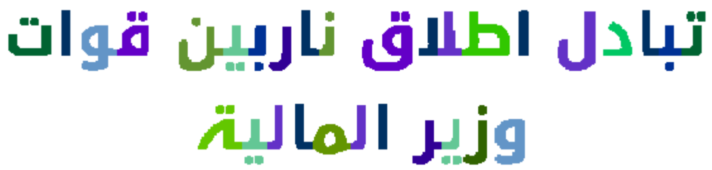
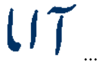
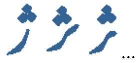
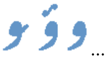
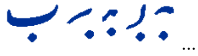
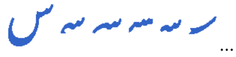
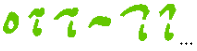
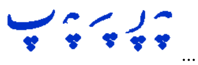
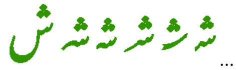
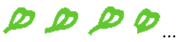
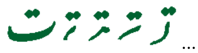
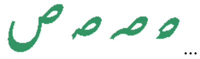
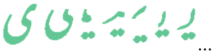
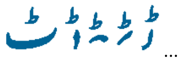
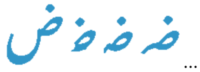
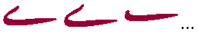
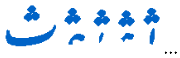
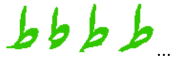
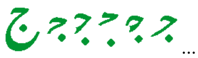
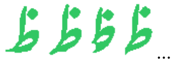
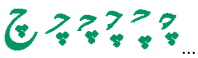
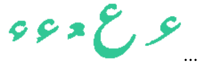
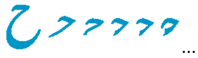
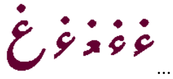
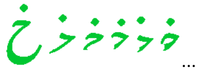
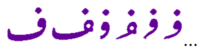
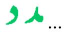
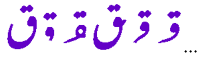
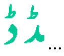
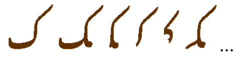
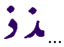
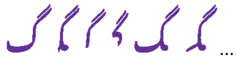
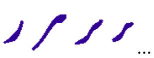
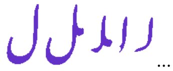
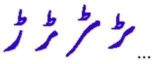
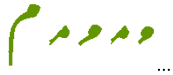
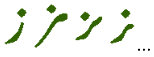
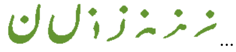


FIGURE 10. Visualization of character-wise labeled arabic news tickers lines in kufi font. Best viewed in color.

TABLE 1. Label and color assignment to isolated nastaleeq font character classes. Best viewed in color.

Class label	Shapes of characters with colormap	Class label	Shapes of characters with colormap	Class label	Shapes of Characters with colormap
1	...	17	...	33	...
2	...	18	...	34	...
3	...	19	...	35	...
4	...	20	...	36	...
5	...	21	...	37	...
6	...	22	...	38	0...
7	...	23	...	39	1...
8	...	24	...	40	2...
9	...	25	...	41	3...
10	...	26	...	42	4...
11	...	27	...	43	5...
12	...	28	...	44	6...
13	...	29	...	45	7...
14	...	30	...	46	8...
15	...	31	...	47	9...
16	...	32	...	48	؟:/()-،,!...

The Arabic dataset contains 1189 news tickers with Kufi font. It has 42 classes. The Urdu dataset contains 5945 news tickers with Nastaleeq font containing 48 classes. Few text lines from the datasets with their character-wise label representations are shown in Fig. 9 and Fig. 10. Label values assignment to character classes for Nastaleeq font are shown in Table 1. Numeric i-e. 0 to 9 are also the part of tickers and are included in the dataset with label values 38 to 47. Miscellaneous characters like brackets, slash, semicolon etc. are also included and assigned 48 label values. Characters with label values 3, 5, 8, 12, 15, 17 and 29 are not part of Arabic letters. However the letter HUMZA is included in the Arabic data as it is the part of tickers.

2) SEMANTIC SEGMENTATION PROCESS

This study is along the way, uses FCN and SegNet deep learning architectures for Urdu text recognition. The network learns to assign each pixel a class label determined by the object's or character's surroundings, spatial location and orientation it belongs to. When using these models for semantic segmentation, the recognized output is also an image of the same size as of label image rather than a vector. Encoder layer performs convolution with filters. The outputs are batch normalized after which an element-wise Rectified Linear Unit ReLU operation is performed as given in Equation. (4). Where x is batch normalization output value.

$$\phi(x) = \begin{cases} 0 & x \leq 0 \\ x & x > 0 \end{cases} \tag{4}$$

In FCN architectures, input image is downsampled by moving through the convolution and fully connected layers. Upsampled output is the predicted label map of the same size as the input image. Finally the softmax classifier predicts label map on probabilistic basis given in Equation (5). Where ϕ_k represents the values at output layer and n is the total number of outputs.

$$softmax(\phi_k) = \frac{\exp(\phi_k)}{\sum_j^n \exp(\phi_j)} \tag{5}$$

We use FCN-8s standard network due to its best performance in FCN variants to avoid spatial location information lost when moving deeper by fusing the process of three pooling layers. FCN-8s sums the 2x upsampled convolution7 (with a stride 2 transposed convolution) with pooling4 outputs, upsamples them with a stride 2 transposed convolution and sums them with pooling3 and applies a transposed convolution layer with stride 8 on the resulting feature maps to obtain the segmentation map [29].

Like FCN, SegNet architectures are also broadly in two parts, operating encoder and decoder. The input text image is first downsampled in encoding process like CNN architecture of ResNet or FCN. Decoding is a reversal process of encoding, processed with upsampling layers rather than downsampling (pooling) layers. At the deepest encoding output, SegNet discards fully connected layer. This reduces the number of parameters compared to other recently proposed architectures [31], [32]. The decoding output is fed to a softmax classifier to form a predicted label map of the same size as the input image on a probabilistic basis. In this investigation, eighteen layers encoder architecture is used. Like VGG16 network, the encoder layers has correspondence with the decoder layers and hence decoder also has eighteen number of layers as shown in Fig. 2.

All network parameters are randomly initialized. Pooling window dimensions and stride values are also the factors of the network. The output of the softmax classifier is a k channel image of probabilities where k is the number of classes. The predicted segmentation corresponds to the class with maximum probability at each pixel.

As compared to SegNet, training performance of FCN-8s is comparatively much lower. The study focuses on SegNet architecture for further investigation. While testing with Nastaleeq font, SegNet characters segmentation model leads to several classes correlated drawbacks. This is because of inter-class correlations between character classes that have homogeneous structure (prime components) with identical diacritic marks, causes learning complexity and underperforms peculiarly on characters with low frequencies. Further details will be discussed in results and experiments sections IV. Following subsection 3 describes the proposed technique for the above mentioned limitations.

3) PROPOSED SEMANTIC SEGMENTATION PROCESS

The proposed learning architecture incorporates the grouping of homogeneously shaped prime components classes with their diacritic marks separately identified in a unified manner. To this end, Arabic and Urdu ticker datasets are re-labeled by grouping character classes on the basis of characters families, having homogeneous structural shaped prime components with their diacritic marks are considered as separate classes. Table 2 tabulates grouped prime components of the character classes having homogeneous shapes (structural shapes) with group class labels. Diacritic marks of group components are shown in Table 3. Diacritic mark with label value 64 and 65 are not part of Arabic letters. The grouping processes of characters comforts low symbolic representations and reduces unfair distribution of data and diversity.

In the post segmentation process, segmented diacritic marks are concatenated with the corresponding prime component within the width of the prime component of characters. A heuristic process concatenates these diacritic marks. Cleaning or filtering is performed before the concatenation step. Minor semantic segmentation errors in recognised ticker images are removed using a statistical modulus procedure with a window size of 3 × 3 pixels. The recognized ticker image's labelled components are investigated. Components area A that fall below specific thresholds are removed (less than or equal to 3 × 3 pixel window). The characters and diacritic marks coexist, separate copies of the recognised ticker image are retained for prime components and diacritic marks, are relabeled in a sequential sequence. In parallel with area-based cleaning, the distances greater then 7 × 5 pixel window size between the multiple components with the same recognised label are split under component area $Amax$ criteria. The component's area exceeding certain limits, which are determined by font size $Amax = (round(fontSizeHeight \text{x} 0.1))^2$ and window size distance 7 × 5 pixels in case of diacritics is considered for splitting process. The left boundary of the component is used to determine the sequence labelled assignment. Within the width of the primary component, conditioned member of a specific family

TABLE 2. Label and color assignment to prime components of nastaleeq font characters incorporating homogeneous structural shapes. Best viewed in color.

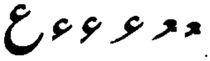
Grouping of class labels	New assign label	Shapes of prime component of characters with new colormap	Grouping of class labels	New assign label	Shapes of prime component of characters with new colormap
2 to 6	51	...	18 to 19	55	...
7 to 10	52	...	20 to 21	56	...
11 to 13	53	...	22 to 23	57	...
14 to 17	54	...	24 to 25	58	 ...

TABLE 3. Label and color assignment to diacritic marks of nastaleeq font characters incorporating homogeneous structural shapes. Best viewed in color.

Class label	Shapes of diacritical notations of characters with colormap	Class label	Shapes of diacritical notations of characters with colormap
61	...	64	...
62	...	65	...
63	...	-	-

with their predefined potential diacritic mark is grouped. The center points of the horizontal width of primary components and probable diacritic marks are determine in the case of multiple diacritic marks, assuming that the diacritic marks are not already paired with other characters. The minimum horizontal distance from the determined center point of diacritic marks and prime component are combined. The diacritic and prime component are assigned identified character label and the coordinates are stored. The character coordinates from the dataset are also saved for performance evaluation, with the same characters as considered identical. It is presumed that no duplicate diacritic marks with the same labels exist inside the width of the prime component based on observation. Heuristics can fail somewhere due to misclassification of pixels, which is especially for Nastaleeq, a challenging font.

This method leads to a significant reduction in the number of classes (42 to 35 for Arabic and 48 to 37 for Urdu) and model parameters (k of softmax classifier). We used encoding and decoding architecture with collectively thirty six layers for deep features learning as shown in Fig. 2. In the next section, we'll discuss the results and accuracy.

B. PROPOSED SYNTAX GENERATION MODEL

In the context of cursive text recognition, the segmented letters or recognized string of letters from the proposed segmentation model are compiled to the complete sentence of words by introducing spaces between words. The objective of this step is not only to determine word ending characters but also the consecutive characters within words. The potential inter-connected consecutive characters with the same class label like seen, sheen, noon (label values 18, 19, 32) occasionally lie in the words of the dataset. Arabic or Urdu characters generally have multiple shapes determined by its appearance and position within the word [33]. Mostly the ending letters in Arabic and Urdu writing has a particular shape which is nearly similar to the individual character. The current research develops an efficient method of classification. The Principal Component Analysis (PCA) transformation is employed for this objective. It is a orthonormal decomposition process by projection of relevant eigenvectors of the covariance matrix C given as Equation. 6 for 3-dimensional (x, y, and z) data. It is a NXN symmetric matrix, where N is the number of dimensions. var and cov are variance and covariance respectively.

$$C = \begin{bmatrix} var(x) & cov(x,y) & cov(x,z) \\ cov(y,x) & var(y) & cov(y,z) \\ cov(z,x) & cov(z,y) & var(z) \end{bmatrix} \tag{6}$$

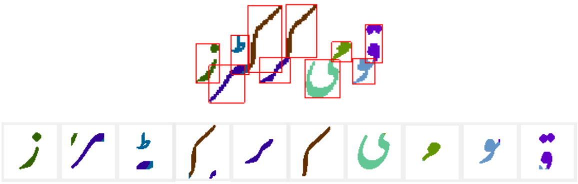


FIGURE 11. Segmented and concatenated nastaleeq font characters in [64 × 64] pixels dimensional buffer. Best viewed in color.

The standardization prior to the PCA transformation, normalization process is preformed to avoid sensitivity regarding the variances of the initial characters variables by using the following Equation. 7.

$$v = \frac{p - \mu}{\sigma} \tag{7}$$

where p is pixel value, μ is mean and σ is standard deviation. The eigenvectors i-e. the diagonal vector of variance as given in equation C directed to maximum variation of the letters prototype data. A precise calculation of the finest wide variety of factors may be received via way of means of the Fisher criterion [34]. In this investigation, the best and minimum number of appropriate eigenvectors in the transformed matrix are selected considering experimental results and will be further detailed in the experiment section IV. As described, diacritic marks are concatenated with their prime components at the post segmentation stage. Segmented characters image from the segmentation model are arranged right-to-left accordingly. These letters are sorted on left boundary basis and are placed in 64 × 64 pixels dimension buffers, as a post concatenation process of diacritic marks with prime components as shown in Fig. 11. The letters larger than 64 × 64 pixels dimension, height-wise, width-wise or both is scaled down to maximum 64 pixels height-wise, width-wise or both by maintaining the aspect ratios. The buffer is flattened to a vector for PCA. This model ignores classification of letter, which have a similar shape at the end of the words as well as ligatures. 45 and 56 classes are used for Arabic and Urdu classification models respectively. This includes possible characters, characters with space ahead, consecutive characters and consecutive characters with space ahead.

Presented approach is adequate for Kufi and Nastaleeq cursive fonts within the mentioned range of font size. Theses processed letter components are the candidates for the proposed classification model. This step is only for ending and consecutive characters identification. Classification process determine ending characters with certain accuracy and limitation for space insertion. In few cases, space between the word do not effect the meaning of word or sentence like کر رہا or کررہا means the same in presented Urdu cursive writing. Similarly, sometimes writing two words without space in between does not affect reading. The study suggest syntax correction process based on searching procedure to identify complete word within the recognized string. Lists of words are organized in ascending order. A track of these key lists is maintained in a separate array of pointers. This pointer array is known as a sparse array containing information of the lists of words. Each list contains a fixed length of words n i-e. from 2 to m. Where m is the maximum word length in data. These lists are arranged in order to perform efficient search of words [35]. Starting from the maximum length word, we tried to determine the complete word by a search process. Failing to find word in list, continuing searching process in list of words with length $n - 1$ up to the word list two characters.

IV. RESULTS AND EXPERIMENTS

As described, semantic segmentation learning networks are explored in this study. SegNet architecture is composed of an encoder and decoder layers. The encoder and decoder layers are symmetrical to each other. The upsampling operation of the decoder uses the max-pooling indices of the corresponding encoder. Unlike FCNs, no learnable parameters are used for upsampling.

Experiments are performed on Intel® Core i3-8100 with CPU@3.60GHzx4, 8GB DDR-5 RAM and SSD drive computer system. High-end GeForce RTX 2070/PCIe/SSE2 NVIDIA Graphics Processing Unit (GPU) is used to accelerate the training process. Architectures are implemented in C/C++. 5945 Urdu news tickers (3000 for training, 1000 for validation and 1945 for testing) are used in the Urdu segmentation model. 1189 Arabic news tickers (715 for training, 200 for validation and 274 for testing) are used in the Arabic segmentation model. These news tickers are single channel, placed in 205 × 1365 dimension with corresponding 205 × 1365 dimensions character-wise labeled images (ground truths). Sigmoid decay is used for learning rate.

A. SEGMENTATION MODEL EXPERIMENT

The encoder network consists of eighteen convolutional layers and each encoder layer has a corresponding decoder layer. ReLU as an activation function is used to further speed up the training. To produce class probabilities for each pixel independently, final decoder output is connected to a multiclass softmax classifier. Table 4 and Table 5 illustrates encoding and decoding layers in the network with parameters and matrix shapes. Number of bias parameters is also included in the total number of parameters.

In the first experiment, training is performed with Nastaleeq font by ignoring proposed technique of character families strategy. The training details are summarized in Table 6 (second column). This took 109 hours to train. The blue line in graph Fig. 12 shows SegNet learning curves for training loss, validation accuracy and validation loss.

In the second experiment, training is performed with Nastaleeq font by considering the proposed learning architecture that incorporates the grouping of homogeneous shaped prime component classes. Their diacritic marks are segmented by considering them as separate classes. Training details are summarized in Table 6 (in third column). It took 104 hours to train. Green line in graph Fig. 12 shows SegNet learning curves of training loss, validation accuracy and validation loss.

In view of graphical representation of training of models (blue and red curves of training and validation loss respectively), the overall convergence of proposed methods for semantic segmentation per epoch is head and shoulders above. An example of miss-classification by the first experiment is shown in Fig. 13. The network gets confused in semantically classification of low frequency character (label value 16). It proves that the proposed strategy of segmentation considering the families of Urdu letters is more appropriate.

In the third experiment, training is performed with Arabic Kufi font with proposed learning architecture. Training details are summarized in Table 6 (in fourth column). It took 34 hours to train. Fig. 14 shows SegNet learning curves of training loss, validation accuracy and validation loss.

TABLE 4. SegNet encoding layers with parameters and matrix shapes.

Encoding layers	Kernel size	Number of filters	Convolution output	Batch normalization	Mask pooling	BN + Bias parameters	Total number of parameters
Convolution	3	64	64x205x1365	Yes	No	128	768
Convolution	3	64	64x205x1365	Yes	No	128	37,056
Max Pooling	2	-	-	No	Yes	-	-
Convolution	3	64	64x103x683	Yes	No	128	37,056
Convolution	3	64	64x103x683	Yes	No	128	37,056
Max Pooling	2	-	-	No	Yes	-	-
Convolution	3	64	64x52x342	Yes	No	128	37,056
Convolution	3	64	64x52x342	Yes	No	128	37,056
Convolution	3	64	64x52x342	Yes	No	128	37,056
Max Pooling	2	-	-	No	Yes	-	-
Convolution	3	64	64x26x171	Yes	No	128	37,056
Convolution	3	64	64x26x171	Yes	No	128	37,056
Convolution	3	64	64x26x171	Yes	No	128	37,056
Max Pooling	2	-	-	No	Yes	-	-
Convolution	3	64	64x13x86	Yes	No	128	37,056
Convolution	3	64	64x13x86	Yes	No	128	37,056
Convolution	3	64	64x13x86	Yes	No	128	37,056
Max Pooling	2	-	-	No	Yes	-	-

TABLE 5. SegNet decoding layers with parameters and matrix shapes.

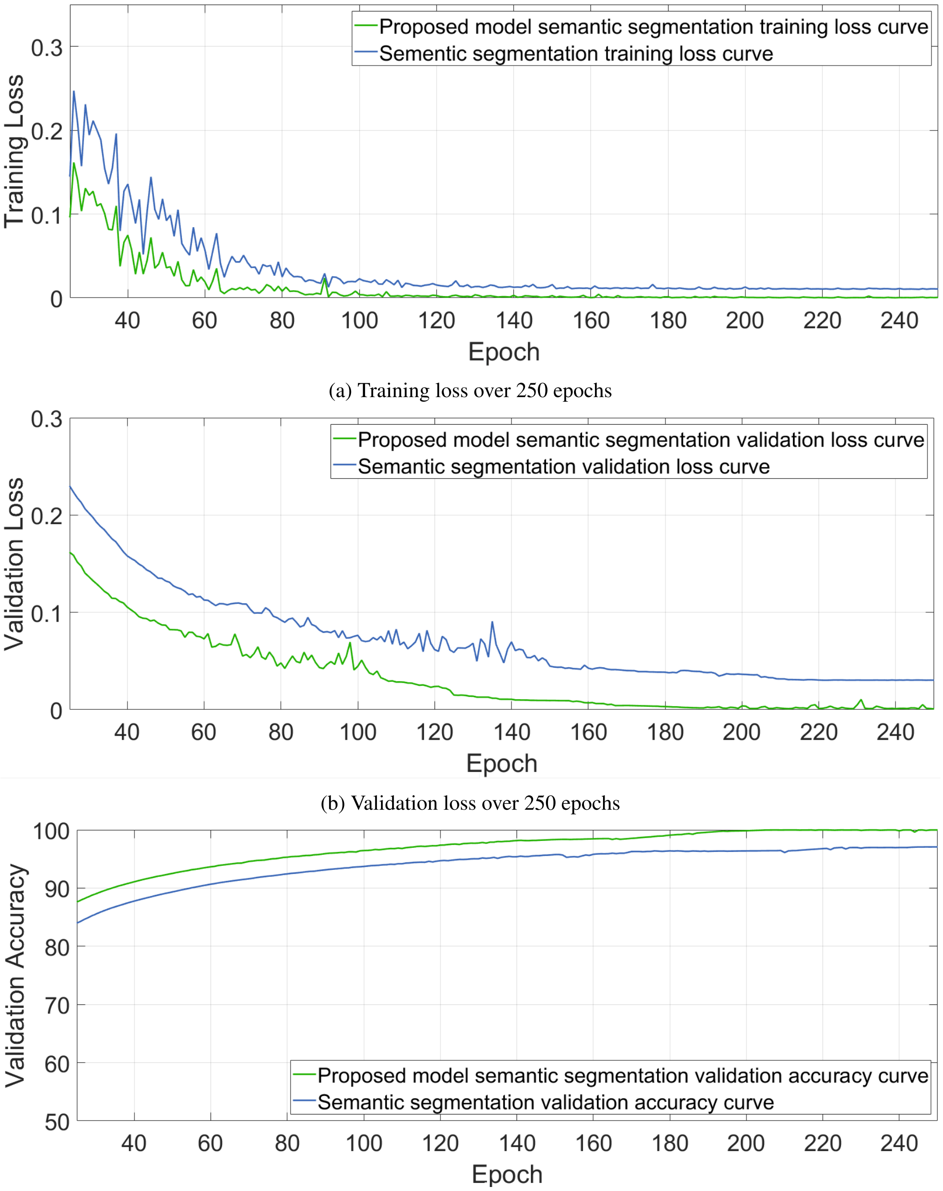
Decoding layers	Kernel size	Number of filters	Deconvolution output	Batch normalization	Mask upsampling	BN + Bias parameters	Total number of narameters
Upsample	2	-	-	No	Yes	-	-
Deconvolution	3	64	64x13x86	Yes	No	128	37,056
Deconvolution	3	64	64x13x86	Yes	No	128	37,056
Deconvolution	3	64	64x13x86	Yes	No	128	37,056
Upsample	2	-	-	No	Yes	-	-
Deconvolution	3	64	64x26x171	Yes	No	128	37,056
Deconvolution	3	64	64x26x171	Yes	No	128	37,056
Deconvolution	3	64	64x26x171	Yes	No	128	37,056
Upsample	2	-	-	No	Yes	-	-
Deconvolution	3	64	64x52x342	Yes	No	128	37,056
Deconvolution	3	64	64x52x342	Yes	No	128	37,056
Deconvolution	3	64	64x52x342	Yes	No	128	37,056
Upsample	2	-	-	No	Yes	-	-
Deconvolution	3	64	64x103x683	Yes	No	128	37,056
Deconvolution	3	64	64x103x683	Yes	No	128	37,056
Upsample	2	-	-	No	Yes	-	-
Deconvolution	3	64	64x205x1365	Yes	No	128	37,056
Deconvolution	3	49	49x205x1365	Yes	No	-	28,401

TABLE 6. SegNet training parameters.

Training parameters	Urdu ticker training without character grouping	Urdu ticker training with characters grouping	Arabic ticker training with characters grouping
Symbol classes	48	37	35
Training epochs	250	250	100
Batch size	1	1	1
Learning rate	0.001	0.001	0.001
Training time	109 hrs	104 hrs	34 hrs

B. CLASSIFICATION MODEL EXPERIMENT

Almost nine thousand Urdu letters candidates are processed by PCA transform with a variety of shapes collected from the Urdu segmentation output for Urdu classification. 50 most independent vectors indicated by high eigenvalues are selected. Improvement on test data by increasing



(a) Training loss over 250 epochs

(b) Validation loss over 250 epochs

(c) Validation accuracy over 250 epochs

FIGURE 12. Graphical representation of training curves comparison with nastaleeq font. Best viewed in color.

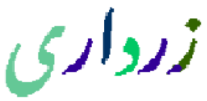


FIGURE 13. Miss-classification without considering character families strategy using nastaleeq font. Best viewed in color.

vectors from 50 to 75 does not generate much impact on the result with increasing cost to multiplications and distance calculations. Similarly decreasing the vector to 40 produces decreasing results in matching up to 6.81%. We have selected 50 appropriate number vectors. The minimum euclidean distance with each of the candidates is the recognized outcome. By considering the same Urdu classification strategy, around twenty five hundred characters of Arabic font are processed by PCA to construct an Arabic classification model.

C. RESULT ANALYSIS

The final validation accuracy of the Urdu SegNet training model without characters grouping method is 97.06% with validation loss 0.0203 and training loss is 0.0106. The validation accuracy of the proposed training model is 99.98% with validation loss 0.0013 and training loss 0.00058. In view of graphical representation of training models (blue and green curves of validation accuracy, training and validation loss) as

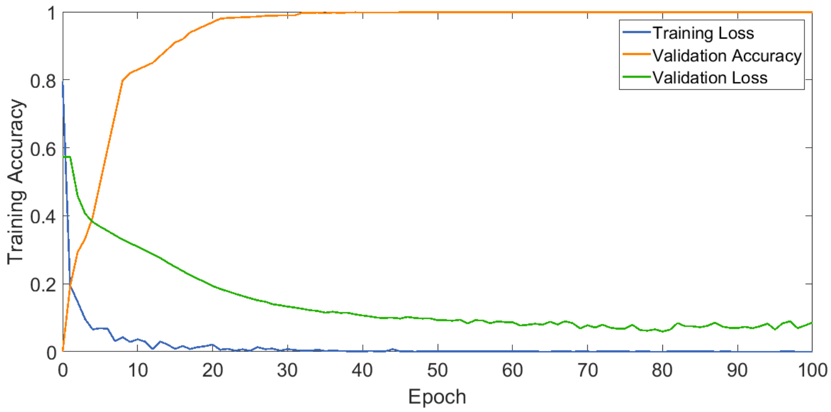


FIGURE 14. Graphical representation of training curves with kufi font. Best viewed in color.

TABLE 7. Detailed arabic characters segmentation performance of proposed method.

Class label	F-Score	Precision	Recall	Class label	F-Score	Precision	Recall
ا	0.92469	0.91381	0.93585	ک	0.97506	0.96506	0.98528
ب	0.97155	0.96765	0.97549	ل	0.97931	0.97699	0.98165
ت	0.93892	0.92795	0.95017	م	0.97745	0.97890	0.97602
ث	0.93223	0.92186	0.94285	ن	0.97609	0.97959	0.97263
ج	0.92981	0.89489	0.96757	و	0.97075	0.95547	0.98654
ح	0.96070	0.94445	0.97753	ہ	0.96359	0.94138	0.98689
خ	0.94993	0.93646	0.96380	ھ	0.94742	0.91413	0.98324
د	0.94598	0.92709	0.96567	ء	0.95941	0.95149	0.96748
ذ	0.93384	0.91754	0.95075	ی	0.93306	0.91257	0.95450
ر	0.95644	0.94276	0.97054	ے	0.95948	0.94840	0.97083
ز	0.93564	0.91679	0.95530	1	0.94072	0.90254	0.98229
س	0.95191	0.93655	0.96779	2	0.96273	0.92814	1
ش	0.92486	0.90162	0.94934	3	0.97563	0.95243	1
ص	0.94462	0.92119	0.96929	4	0.95794	0.91929	1
ض	0.93279	0.91098	0.95568	5	0.96022	0.92350	1
ط	0.94625	0.91959	0.97452	6	0.93727	0.88196	1
ظ	0.93674	0.93340	0.94011	7	0.96507	0.93251	1
ع	0.93300	0.89585	0.97337	8	0.93964	0.88616	1
غ	0.88355	0.83223	0.94162	9	0.99230	0.98473	1
ف	0.94205	0.91751	0.96794	0	0.98657	0.97351	1
ق	0.97272	0.96255	0.98311	.،!؟	0.86173	0.81830	0.91004
				Average	0.9493	0.9270	0.9723

shown in Fig. 12. In the post filtration process that removes misclassified pixels based on area within recognised components, segmented letter pixels are boundaried as prediction masks.The appropriate information retrieval metrics, F-Score and Accuracy, are illustrated below in Equation. 8, 9, 10 and 11 to evaluate predicted masks with the available target masks of provided input. The scores are calculated on overlapping masks with a minimum of 50% overlap. Only the pixels in the text area are taken into account when calculating pixel accuracy.

$$Accuracy = \frac{Total\ Correct\ Pixels}{Total\ Number\ of\ Pixels} \tag{8}$$

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

$$F - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{11}$$

Training improvement in the proposed segmentation model is due to minimizing inter-character correlation at pixel level by the grouping process. This restrains unfair distribution of data having low frequency representations and diversity. The imbalance in data distribution inherits biasness to individual datasets. On that basis, we extend the Arabic model by considering the character's family. The final validation accuracy

TABLE 8. Detailed urdu characters segmentation performance of proposed method.

Class label	F-Score	Precision	Recall	Class label	F-Score	Precision	Recall
ا	0.93062	0.92774	0.93353	غ	0.90633	0.85123	0.96906
ب	0.95808	0.94817	0.96821	ف	0.94704	0.91183	0.98508
پ	0.93929	0.92868	0.95015	ق	0.94846	0.93240	0.96510
ت	0.92561	0.92084	0.93043	ک	0.93601	0.92417	0.94817
ٹ	0.91763	0.90399	0.93169	گ	0.91359	0.89049	0.93794
ث	0.91948	0.91259	0.92649	ل	0.96271	0.95902	0.96644
ج	0.98263	0.97800	0.98731	م	0.92144	0.90944	0.93378
چ	0.98688	0.98431	0.98947	ن	0.93636	0.92490	0.94811
ح	0.97179	0.96910	0.97450	و	0.97499	0.96489	0.98532
خ	0.97859	0.97181	0.98547	ہ	0.91299	0.89337	0.93350
د	0.92078	0.89963	0.94296	ھ	0.98919	0.98900	0.98939
ڈ	0.91592	0.91145	0.92044	ی	0.93089	0.91369	0.94875
ذ	0.90659	0.90136	0.91189	ے	0.95992	0.94111	0.97950
ر	0.96739	0.94869	0.98686	0	0.92922	0.86780	1
ڑ	0.94863	0.93579	0.96183	1	0.90699	0.84389	0.98031
ز	0.94419	0.92549	0.96368	2	0.93753	0.88241	1
ژ	0.96369	0.95145	0.97625	3	0.93278	0.87403	1
س	0.97294	0.95853	0.98780	4	0.94233	0.89096	1
ش	0.95345	0.94622	0.96081	5	0.96443	0.93132	1
ص	0.97133	0.96351	0.97929	6	0.94147	0.88942	1
ض	0.96139	0.95513	0.96775	7	0.93592	0.87956	1
ط	0.95346	0.92401	0.98486	8	0.95054	0.90575	1
ظ	0.94148	0.91076	0.97435	9	0.94212	0.89059	1
ع	0.93704	0.90239	0.97446	؟؛،.	0.84897	0.80241	0.90127
				Average	0.9425	0.9197	0.9675

of the proposed Arabic SegNet training model with 35 numbers of classes is 99.93% with validation loss 0.085 and training loss is 0.00017.

TABLE 9. Performance comparison with UPTI nastaleeq font dataset.

Recent works with UPTI dataset	Accuracy in percentage
Hassan et al. [22]	86.40%
Ahmed et al. [21]	89.00%
Hassan et al. [22]	94.85%
Naz et al. [23]	94.97%
Naz et al. [10]	96.40%
Naseer et al. [36]	97.07%
Naz et al. [24]	98.00%
Naz et al. [27]	98.12%
This Study	98.36%

Urdu classification model results 94.51% on test data whereas Arabic classification model results 96.74% of test data. From the viewpoint of semantic segmentation, pixel-level segmentation refers to segmentation of text pixels from background whereas character-level segmentation refers to segmenting characters within a ligature. Testing on the Arabic dataset with the proposed Arabic model results weighted average f-score of 94.93% on character-level and 91.01% on pixel-level segmentation within text. Testing on the Urdu dataset with the proposed model results weighted average f-score of 94.25% on character-level and 89.10% on pixel-level segmentation within text. Table 7 and Table 8 shows character-wise detailed performance for Arabic and Urdu letters respectively. Model evaluation performed on UPTI data with 1600 selected text images, declares 98.36% at textual level. Results comparison on UPTI dataset are summarized in Table 9.

V. CONCLUSION

We propose highly accurate Arabic scripting ticker segmentation and text recognition models. Urdu ticker recognition model is developed in parallel to evaluation cursive language proposed technique. Arabic tickers are collected from Al Arabiya news channel whereas Urdu tickers are gathered from several popular Urdu Pakistani news channels. We investigate explicit method of character-wise text segmentation technique with Kufi and Nastaleeq fonts. The proposed learning architecture incorporates the grouping of homogeneous shaped prime component character classes. Our technique improves the performance of character classes that have low symbol frequencies. This is because the character's prime components and diacritic marks are considered separately for the learning model. Novel news ticker datasets, which provide accurate character-level labeling, are introduced. Proposed model achieves a textual recognition rate of 98.36% on UPTI dataset, outperformed over state-of-the-art methods. Furthermore, it is hard to segment text with a smaller font size and more de-shaped symbols. Datasets consisting of more balanced data distribution with different learning architectural parameters can be useful for more robust models.

REFERENCES

[1] N. H. Khan and A. Adnan, ''Urdu optical character recognition systems: Present contributions and future directions,'' *IEEE Access*, vol. 6, pp. 46019–46046, 2018, doi: 10.1109/ACCESS.2018.2865532.

[2] N. Sabbour and F. Shafait, ''A segmentation-free approach to Arabic and Urdu OCR,'' *Proc. SPIE*, vol. 8658, Feb. 2013, Art. no. 86580N, doi: 10.1117/12.2003731.

[3] S. U. Khan, I. U. Haq, Z. A. Khan, N. Khan, M. Y. Lee, and S. W. Baik, ''Atrous convolutions and residual GRU based architecture for matching power demand with supply,'' *Sensors*, vol. 21, no. 21, pp. 1–18, 2021, doi: 10.3390/s21217191.

[4] S. U. Khan, T. Hussain, A. Ullah, and S. W. Baik, ''Deep-ReID: Deep features and autoencoder assisted image patching strategy for person re-identification in smart cities surveillance,'' *Multimedia Tools Appl.*, pp. 1–22, Jan. 2021. [Online]. Available: https://link.springer.com/article/10.1007/s11042-020-10145-8#citeas, doi: 10.1007/s11042-020-10145-8.

[5] S. Naz, K. Hayat, M. I. Razzak, M. W. Anwar, S. A. Madani, and S. U. Khan, ''The optical character recognition of Urdu-like cursive scripts,'' *Pattern Recognit.*, vol. 47, no. 3, pp. 1229–1248, Mar. 2014, doi: 10.1016/j.patcog.2013.09.037.

[6] *Center for Language Engineering*. Accessed: 2022. [Online]. Available: http://www.cle.org.pk

[7] S. T. Javed and S. Hussain, ''Segmentation based Urdu nastalique OCR,'' in *Proc. Iberoamerican Congr. Pattern Recognit.* Berlin, Germany: Springer, 2013, pp. 41–49, doi: 10.1007/978-3-642-41827-3_6.

[8] S. Hussain, S. Ali, and Q. U. A. Akram, ''Nastalique segmentation-based approach for Urdu OCR,'' *Int. J. Document Anal. Recognit.*, vol. 18, pp. 357–374, Aug. 2015, doi: 10.1007/s10032-015-0250-2.

[9] N. Rahal, M. Tounsi, A. Hussain, and A. M. Alimi, ''Deep sparse auto-encoder features learning for Arabic text recognition,'' *IEEE Access*, vol. 9, pp. 18569–18584, 2021, doi: 10.1109/ACCESS.2021.3053618.

[10] 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.

[11] S. B. Ahmed, I. A. Hameed, S. Naz, M. I. Razzak, and R. Yusof, ''Evaluation of handwritten Urdu text by integration of MNIST dataset learning experience,'' *IEEE Access*, vol. 7, pp. 153566–153578, 2019, doi: 10.1109/ACCESS.2019.2946313.

[12] J. Gan, W. Wang, and K. Lu, ''In-air handwritten Chinese text recognition with temporal convolutional recurrent network,'' *Pattern Recognit.*, vol. 97, Jan. 2020, Art. no. 107025, doi: 10.1016/j.patcog.2019.107025.

[13] A. Mirza, I. Siddiqi, S. G. Mustufa, and M. Hussain, ''Impact of pre-processing on recognition of cursive video text,'' in *Pattern Recognition and Image Analysis*. Cham, Switzerland: Springer, 2019, pp. 565–576, doi: 10.1007/978-3-030-31332-6_49.

[14] A. Mirza, O. Zeshan, M. Atif, and I. Siddiqi, ''Detection and recognition of cursive text from video frames,'' *EURASIP J. Image Video Process.*, vol. 2020, no. 1, pp. 1–19, Dec. 2020, doi: 10.1186/s13640-020-00523-5.

[15] Q. Akram, S. Hussain, A. Niazi, U. Anjum, and F. Irfan, ''Adapting tesseract for complex scripts: An example for Urdu Nastalique,'' in *Proc. 11th IAPR Int. Workshop Document Anal. Syst. (DAS)*, 2014, pp. 191–195, doi: 10.1109/DAS.2014.45.

[16] F. Nashwan, M. Rashwan, H. Al-Barhamtoshy, S. Abdou, and A. Moussa, ''A holistic technique for an Arabic OCR system,'' *J. Imag.*, vol. 4, no. 1, p. 6, Dec. 2017, doi: 10.3390/jimaging4010006.

[17] A. U. Rehman and S. U. Hussain, ''Large scale font independent Urdu text recognition system,'' in *Pattern Recognition*. Amsterdam, The Netherlands: Elsevier, 2020.

[18] 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, Mar. 2018, doi: 10.1007/s10586-017-0990-5.

[19] 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. (FIT)*, Dec. 2017, pp. 93–97, doi: 10.1109/FIT.2017.00024.

[20] G. S. Lehal and A. Rana, ''Recognition of nastalique Urdu ligatures,'' in *Proc. 4th Int. Workshop Multilingual OCR (MOCR)*, 2013, pp. 1–5.

[21] S. B. Ahmed, S. Naz, M. I. Razzak, S. F. Rashid, M. Z. Afzal, and T. M. Breuel, ''Evaluation of cursive and non-cursive scripts using recurrent neural networks,'' *Neural Comput. Appl.*, vol. 27, no. 3, pp. 603–613, Apr. 2016, doi: 10.1007/s00521-015-1881-4.

[22] 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.*, 2013, pp. 1061–1065, doi: 10.1109/ICDAR.2013.212.

[23] S. Naz, A. I. Umar, R. Ahmad, S. B. Ahmed, S. H. Shirazi, and M. I. Razzak, ''Urdu Nasta'liq text recognition system based on multi-dimensional recurrent neural network and statistical features,'' *Neural Comput. Appl.*, vol. 28, no. 2, pp. 219–231, Feb. 2017, doi: 10.1007/s00521-015-2051-4.

[24] S. Naz, A. I. Umar, R. Ahmed, M. I. Razzak, S. F. Rashid, and F. Shafait, ''Urdu Nasta'liq text recognition using implicit segmentation based on multi-dimensional long short term memory neural networks,'' *SpringerPlus*, vol. 5, no. 1, pp. 1–16, Dec. 2016, doi: 10.1186/s40064-016-3442-4.

[25] S. Naz, S. B. Ahmed, R. Ahmad, and M. I. Razzak, ''Zoning features and 2DLSTM for Urdu text-line recognition,'' in *Proc. Int. Conf. Knowl. Based Intell. Inf. Eng. Syst.*, 2017, pp. 16–22, doi: 10.1016/j.procs.2016.08.084.

[26] Q. U. A. Akram and S. Hussain, ''Improving Urdu recognition using character-based artistic features of nastalique calligraphy,'' *IEEE Access*, vol. 7, pp. 8495–8507, 2019, doi: 10.1109/ACCESS.2018.2887103.

[27] 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, doi: 10.1016/j.neucom.2017.02.081.

[28] S. B. Ahmed, S. Naz, M. I. Razzak, and R. Yousaf, ''Deep learning based isolated Arabic scene character recognition,'' in *Proc. 1st Int. Workshop Arabic Script Anal. Recognit. (ASAR)*, Apr. 2017, pp. 46–51, doi: 10.1109/asar.2017.8067758.

[29] J. Long, E. Shelhamer, and T. Darrell, ''Fully convolutional networks for semantic segmentation,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[30] S. A. Hussain, S. Zaman, and M. Ayub, ''A self organizing map based Urdu nasakh character recognition,'' in *Proc. Int. Conf. Emerg. Technol. (ICET)*, Oct. 2009, pp. 267–273.

[31] K. Simonyan and A. Zisserman, ''Very deep convolutional networks for large-scale image recognition,'' in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2014, pp. 1409–1556.

[32] V. Badrinarayanan, A. Kendall, and R. Cipolla, ''SegNet: A deep convolutional encoder-decoder architecture for image segmentation,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017, doi: 10.1109/TPAMI.2016.2644615.

[33] S. Naz, A. I. Umar, S. H. Shirazi, S. B. Ahmed, M. I. Razzak, and I. Siddiqi, ''Segmentation techniques for recognition of arabic-like scripts: A comprehensive survey,'' *Educ. Inf. Technol.*, vol. 21, no. 5, pp. 1225–1241, Sep. 2016, doi: 10.1007/s10639-015-9377-5.

[34] A. N. Khan, J. R. De la Haye, and A. H. Hegt, ''A license plate recognition system,'' *Proc. SPIE*, vol. 3460, pp. 14–24, Oct. 1998.

[35] S. Ali, I. Ul, M. Asif, M. Bin, and M. Tayyab, ''An efficient scheme for real-time information storage and retrieval systems: A hybrid approach,'' *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 8, pp. 427–431, 2017.

[36] A. Naseer and K. Zafar, ''Comparative analysis of raw images and meta feature based Urdu OCR using CNN and LSTM,'' *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 1, pp. 419–424, 2018, doi: 10.14569/IJACSA.2018.090157.

MOEEN TAYYAB received the B.S. degree in computer and information sciences from the Pakistan Institute of Engineering and Applied Sciences, Pakistan, in 2007, and the M.S. degree in electronic engineering from International Islamic University, Pakistan, in 2009, where he is currently pursuing the Ph.D. degree with the Department of Applied Sciences. Since 2009, he has been working as a Software Engineer at USA-based software firm. His research interests include image/video processing, machine learning, and embedded systems.

AYYAZ HUSSAIN received the Ph.D. degree from the FAST, National University of Computer and Emerging Sciences (NUCES), Pakistan, in 2009. From 2013 to 2014, he was a Postdoctoral Scientist with the Gwangju Institute of Science & Technology, South Korea. He worked as an Assistant Professor, an Associate Professor, and the Chairman of the Department of Computer Science & Software Engineering, International Islamic University, Pakistan. He is currently working as a Professor at the Department of Computer Sciences, Quaid-i-Azam University, Pakistan. His research interests include pattern recognition, machine intelligence, data mining, and image processing.

MOHAMMED ALI ALSHARA received the B.Sc. degree from Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia, in 2002, and the M.Sc. and Ph.D. degrees from the University of Texas, in 2008 and 2016, respectively. He is currently working as an Assistant Professor and the Vice Dean of the College of Computer and Information Sciences (CCIS), IMSIU. His research interests include data and content analysis, knowledge management, and information retrieval.

SHAKIR KHAN received the B.Sc., M.Sc., and Ph.D. degrees in computer science, in 1999, 2005, and 2011, respectively. He is currently working as an Associate Professor at the College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia. He is teaching bachelor's and master's degree courses with the College of Computer Science, Imam University. He has around more than 15 years of teaching, research, and IT experience in India and Saudi Arabia. His research interests include big data, data science, data mining, machine learning, the Internet of Things (IoT), eLearning, artificial intelligence, emerging technology, open-source software, library automation, and mobile/web application. He published many research papers in international journals and conferences in his research domain. He is a member of the International Association of Online Engineering (IAOE). He is s reviewer of many international journals.

REEMIAH MUNEER ALOTAIBI received the B.Sc. degree from Imam University, and the M.Sc. and Ph.D. degrees in information management from Leeds Beckett University, U.K. Moreover, she has some courses, such as Professional Development Diploma in project management with the Midlands Academy of Business and Technology, U.K. She is currently an Assistant Professor with the College of Computer Sciences and Information, Imam Muhammad Ibn Saud Islamic University, Saudi Arabia. She has also authored articles published in scientific journals, book chapters, and conference materials, and has acted as a Reviewer of the *International Journal of Civic Engagement and Social Change* (IJCESC) (IGI Global).

ABDUL RAUF BAIG received the B.E. degree in electrical engineering from the NED University of Engineering and Technology, Karachi, Pakistan, in 1987, the Diplôme de Spécialisation degree in computer science from Supélec, Rennes, France, in 1996, and the Ph.D. degree in computer science from the University of Rennes 1, Rennes, in 2000. He was with the National University of Computer and Emerging Sciences, Islamabad, Pakistan, as a Faculty Member, an Assistant Professor, an Associate Professor, and a Professor, from 2001 to 2010. Since 2010, he has been a Professor with Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia. He has more than 100 publications in journals and international conferences. His research interests include machine learning, data mining, and evolutionary algorithms. He is a member of the IEEE Computational Intelligence Society.