











## Research Article

# Urdu Handwritten Characters Data Visualization and Recognition Using Distributed Stochastic Neighborhood Embedding and Deep Network

Mujtaba Husnain <sup>1</sup>, Malik Muhammad Saad Missen <sup>1</sup>, Shahzad Mumtaz <sup>1</sup>,  
Dost Muhammad Khan <sup>1</sup>, Mick  el Coustaty <sup>2</sup>, Muhammad Muzzamil Luqman <sup>2</sup>,  
Jean-Marc Ogier <sup>2</sup>, Hizbullah Khattak <sup>3</sup>, Sikandar Ali <sup>4</sup>, and Ali Samad <sup>1</sup>

<sup>1</sup>Department of Information Technology, Faculty of Computing, The Islamia University of Bahawalpur, Bahawalpur 63100, Pakistan

<sup>2</sup>L3i Lab, Universit   de La Rochelle Av. Michel Cr  peau, 17000 La Rochelle, France

<sup>3</sup>Department of Information Technology, Hazara University Mansehra, 21120 Khyber Pakhtunkhwa, Pakistan

<sup>4</sup>Department of Information Technology, The University of Haripur, Khyber Pakhtunkhwa, Pakistan

Correspondence should be addressed to Sikandar Ali; sikandar@cup.edu.cn

Received 25 May 2021; Revised 18 July 2021; Accepted 14 August 2021; Published 3 September 2021

Academic Editor: Shahzad Sarfraz

Copyright    2021 Mujtaba Husnain et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we make use of the 2-dimensional data obtained through *t*-Stochastic Neighborhood Embedding (*t*-SNE) when applied on high-dimensional data of Urdu handwritten characters and numerals. The instances of the dataset used for experimental work are classified in multiple classes depending on the shape similarity. We performed three tasks in a disciplined order; namely, (i) we generated a state-of-the-art dataset of both the Urdu handwritten characters and numerals by inviting a number of native Urdu participants from different social and academic groups, since there is no publicly available dataset of such type till date, then (ii) applied classical approaches of dimensionality reduction and data visualization like Principal Component Analysis (PCA), Autoencoders (AE) in comparison with *t*-Stochastic Neighborhood Embedding (*t*-SNE), and (iii) used the reduced dimensions obtained through PCA, AE, and *t*-SNE for recognition of Urdu handwritten characters and numerals using a deep network like Convolution Neural Network (CNN). The accuracy achieved in recognition of Urdu characters and numerals among the approaches for the same task is found to be much better. The novelty lies in the fact that the resulting reduced dimensions are used for the first time for the recognition of Urdu handwritten text at the character level instead of using the whole multidimensional data. This results in consuming less computation time with the same accuracy when compared with processing time consumed by recognition approaches applied to other datasets for the same task using the whole data.

## 1. Introduction

Data visualization deals with presenting the data in some visual context to make it trivial for the human to understand the nature of the data [1]. Furthermore, this activity helps in finding the patterns and hidden information, if they exist, in the data for further processing like data clustering and data classification. Nowadays, it is of common observation that the information related to the data science is of high dimensions, and therefore, its visualization in low-

dimensional space becomes impractical (Buja et al. [2]; Saeed et al. [1]). In almost all of the data science datasets, the researchers have had to deal with this acute and critical issue. While analyzing the high-dimensional data, almost every other researcher is interested in finding the optimal number of dimensions (or features) in order to apply any appropriate classifier for giving better performance (Nguyen and Holmes [3]; Song et al. [4]; ur Rehman et al. [5]). It is pertinent to mention that the terms “high-dimensional data visualization” and “high-dimensional visualization” are used

interchangeably in the literature; however, there is a succinct difference between these. In the first, the term *high* refers to data itself, whereas, in the second, it refers to the visualization. The interesting fact lies in visualizing the high-dimensional data on 2D or 3D plane that we have to apply some appropriate dimensionality reduction approaches on the whole data, since it is next to impossible in order to visualize the high-dimensional data on low-dimensional space (Engel et al. [6]; Song et al. [4]; ur Rehman et al. [5]).

The term “dimensionality” refers to the number of variables, characteristics, or features in which most of the datasets exist in the field of data science nowadays. Generally, these dimensions are represented as columns, and the main purpose is to reduce this number of columns. In the majority of cases, these column values are correlated and are also having redundant information that causes noise in data. This redundant information may impact negative and adverse effects in training any machine learning model thus producing error-prone results. That is the reason, the dimensionality reduction approaches have become of vital importance. Furthermore, it also helps in finding the patterns, if they exist, in the data set prior to applying any clustering or classification approach by reducing the model’s complexity thus avoiding the overfitting.

One of the key objectives of the dimensionality reduction technique is to reduce the high-dimensional data points  $D = d_1, d_2, d_3, \dots, d_n$  to a rather low dimension space, ideally, in two- (or three-) dimensional spaces  $S = s_1, s_2, s_3, \dots, s_n$  in order to get better visualization of the data, where  $S$  represents the equivalent low-dimension transformation and map of  $D$  and  $s_i$  is the corresponding data point of  $d_i$  that can be viewed on some appropriate scatter plot. The main purpose in this transformation is to preserve the characteristic features of the high-dimensional data as much as possible while transforming to the low-dimensional space. It is pertinent to mention that different dimensionality reduction approaches have a number of various capabilities of preserving the different types of properties of high-dimensional data (Engel et al. [6]; Saeed et al. [1]; Sorzano et al. [7]). Some are specific to preserve the linear dependencies, and some are experts in taking care of nonlinear features only. In order to solve these challenging issues, we engaged one of the most popular and widely used algorithms, the  $t$ -Distributed Stochastic Neighbor Embedding ( $t$ -SNE) (Maaten and Hinton [9]). The results showed that the  $t$ -SNE produced quite *faithful* clusters with clear and accurate separations while converting to low-dimensional data, thus retaining the characteristic features of the high-dimensional data. Furthermore, the reduced dimensions are then plugged into the Convolution Neural Network (CNN) to recognize and classify the Urdu handwritten numerals and characters in a separate set of experiments. The quality and efficiency of the results using the reduced dimensions obtained through  $t$ -SNE are remarkably better than those of the approaches used previously for the said purpose.

The paper is outlined as follows: Section 2 gives an overview of dimensionality reduction approaches. In Section 3, we discuss the motivation behind our work. In Section 4, the processing steps used in generating state-of-art datasets

are discussed in detail. The experimental results obtained by using reduced dimensions obtained through  $t$ -SNE and other approaches are presented in Section 5. Section 6 provides recognition results of Urdu handwritten characters and numerals using a deep CNN based model. In Section 7, we conclude the paper with some proposed future works.

## 2. Review of the Approaches Used in Dimensionality Reduction

A number of dimensionality reduction approaches for nonlinear data have been proposed in the last decade (Camastra [10]; Cunningham and Ghahramani [11]; Sorzano et al. [7]). The nonlinear techniques are more capable in comparing the standard and conventional linear techniques of dimensionality reduction; in dealing with complex nonlinear data, since most of the data sets associated with the data science and big data are likely to be strongly nonlinear in nature (Tsai [12]; Van Der Maaten et al. [8]). The related literature concludes that, among the existing number of dimensionality reduction techniques, the Principal Component Analysis (PCA) (Roweis and Saul [13]) is regarded as the most popular (unsupervised) linear technique (Maimon and Rokach [14]; Saul et al. [15]; Tsai [12]). Therefore, in this paper, we considered the PCA Roweis and Saul [13] as a benchmark. There also exist other techniques like Multi-Dimensional Scaling (MDS) Torgerson [16] that favor the data in linear form. This approach primarily focuses on the structural properties of the data points that vary in similarity. It is pertinent to mention that researchers must consider the nonlinear features of the high-dimensional data and also the very similar data points to produce clear separated clusters. This activity will also help in resolving issues associated with intracluster separations.

Some noteworthy survey articles (Camastra [10]; Cunningham and Ghahramani [11]; Sorzano et al. [7]) provide detailed information about dimensionality reduction approaches including Local Linear Embedding (LLE) (Roweis and Saul [13]); Laplacian Eigenmaps (Belkin and Niyogi [17]); Maximum Variance Unfolding (MVU) (Weinberger et al. [18]); Stochastic Neighbor Embedding (SNE) (Hinton and Roweis [19]); and Curvilinear Components Analysis (CCA) (Demartines and Herault [20]) that specifically deal with nonlinear data by preserving the structural features of the whole data. It is also concluded from the work of Engel et al. [6] and Maaten and Hinton [9] that the approaches mentioned above did not produce the effective visualization results, since these approaches failed to manage the nonlinear characteristics of the whole data in the projected low-dimension map. Therefore, these approaches are not recommended to get the correct and perfect visualization of realistic data set having high-dimensional data points. The authors Engel et al. [6], Maaten and Hinton [9], and Song, Gretton, Borgwardt, and Smola [21] also observed that the MVU failed to visualize the English handwritten digits and provided highly overlapped clusters. To address these issues, we used the  $t$ -Distributed Stochastic Neighbor Embedding ( $t$ -SNE) (Maaten and Hinton [9]) approach for producing an efficient and effective visualization of the multidimensional

data in the form of clusters with clear and accurate separations by embedding both the pixel- and structural-based information in a principled way. It is pertinent to mention that  $t$ -SNE is a modified and extended form of SNE (Hinton and Roweis [19]). The categorization of the dimensionality reduction techniques is shown in Figure 1.

The literature related to the concept of t-SNE concludes that it is also one of the dimensionality reduction and data visualization techniques that deals with nonlinear data in an efficient way, since the math behind t-SNE is quite complex, but the idea is simple. The novelty in the efficacy of t-SNE is that it embeds the points from a higher dimension to a lower dimension trying to preserve the neighborhood of that specific point more efficiently as compared to other conventional and classical approaches like PCA, auto-encoders, High Correlation Filter, etc. Most classical dimensionality reduction approaches inherently work on preserving the global structure of the data, while t-SNE focuses tabs on both the local and global attributes of the data. This novelty of t-SNE assists in generating the clusters with high degree of compactness and intercluster separations.

### 3. Our Motivation

In this work, we performed experiments in two phases, namely, (i) the visualization of Urdu handwritten characters and numerals containing pixel-based features embedded with structural-based features using dimensionality reduction approaches and (ii) the recognition of these characters and numerals using deep network model of CNN using both pixel-based and structural based features and then using reduced dimensions of the same instances obtained through  $t$ -SNE and other approaches. In order to perform the abovementioned tasks, we prepared a novel data set of Urdu handwritten characters and numerals.

One of the issues associated with Urdu script is the shape similarity among its characters and numerals, as shown in Figure 2.

These issues may result in generating overlapping clusters during visualization in low-dimensional space, which may directly affect the accuracy rate in the recognition process. Therefore, we have to apply some suitable approach that results in precise and correct clusters with perfect separations. Moreover, intracluster separations of the data instances should be clear to depict the separations among the individual instances of the Urdu characters and numerals. The following factors and issues are the cause of our motivation to perform some experiments to resolve these issues.

To the best of our knowledge, there is a lack of noteworthy work done to date in order to transform *faithfully* the high-dimensional data of the Urdu handwritten characters to low-dimensional space.

- (i) There is no dataset associated with the Urdu handwritten characters and numerals that is available publicly to perform text recognition tasks at the character level.
- (ii) Also, there is no process of recognition adopted at the character level in Urdu handwritten text using

reduced dimensions obtained through dimensionality reduction approaches.

- (iii) In the subsequent section, we outline the procedure to generate a state-of-the-art dataset consisting of the images of the Urdu handwritten characters and numerals. Furthermore, the experimental results produced by the dimensionality reduction and recognition approaches are also given in subsequent sections.

### 4. Dataset Preparation

As mentioned in the earlier section, there is a lack of an appropriate and concise data corpus containing Urdu handwritten characters and numerals to perform text recognition tasks at the character level. There are some publicly available datasets of Urdu handwritten text like Urdu Nastaliq Handwritten Dataset (UNHD) (Ahmed et al. [22]; Das et al. [23]; Husnain et al. [24]; Sagheer et al. [25]), but unfortunately, the data sets have only the Urdu handwritten numerals. Furthermore, these instances are not enough to apply state-of-the-art machine learning algorithms to get better results. In order to bridge the gap and provide the state-of-the-art dataset of this kind, we invited about 1000 native Urdu speaking persons from different academic, administrative, and social groups of different age groups and gender. Moreover, the handicapped and physically disabled people are also involved to make the dataset more concise and comprehensive. All the authors were directed to write in a separate column of the printed sheets in his (or her) handwriting. Each of the sheets has printed images of 40 basic Urdu alphabets along with the 10 Urdu numerals in Nastaliq font.

Figure 3 depicts the sample page of our dataset. Furthermore, we have also recorded the demographic information of each author to generate the ground truth values of the whole data set. This information includes the basic information about each author, namely, age, race, gender, level of education, type of job, physical disability (if any), preference of left (or right) hand while writing, etc. This activity helped us in making the dataset more concise and comprehensive. After collection of an appropriate amount of data instances, the handwritten pages of both the Urdu characters and numerals were carefully scanned on a flatbed scanner at standard 300 dpi resolution. Furthermore, the scanned pages are then segmented manually into an image size of  $28 \times 28$  to capture Urdu handwritten character and numeral individually. As mentioned earlier, the whole dataset consists of  $1000 \times 10 = 10,000$  Urdu numerals and  $1000 \times 40 = 40,000$ .

Urdu characters images: for experimental work, we randomly selected the 6000 (600 each for the ten numerals) images of Urdu numerals and 28,000 (700 each of 40 characters) images of Urdu characters. It is pertinent to mention that we have planned to increase the number of participants to 1500 in order to include as many as possible the variations of different handwriting to create more comprehensive and multifaceted data. The complete data set, after completion, will be made publicly available for the

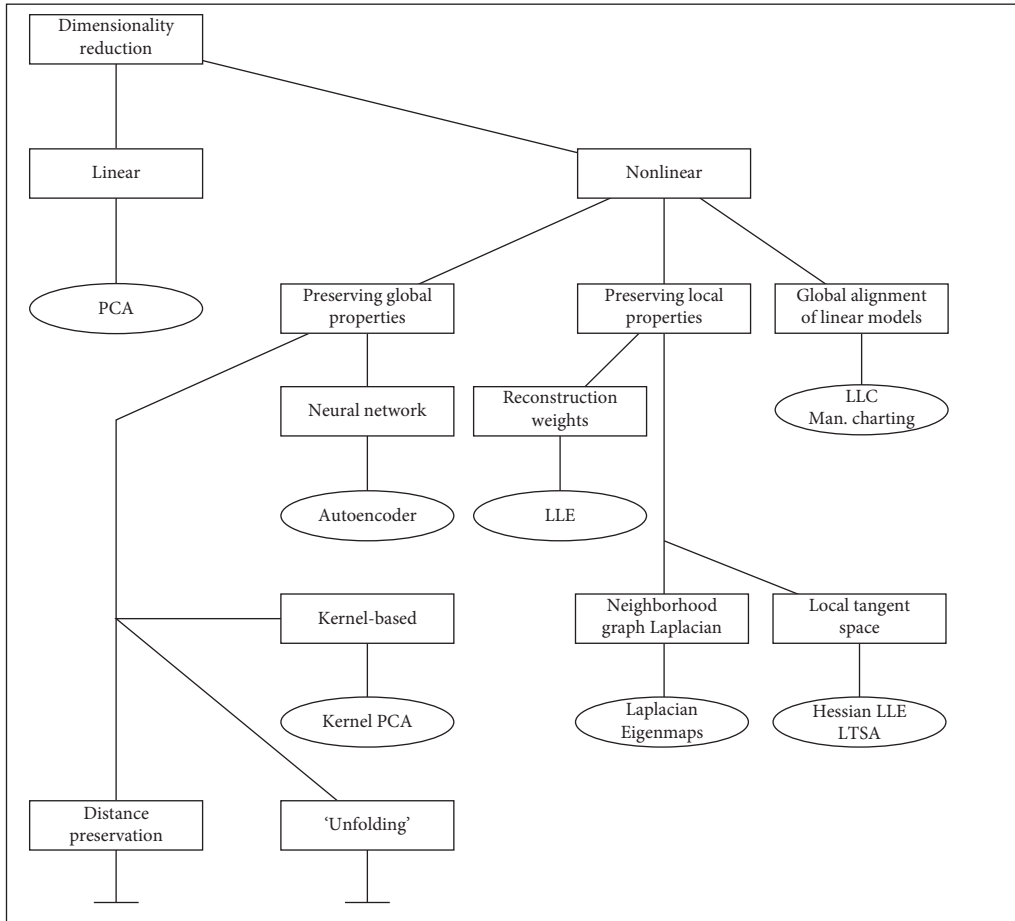


FIGURE 1: Categorization of dimensionality reduction approaches (Van Der Maaten et al. [8]).

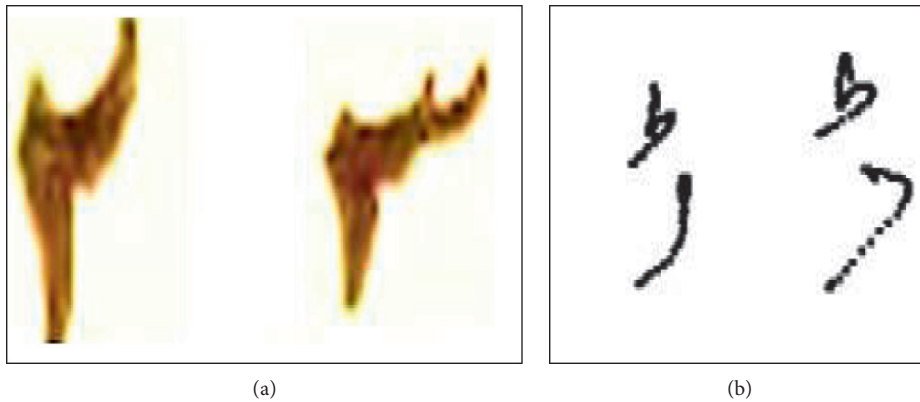


FIGURE 2: Similarity among Urdu characters and numerals.

researchers, since there is no dataset of such kind that is available till date. It is common that noise and distortion may likely occur while scanning the images. In order to remove the noise, we directed the authors to write in black ink only. This activity makes the noise removal process rather trivial task in a way that the colors other than black were considered as noise and removed easily. Furthermore, before applying any dimensionality reduction approach, we make use of some data

transformation algorithms, like gray-scale conversion, image segmentation, image resizing, extracting area of interest from the text image, normalizing the raw data, etc., to prepare our data set in some appropriate form. It is pertinent to mention that the Urdu characters as shown in Figure 4 and numerals share common characters with Arabic and Persian; therefore, our approach is equally applicable in these domains also with some minor modifications.



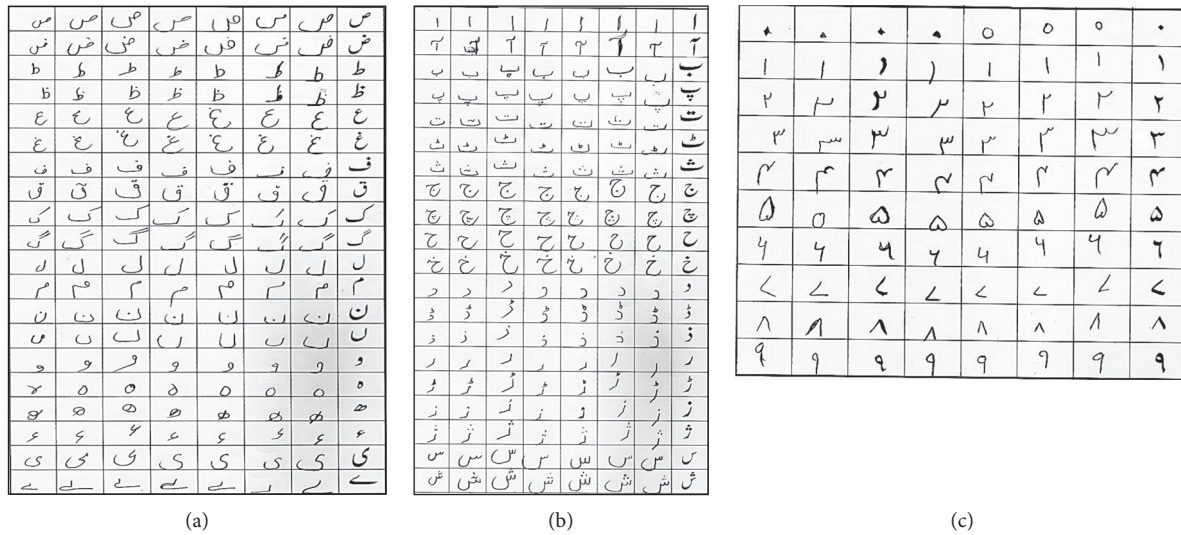


FIGURE 3: Sample pages of our dataset showing Urdu characters in (a) and (b) and Urdu handwritten numerals in (c).

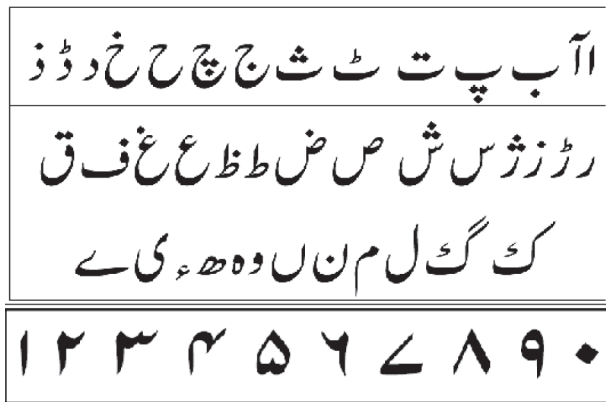


FIGURE 4: Urdu characters and numerals.

## 5. Experimental Results of Dimensionality Reduction Approaches

In this section, we present the results obtained by applying PCA (Roweis and Saul [13]), AE, and  $t$ -SNE on our two variants of the dataset. One variant contains the pixel-based features of both the Urdu numeric and characters. The second variant contains structural-based features embedded with pixel-based data of both the Urdu numeric and characters. The experimental results depict that adding structural based features presents visualization results with accurate separations among the clusters and also maintains intracluster variations.

As mentioned earlier, we represented each text image in our dataset in  $28 \times 28 = 784$  pixel values (or dimensions). For evaluation purposes, we make use of the three different dimensionality reduction approaches, namely, Principal Component Analysis (PCA) (Roweis and Saul [13]); Autoencoders (Liou et al. [26]); and  $t$ -SNE (Van Der Maaten et al. [8]) to our Urdu handwritten character and numeral dataset.

We used the following parameter setting while using  $t$ -SNE (Van Der Maaten et al. [8]) and its variant while

producing visualization results: the number of iterations,  $T$ , set to 1000 in order to achieve the optimized value of gradient descent, the fine-tuned parameter momentum term,  $\alpha(t)$ , is regulated to 0.5, where  $t$  is less than 250, while  $\alpha(t)$  is set to 0.8 for  $t$  is greater than 250. The initial value of learning rate  $\eta$  is set to 100 that may be regulated for each iteration equipped with a highly adaptive learning rate scheme. It is pertinent to mention that the experiments were executed using varying initial learning rates; however, we observed little variations in the quality of the resulting visualization results. Moreover, along with the other parameters, the perplexity is a tunable parameter that depicts how to correspond and normalize the local and global aspects of the data. In other words, we can say that perplexity helps in finding how many close neighbors each point has. It also has a complex effect on the resulting visualizations, as explained in the original  $t$ -SNE paper (Maaten and Hinton [9]). The selection of an optimal value of perplexity is of significant importance; therefore, one must have to take care, since it can be achieved only by producing multiple visualizations with varying perplexity values. Therefore, in this paper, we chose the best result based on the quality of visualization. Furthermore, it is an interesting fact that both the standard and proposed versions of  $t$ -SNE work equally and uniformly on a single assumed value of the perplexity for the whole dataset. The following subsection depicts the results generated through standard  $t$ -SNE on our proposed data set.

**5.1. Applying Standard  $t$ -SNE.** The results obtained through the standard  $t$ -SNE on the dataset having only the pixel-based information of the Urdu handwritten numerals are shown in Figure 5. It is clear from the results that there are some overlapping clusters while considering pixel-based information only. We performed a series of experiments using  $t$ -SNE on the same dataset using multiple perplexity values. It was observed from the output that, with the perplexity value of 70, the results showed some little

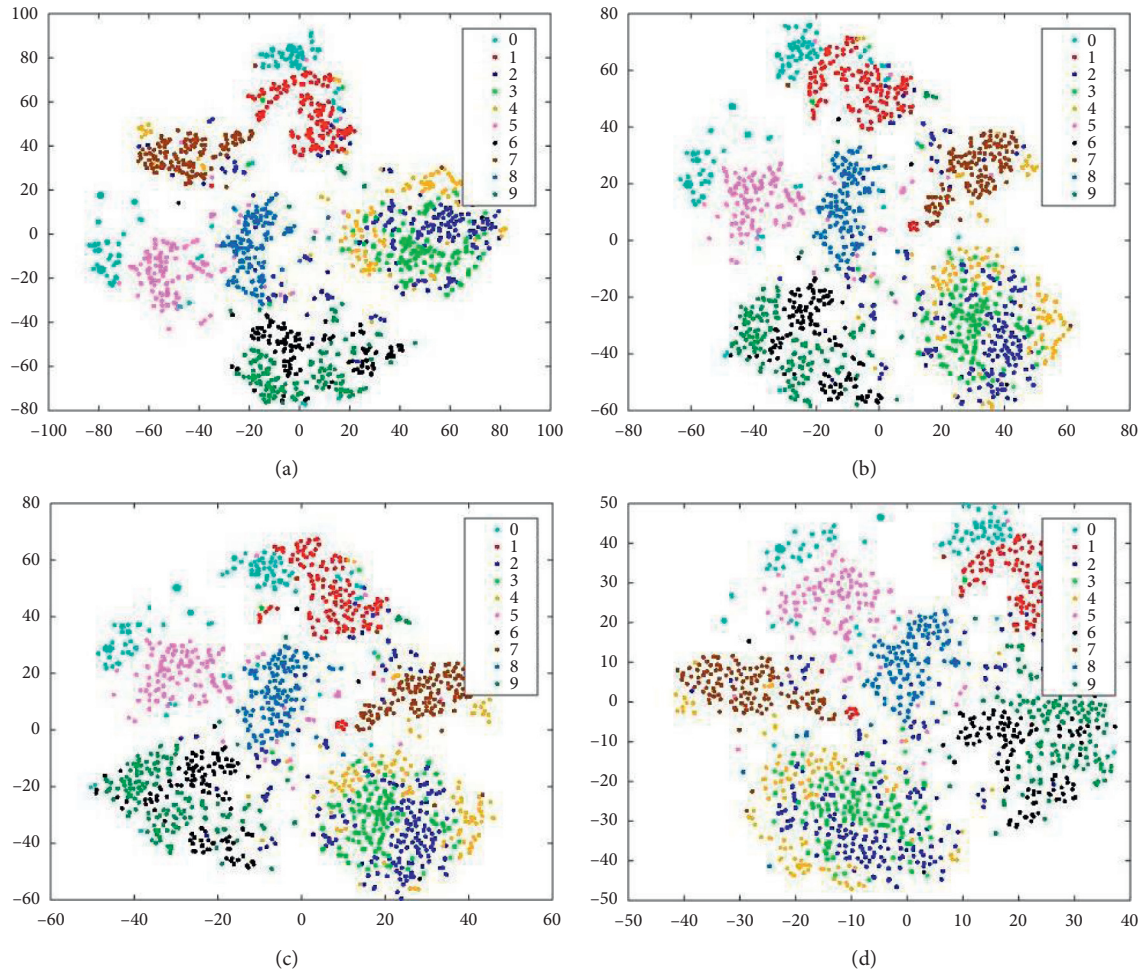


FIGURE 5: Standard  $t$ -SNE based visualization results of the data set containing pixel-based information only of the Urdu handwritten numerals. (a) Perplexity: 30 Iterations: 1,000. (b) Perplexity: 50 Iterations: 1,000. (c) Perplexity: 70 Iterations: 1, 000. (d) Perplexity: 100 Iterations: 1,000.

improvement in terms of separations among the clusters of each class of the Urdu handwritten numerals when compared with results using different perplexity values of 30, 50, and 100.

It is of general observation that, with lower perplexity value, the local structure of the data tends to show higher perseverance, i.e., the clusters having a smaller number of data points plotted very close to each other resulting in a compact visualization. On the other hand, the higher the perplexity value, the higher the perseverance in the global structure of the data; i.e., the data points will be plotted with some notable difference (intracluster difference) and also maintaining the separation between the clusters (intercluster difference).

Figure 6 depicts the detailed description of the structural features of both the Urdu numerics and characters. The results obtained by applying the standard  $t$ -SNE on the dataset having a combination of both the pixel- and structural-based features are shown in Figure 7. The results depicted much improvement by producing the clusters with clear separations. From the results presented in Figure 7, it can be observed easily that some of the Urdu numerals like 2,

3, and 4 have overlapping clusters. This overlapping is based on the fact that these Urdu numerals share much shape similarity. A similar behavior can also be witnessed in the case of Urdu numerals 0 and 1. It can be concluded that combining the pixel- (i.e., 784 features) and structural-based features (i.e., 10 features) failed to be useful when applied to the standard  $t$ -SNE algorithm.

The same approach of standard  $t$ -SNE is applied to the datasets of Urdu handwritten characters. One dataset contains the pixel-based data, and the other contains both the pixel and structural based features. It is pertinent to mention that only those Urdu characters are considered for the experiments that share much of the shape similarity. Figure 8 shows the Urdu characters grouped on the basis of shape similarity. The remaining characters are not considered to reduce the ink-noise ratio while visualizing the 40 characters individually. Therefore, it is better to visualize the characters grouped according to shape similarity.

Figure 9 shows the results of applying standard  $t$ -SNE on both the data set of Urdu handwritten characters mentioned earlier. It is pertinent to mention that the results shown are chosen among the better results produced by

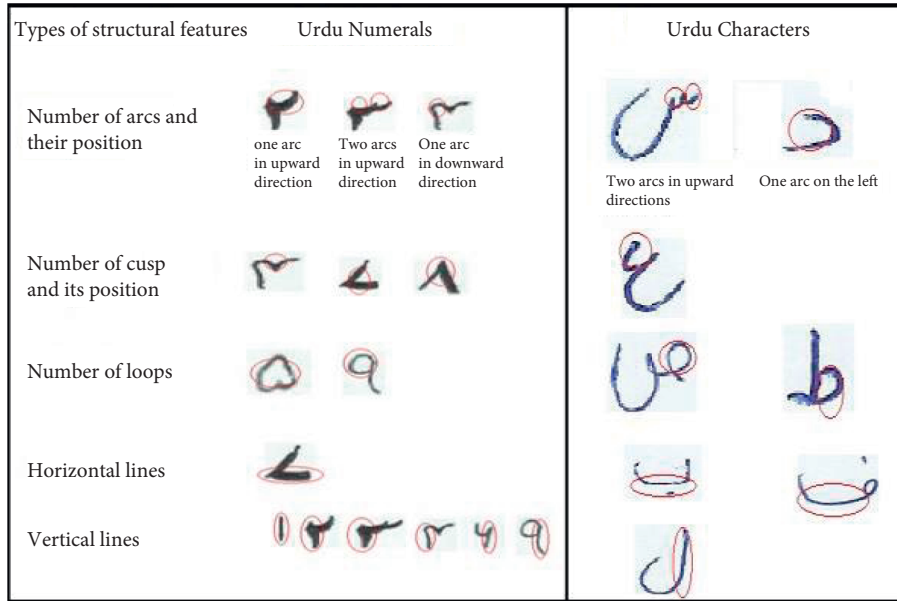


FIGURE 6: Types of structural features of Urdu numerics and characters.

fine-tuning the parameters. One of the results in Figure 9(a) shows that some of the clusters show higher overlapping than other clusters. This overlapping is due to shape similarity among the characters in Groups 2, 10, and 11. A similar behavior is observed in the characters of Groups 3 and 9; Groups 4 and 5; and Groups 6 and 7. Only Group 1, Group 8, and Group 12 characters are correctly drawn by  $t$ -SNE. This issue of overlapping is solved to some extent by embedding structural features of Urdu characters, as we have done with Urdu numerics. The results shown in Figure 9(b) depict the better results. The intracluster separation is better than the previous result. However, there is a need for modifying the standard  $t$ -SNE algorithm to make it capable of generating more precise results. To resolve these issues, a novel idea is proposed to build a fusion matrix having the pair-wise Euclidean distances of more (or multiple) independent observation spaces (i.e., pixel- and structural-based information). The standard  $t$ -SNE is then modified to assist the data in the fused matrix. The details about the fused data matrix and modified  $t$ -SNE are given in the subsequent section.

**5.2. Fused Data Matrix.** In this section, we discussed a novel way to embed the two or multiple observation spaces by calculating the pairwise Euclidean distances of the instances resulting in a fusion matrix. Furthermore, we also modified the standard  $t$ -SNE to make it able to assist the data in a fused matrix form. In our dataset, we build a single fusion matrix by calculating pairwise Euclidean distance of the data instances of the two independent spaces, i.e., pixel- and structure-based information. The resulting fusion matrix is then plugged into the modified  $t$ -SNE that makes use of both the features mentioned earlier. This modified  $t$ -SNE will give equal importance to both the features, thus generating even more clear and accurate clusters with precise separations.

Our assumption lies in the fact that since the data from both the independent spaces sources are highly conjunctive and dependent, therefore, their fusion will produce the more accurate visualization results in some low-dimensional space when the visualization results are compared with the output generated using either the pixel-based data or the structural data alone.

The pixel-based features are saved in a matrix form of size  $1 \times n$  for a single image of Urdu handwritten character, where  $n$  is the pixel-wise binary data for each image of size  $32 \times 3$ . If we consider Urdu numerals (for example), we have used 5000 images of Urdu handwritten numerals (500 each of ten numerals) for our experimental work; therefore, the dataset is of size  $500 \times 32$ . This pixel-based information is then embedded with the structural features, using the Euclidean Distance, of the Urdu handwritten numerals. Since the Urdu numerals share shape similarity, for example, digits two and three (shown in Figure 2), these structural based features are embedded to the pixel-based features to reduce the visualization issues while plotting similar shape images. Furthermore, we introduced equation (1) to balance the weighted combinations of both of the independent original spaces. It is pertinent to mention that  $t$ -SNE works on one of the tunable parameters called perplexity that can be thought of as “the number of neighboring points  $t$ -SNE must consider,” and we used different values of perplexity to encompass the whole data.  $t$ -SNE shrinks widespread data and expands densely packed data. It is, hence, suggested not to decide the size and density/spread/variance of the clusters based on the output. Furthermore, equation (1) is used for calculating the minimum value for the fused Euclidean distances that play a role in winning value for both the independent spaces. This novel activity helped us in performing the fusion process in an efficient and principled way that makes it practically possible for the independent spaces to contribute equally in order to maintain the separation of



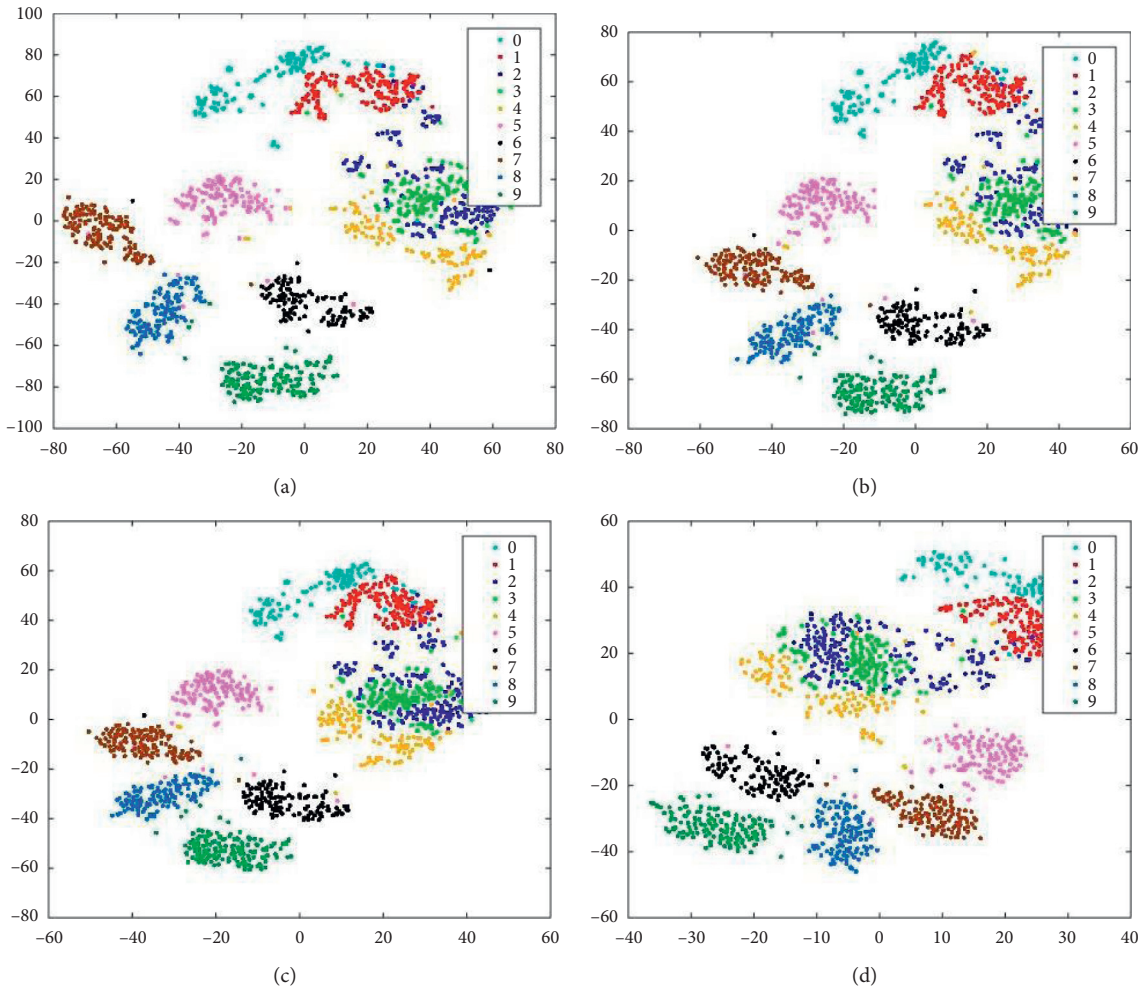


FIGURE 7: Standard (t) -SNE based visualization results of the combined data of pixel- and structure-based information of the Urdu handwritten numerals. (a) Perplexity: 30 Iterations: 1,000. (b) Perplexity: 50 Iterations: 1,000. (c) Perplexity: 70 Iterations: 1,000. (d) Perplexity: 100 Iterations: 1,000.

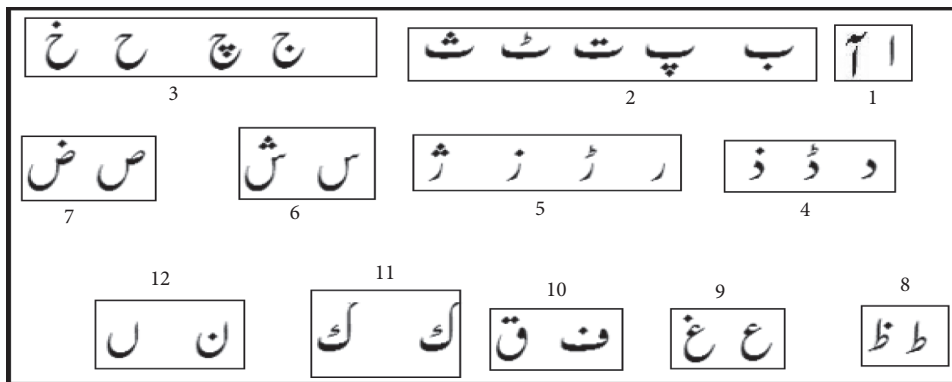


FIGURE 8: Grouping of Urdu characters having shape similarity.



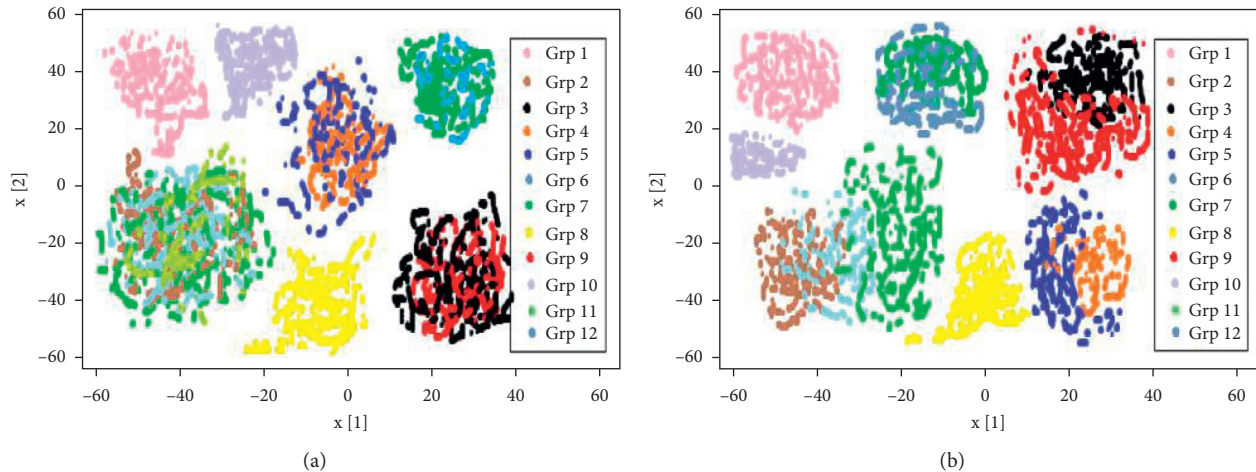


FIGURE 9: Visualization of (a) pixel-based and (b) structure-based data of our dataset of handwritten Urdu characters using standard  $t$ -SNE took 12 minutes 31 seconds. (a) Perplexity: 30 Iterations: 1,000. (b) Perplexity: 50 Iterations: 1,000.

the data instances within a cluster. In order to make the equal contribution of both the independent spaces, we assigned an equal weight ( $\alpha(t) = 0.5$ ) to both spaces.

$$\text{Euclid}_{\text{fused}}(a, b) = (t)\text{Euclid}(P_a, P_{\text{map}_b}) + (1 - \alpha(t))\text{Euclid}(S_a, S_{\text{map}_b}). \quad (1)$$

In equation (1), we computed the similarity patterns in a very disciplined way that are likely to exist in data instances of both the independent spaces where the pixel space value is represented by  $P$  and structural attribute by  $S$ . The relative weight,  $\alpha$ , depicts the relative weight among the similarities of the data instances of both the independent spaces. Whereas the  $t$  depicts the epoch number depicts the number of iterations for the dimensionality reduction process.  $\alpha$  is carefully set to 0.5 to observe the equal contribution of both of the independent spaces. This tuning helps in locating the minimum fused Euclidean distance ( $\text{Euclid}_{\text{fused}}(a, b)$ ), which in return determines the common successful unit by locating. It is pertinent to mention that in order to normalize the Euclidean distances of the two independent spaces (in equation (1)), we apply the product formula. This activity played a key role in improving the results by maintaining the intercluster separations while visualizing in low dimensions. In the next section, we discussed the results obtained by our modified  $t$ -SNE (Van Der Maaten et al. [8]); PCA (Roweis and Saul [13]); and AE (Liou et al. [26]).

The reason for reducing to two-dimensional space is to observe the behavior of the high data that assist in finding the patterns (if they exist). This activity guides the researchers to apply suitable set of classifiers. The resulting 2D features, in our case, are representing the  $(x, y)$  coordinates of each individual instance drawn by the  $t$ -SNE. These 2D features are correctly representing each instance on the map whether they are similar in shape or not. As a result, this information may be used for classification purpose using any

classifier. We used CNN again for reduced dimensional data, since it was used for classification using the pixel-based data only. It is pertinent to mention that 2D does not mean the kernel sliding window; it means that the CNN is accepting two inputs in case of reduced dimension.

**5.3. Complexity Comparison of Standard  $t$ -SNE and Our Modified  $t$ -SNE.** In the original source papers (Maaten and Hinton [9]; Van Der Maaten et al. [8]), the standard  $t$ -SNE's computational and memory cost is  $O(n^2)$ , where  $n$  is the number of data points, which constrain the application of the technique. We evolved the algorithm by reducing the computational complexity to  $O(n \log(n))$  and the memory complexity to  $O(n)$  since the data from both the independent distributions involve a normalization term that sums overall  $n \times (n - 1)$  pairs of unique objects (see Equation 1). It is also observed that the  $t$ -SNE scales quadratically in the number of objects  $n$ , and its applicability is limited to data sets with only a few thousand input objects.

**5.4. Experimental Results Obtained through PCA, AE, and Modified  $t$ -SNE.** In this section, we covered the visualization results of the fused matrix dataset of both the Urdu handwritten numerics and characters. The results showed (see Figure 10) that our modified  $t$ -SNE with fused data matrices of our dataset outperformed the classical approaches of PCA and AE.

Similarly, while visualizing the fused matrix of Urdu handwritten characters, we applied the same set of

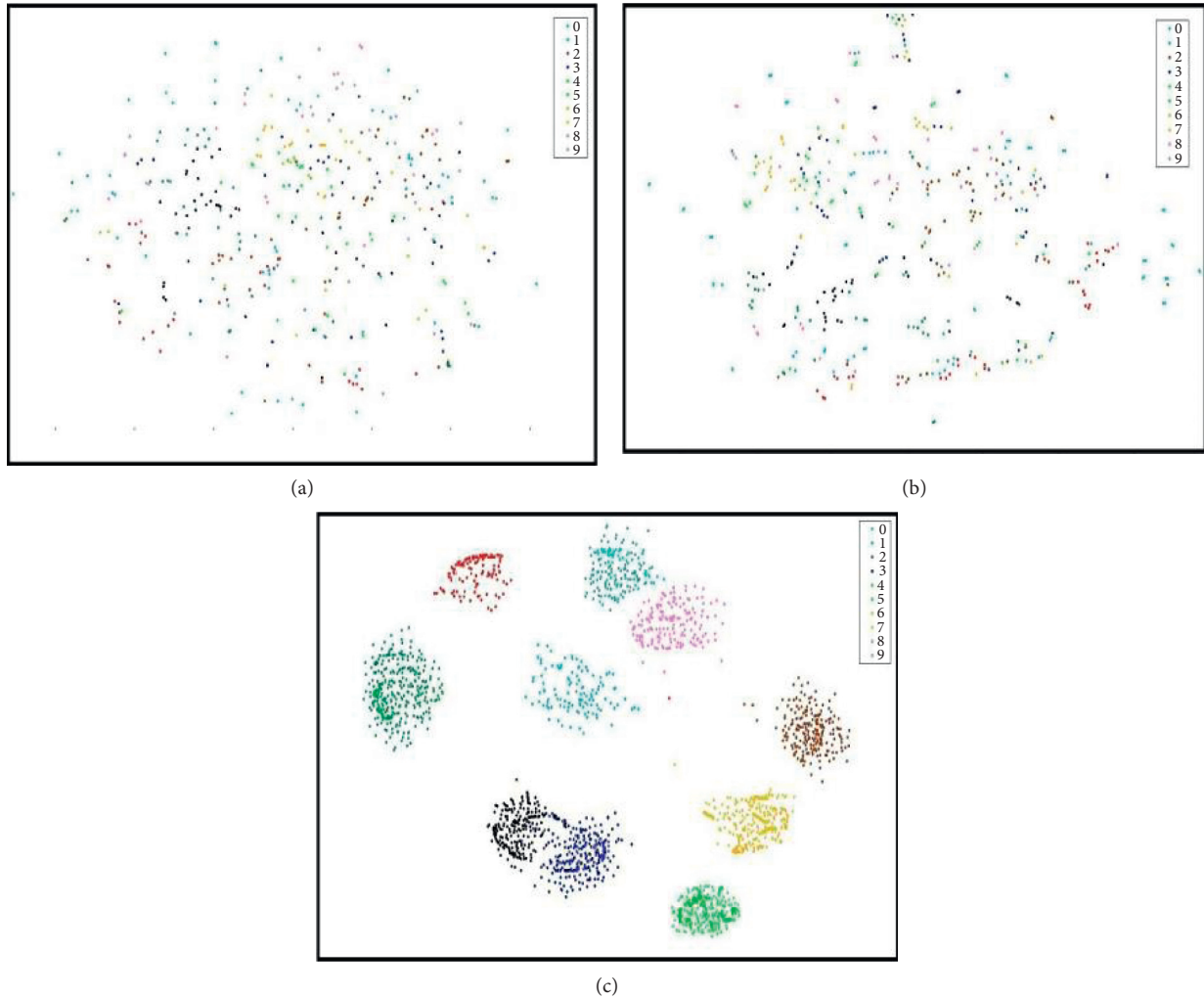


FIGURE 10: The visualization results of the high-dimensional fusion matrix data of the Urdu handwritten Urdu numerals using (a) autoencoders, (b) principal component analysis and (c) *t*-SNE.

algorithms having the same parameter settings. Figure 11 shows the visualization results produced when applied to Urdu handwritten characters.

## 6. Recognition of Urdu Handwritten Characters Using Deep Network

We make use of a deep convolutional neural network (CNN) model with an output layer generating the output on feature mapping in order to recognize the Urdu handwritten characters. CNN is one of the deep networks that are widely used in image classification problems and recognition because of its high accuracy. The CNN follows a hierarchical model, which works on building a network, like a funnel, and finally gives out a fully connected layer, where all the neurons are connected to each other, and the output is processed. Furthermore, we used 2D convolutional layers, which are ideal for processing 2D images. Compared to other image classification algorithms, CNNs actually use very little preprocessing. The key objective of our model is to classify the given input out of 10 classes of the Urdu

handwritten numerals. On the other hand, the same model will also be used in classifying the given Urdu character out of 12 classes of the Urdu handwritten characters (see Figure 8).

In research activities related to image processing, it was observed that CNN and its variants are most widely used. While dealing with two-dimensional images, we used 2 VGG16 model that is equipped with 16- and 19-layer network capable of dealing with a maximum input size of  $224 \times 224$ . It is considered to be one of the excellent vision model architectures till date. The most unique thing about VGG16 is that, instead of having a large number of hyperparameters, it focused on having convolution layers of  $3 \times 3$  filters with a stride 1 and always used the same padding and max-pool layer of  $2 \times 2$  filter of stride 2. While analyzing the high-dimensional data of our manuscript, we came along with the exploitation of both the structural- and pixel-based data in order to generate precise classification results. In order to resolve this issue, we make the standard *t*-SNE compatible with our data by implementing the pair-wise Euclidean distance formula to the data points of our dataset.

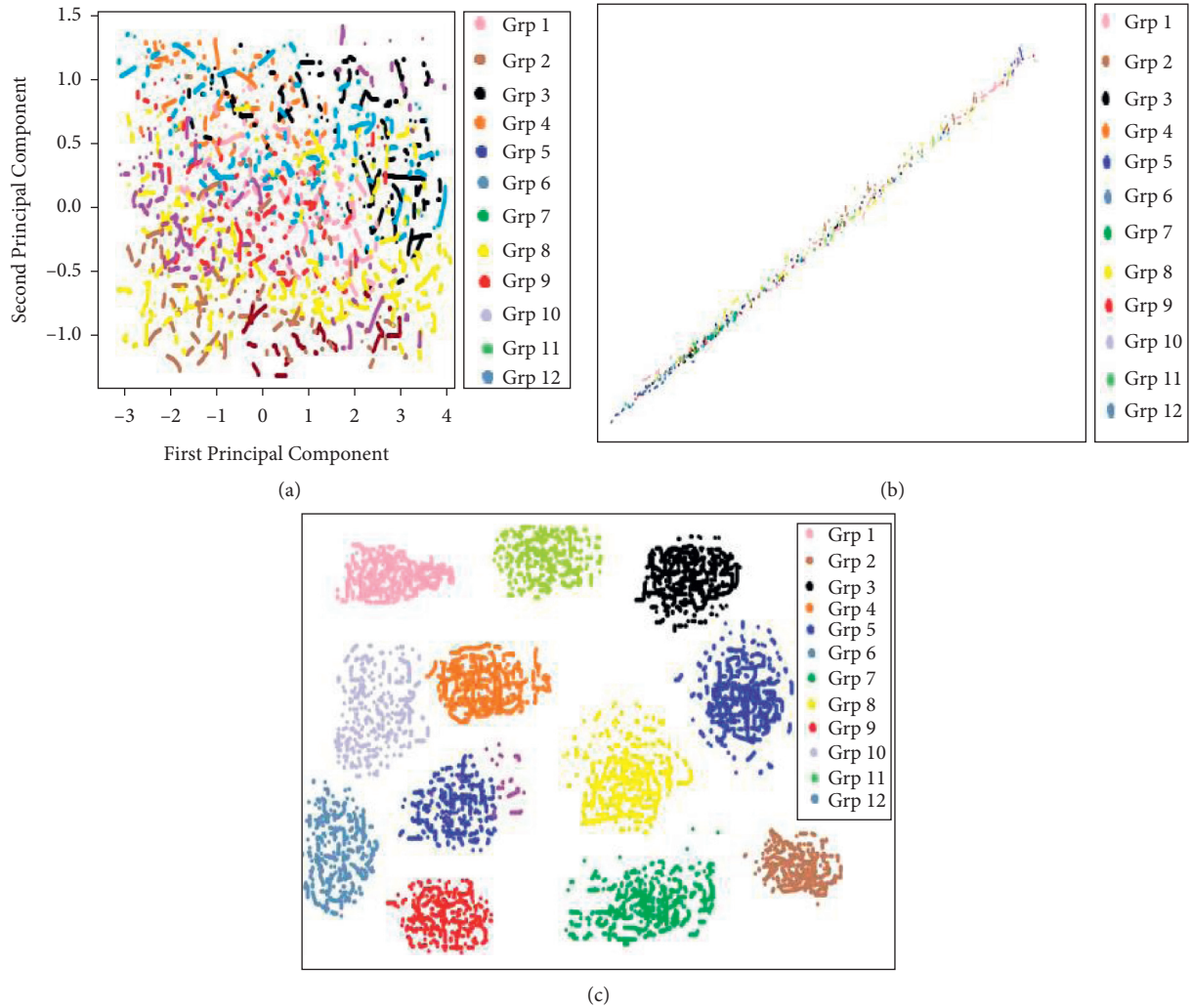


FIGURE 11: The visualization results of the high-dimensional fusion matrix data of the Urdu handwritten Urdu characters using (a) Autoencoders, (b) Principal component analysis, and (c)  $t$ -SNE.

This activity embedded the data points coming from two independent spaces in one space, thus making it compatible with the standard  $t$ -SNE.

It is noteworthy that we have not reduced the size of the image; rather, we have reduced the dimension of the feature-space, that is, the embedded version of structural and pixel-based features. These reduced dimensions are produced by the dimensionality reduction approaches discussed in detail in the sections above. This reduced-dimensional data is then plugged into the proposed model of CNN in order to recognize the numeric (or character) data. This activity takes minimum time (12 CPU seconds) and reported same accuracy rate in the classification of both the Urdu handwritten characters and numerals as compared with the same model applied on the original dimensions of the text images, reported in our work (Husnain et al. [27]) that takes 8 minutes. It is pertinent to mention that there is no need to increase the number of convolutional cores of the proposed model as we have performed in our previous work (Husnain et al. [27]) since

the dimensions of the input data are too small, and they can be handled trivially by the original model of CNN.

In order to decrease the ambiguity in the quality of results, we performed a series of experiments using different variations of the  $n$ -fold cross-validation. This activity helped in retreating the confusion among the biased results obtained through the conventional ratio of training and testing data. Tables 1 and 2 depict the confusion matrices for Urdu handwritten numerals, showing an average accuracy of 96.5% and 94.7%, respectively.

Similarly, Tables 3 and 4 show the results of Urdu handwritten characters (shown in groups in Figure 8). The results showed that our proposed model of CNN outperformed the previous approaches to perform this task, see Table 5. We also presented the comparison of the results produced by our proposed model with some state-of-the-art related approaches for the same task in Table 5. It can be observed that our approach is significantly better in terms of the number of parameters, accuracy, the number of dimensions used, and the amount of calculation.

TABLE 1: The confusion matrix showing accuracy rate of classification of Urdu handwritten numeral using 10-fold cross validation.

Eight	Five	Four	Nine	One	Seven	Six	Three	Two	Zero	Classified
8	5	4	9	1	7	6	3	2	0	As
<b>95%</b>	3%	0	0	0	0	0	0	0	2%	Eight
2%	<b>97%</b>	0	0	0	0	0	0	0	1%	Five
0	0	<b>96%</b>	0	0	0	1%	2%	1%	0	Four
0	0	0	<b>97%</b>	2%	0	1%	0	0	0	Nine
0	0	0	1%	<b>98%</b>	0	1%	0	0	0	One
0	2%	0	0	0	<b>98%</b>	0	0	0	0	Seven
0	0	0	0	3%	0%	<b>95%</b>	0	0	2%	Six
0	0	2%	0	0	0	0	<b>97%</b>	1%	0	Three
0	0	1%	0	3%	0	0	2%	<b>94%</b>	0	Two
1%	1%	0	0	0	0	0	0	0	<b>98%</b>	Zero

The values in bold (in diagonal) are representing the accuracy percentage of the classification of Urdu handwritten numerals.

TABLE 2: The confusion matrix showing accuracy rate of classification of Urdu handwritten numeral using 8-fold cross validation.

Eight	Five	Four	Nine	One	Seven	Six	Three	Two	Zero	Classified
8	5	4	9	1	7	6	3	2	0	As
<b>94%</b>	3%	0	0	0	0	0	0	0	3%	Eight
3%	<b>95%</b>	0	0	0	0	0	0	0	2%	Five
0	0	<b>94%</b>	0	0	0	2%	3%	1%	0	Four
0	0	0	<b>95%</b>	3%	0	2%	0	0	0	Nine
0	0	0	3%	<b>96%</b>	0	1%	0	0	0	One
0	4%	0	0	0	<b>96%</b>	0	0	0	0	Seven
0	0	0	2	4%	0	<b>93%</b>	0	0	3%	Six
0	0	2%	0	0	0	0	<b>96%</b>	2%	0	Three
0	0	3%	0	4%	0	0	1%	<b>92%</b>	0	Two
1%	3%	0	0	0	0	0	0	0	<b>96%</b>	Zero

The values in bold (in diagonal) are representing the accuracy percentage of the classification of Urdu handwritten numerals obtained by applying using variations of n-fold cross-validation approach.

TABLE 3: The confusion matrix showing accuracy rate of classification of Urdu handwritten characters using 10-fold cross validation.

Grp	Grp	Grp	Grp	Grp	Grp	Grp	Grp	Grp	Grp	Grp	Grp	Classified
1	2	3	4	5	6	7	8	9	10	11	12	As
<b>93%</b>	0	0	1%	2%	0	0	2%	1%	1%	0	0	Grp 1
0	<b>89%</b>	0	3%	0	0	0	3%	2%	1%	1%	1%	Grp 2
0	2%	<b>92%</b>	0	2%	1%	0	0	0	0	2%	1%	Grp 3
2%	0	3%	<b>91%</b>	0	2%	0	2%	0	0	0	0	Grp 4
2%	0	5%	0	<b>93%</b>	0	0	0	0	0	0	0	Grp 5
0	0	2%	3%	2%	<b>90%</b>	0	0	2%	0	1%	0	Grp 6
0	0	0	0	0	0	<b>93%</b>	4%	0	0	1%	2%	Grp 7
0	2%	3%	0	0	0	0	<b>87%</b>	4%	1%	1%	2%	Grp 8
3%	1%	0	0	0	0	0	0	<b>89%</b>	0	3%	4%	Grp 9
0	2%	0	0	2%	0	0	3%	1%	<b>90%</b>	0	2%	Grp 10
0	1%	0	0	1%	0	0	0	3%	0	<b>91%</b>	0	Grp 11
3%	2%	0	0	0	1%	1%	0	2%	0	1%	<b>90%</b>	Grp 12

Following are the reasons behind using  $k$ -fold cross validation, and its variants are that the computation time is reduced as we repeated the process only 10 times when the value of  $k$  is 10. It also reduces the biasness in the results when using the conventional 70-30 training-testing ratio, thus limiting the making the classifier to strictly select the data points from the specified training data. Furthermore, every data point gets to be tested exactly once and is used in the training process for at least  $k - 1$  times. Similarly, the variance of the resulting estimate is reduced as the value of  $k$  increases, that is, the reason we make use of 10- and 8-fold cross validation to observe the change in variance.

Our proposed model was found quite efficient (in terms of accuracy) and effective also in performing the recognition and classification tasks among the approaches used so far for the same task. The novelty of our work lies in the fact that the reduced dimensions obtained through different dimensionality reduction approaches are used for the first time for the recognition of the Urdu handwritten characters. Furthermore, our proposed approach is equally applicable for developing an efficient system for both online and offline character recognition for mobile (or handheld) devices for learning applications for children.



TABLE 4: The confusion matrix showing accuracy rate of classification of Urdu handwritten characters using 8-fold cross validation.

Grp	Grp	Grp	Grp	Grp	Grp	Grp	Grp	Grp	Grp	Grp	Grp	Classified
1	2	3	4	5	6	7	8	9	10	11	12	As
<b>91%</b>	3%	0	0	0	0	2%	0	2%	1%	0	1%	Group 1
0	<b>85%</b>	4%	0%	1%	3%	0	2%	0	1%	3%	1%	Group 2
0	0	<b>90%</b>	3%	0	1%	2%	0	0	0	2%	2%	Group 3
1%	1%	2%	<b>89%</b>	0	3%	0	2%	0	1%	0	1%	Group 4
0	0	1%	0	<b>95%</b>	0	1%	0	1%	0	1%	1%	Group 5
0	0	2%	0	0	<b>94%</b>	2%	0	2%	0	0	0	Group 6
0	0	0	0	0	0	<b>93%</b>	4%	0	0	1%	2%	Group 7
0	0	0	0	2%	1%	0	<b>94%</b>	0	1%	1%	1%	Group 8
0	0	1%	0	3%	0	3%	0	<b>87%</b>	0	2%	4%	Group 9
0	0	0	3%	2%	0	2%	0	0	<b>90%</b>	1%	2%	Group 10
0	0%	3%	0	0	0	0	0	4%	0	<b>90%</b>	3%	Group 11
0	0	0	3%	0	2%	1%	0	1%	0	0	<b>93%</b>	Group 12

The values in bold (in diagonal) are representing the accuracy percentage of the classification of Urdu handwritten characters obtained by applying using variations of the n-fold cross-validation approach.

TABLE 5: Comparison Analysis of our proposed approach for Urdu handwritten character classification with state-of-the-art approaches.

Reference	Approach	Features	Accuracy (%)
Accuracy rate of classification of the Urdu handwritten character			
Ali et al. [28]	Neural network	Geometrical strokes	75–80%
Haider and Khan [29]	BPNN, PNN	Geometrical strokes	66%
Shahzad et al. [30]	Linear classifier	Statistical features	66%
Ahmed et al. [31]	BLSTM	Pixel-based	92–94%
Ko and Poruran [32]	SVM	Transfer-learning features	82.30%
Ali et al. [33]	SVM	Pixel-based features	95.79%
Our approach	CNN	Pixel- and geometrical-based	96.04%
Accuracy rate of classification of Urdu handwritten numeral			
Borse and Ansari [34]	Daubechies wavelet	Pixel-based	92.05%
Razzak et al. [35]; Razzak et al. [36]	HMM, fuzzy rule	Pixel-based	97.45%, 97.09%
Sarkhel et al. [37]	Multi-column multi-scale CNN	Non-explicit feature based approach	98.90%
Takruri et al. [38]	(MMCNN) fuzzy C-Means classifier ANN	Pixel-based features	88.00%
Said et al. [39]	ANN	Pixel-based features	94.00%
Mowlaei et al. [40]	ANN	Wavelet-based features	97.34%
Our approach	CNN	Pixel- and geometrical-based	99.01%

## 7. Conclusion

In this paper, we made use of the reduced dimensions obtained through the dimensionality reduction approaches like PCA, AE, and  $t$ -SNE, in recognition of Urdu handwritten characters and numerals. Furthermore, the structural features of each handwritten character are extracted and embedded in pixel-based features to enrich the features of our dataset. In order to make it more compatible, we modified the standard  $t$ -SNE by including equations that support the pairwise Euclidean distances of the features from two independent spaces. This modification results in accelerating the efficiency of standard  $t$ -SNE by producing a quite better low-dimensional data that eventually helped in visualizing both the Urdu handwritten characters and numerals. Furthermore, this reduced dimensional data is fed to the CNN model to recognition purposes. The results produced are quite similar to our previous work, in which we used all the dimensions of the text-images. The only difference is the time efficiency shown by our approach that took about 12 CPU seconds as compared

to our previous work (Husnain et al. [27]) that consumed 12 CPU minutes in producing the confusion matrices. Hence, it can be concluded that, to make the task of classification/recognition of high-dimensional data, it is better to apply a suitable dimensionality reduction approach that will show the faithful representation of the data. Then, plug this low-dimensional data into any machine learning classifier for testing/training to perform recognition/classification tasks. The limitation of our proposed  $t$ -SNE is that the algorithms can only be used to embed or fuse data coming from two or three independent spaces. Generalizations to a higher number of independent spaces are practically not possible since the computation time increases while computing pairwise distances among the high-dimensional data instances.

To the best of our knowledge, a very limited work is observed in the field of handwritten text recognition at character level, and the dataset of this kind is not available to date. Our results are the initiatives towards classification of the handwritten text at character level in the Urdu script, and there may be some lack of quality and comprehensiveness.

Our future work will encompass the recent trends and resolve these issues observed in our current work.

Furthermore, we have also created a state-of-the-art dataset containing the Urdu handwritten characters and numerals; to the best of our knowledge, there is no publicly available dataset of such kind. The existing datasets of Urdu handwritten text mainly consist of the Urdu handwritten words and sentences. The usage of these existing datasets cannot be used efficiently for recognition of the Urdu text at the character level. We also presented a comparative analysis of the results obtained through different approaches to propose recommendations based on parameter tuning. It is also concluded that the deep network can help in performing the recognition and classification task of the handwritten text of cursive scripts in minimum time. Furthermore, our approach also helps in providing a platform to the researchers and developers to develop the applications for the children to learn how to write Urdu (and other cursive languages) characters and numerals with higher accuracy. As mentioned earlier, there is also a deficiency of some standard data repository in the Urdu domain for generating and comparing the benchmark results. In order to bridge this gap, we are working on generating and extending our dataset that will be published publicly in the near future.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare no potential conflicts of interest.

## Acknowledgments

This study was supported by the China University of Petroleum-Beijing and Fundamental Research Funds for Central Universities under Grant no. 2462020YJRC001.

## References

- [1] N. Saeed, H. Nam, M. I. U. Haq, and D. B. Muhammad Saqib, "A survey on multidimensional scaling," *ACM Computing Surveys*, vol. 51, no. 3, p. 47, 2018.
- [2] A. Buja, D. Cook, and D. F. Swayne, "Interactive high-dimensional data visualization," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 78–99, 1996.
- [3] L. H. Nguyen and S. Holmes, "Ten quick tips for effective dimensionality reduction," *PLoS Computational Biology*, vol. 15, no. 6, Article ID e1006907, 2019.
- [4] L. Song, H. Ma, M. Wu, Z. Zhou, and M. Fu, "A brief survey of dimension reduction," in *Proceedings of the International Conference on Intelligent Science and Big Data Engineering*, pp. 189–200, Nanjing, China, October 2018.
- [5] M. H. ur Rehman, C. S. Liew, A. Abbas, P. P. Jayaraman, T. Y. Wah, and S. U. Khan, "Big data reduction methods: a survey," *Data Science and Engineering*, vol. 1, no. 4, pp. 265–284, 2016.
- [6] D. Engel, L. Hu'ttenberger, and B. Hamann, "A survey of dimension reduction methods for high-dimensional data analysis and visualization," in *Visualization of Large and Unstructured Data Sets: Applications in Geospatial Planning, Modeling and Engineering-Proceedings of IRTG 1131 Workshop 2011*, Kaiserslautern, Germany, March 2012.
- [7] C. O. S. Sorzano, J. Vargas, and A. P. Montano, "A survey of dimensionality reduction techniques," 2014, <https://arxiv.org/abs/1403.2877>.
- [8] L. Van Der Maaten, E. Postma, and J. Van den Herik, "Dimensionality reduction: a comparative," *Journal of Machine Learning Research*, vol. 10, no. 66–71, p. 13, 2009.
- [9] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
- [10] F. Camastra, "Data dimensionality estimation methods: a survey," *Pattern Recognition*, vol. 36, no. 12, pp. 2945–2954, 2003.
- [11] J. P. Cunningham and Z. Ghahramani, "Linear dimensionality reduction: survey, insights, and generalizations," *Journal of Machine Learning Research*, vol. 16, no. 1, pp. 2859–2900, 2015.
- [12] F. Tsai, "Comparative study of dimensionality reduction techniques for data visualization," *Journal of Artificial Intelligence*, vol. 3, no. 3, pp. 119–134, 2010.
- [13] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [14] O. Maimon and L. Rokach, *Data Mining and Knowledge Discovery Handbook*, Springer, Berlin, Germany, 2005.
- [15] L. K. Saul, K. Q. Weinberger, J. H. Ham, F. Sha, and D. D. Lee, "Spectral methods for dimensionality reduction," *Semi-supervised Learning*, pp. 293–308, 2006.
- [16] W. S. Torgerson, "Multidimensional scaling: I. theory and method," *Psychometrika*, vol. 17, no. 4, pp. 401–419, 1952.
- [17] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Information Processing Systems*, pp. 585–591, 2002.
- [18] K. Q. Weinberger, F. Sha, and L. K. Saul, "Learning a kernel matrix for nonlinear dimensionality reduction," in *Proceedings of the Twenty-First International Conference on Machine Learning*, p. 106, Banff, Canada, July 2004.
- [19] G. E. Hinton and S. T. Roweis, "Stochastic neighbor embedding," in *Advances in Neural Information Processing Systems*, pp. 857–864, 2003.
- [20] P. Demartines and J. Herault, "Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 148–154, 1997.
- [21] L. Song, A. Gretton, K. Borgwardt, and A. J. Smola, "Colored maximum variance unfolding," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 1385–1392, Vancouver, Canada, December 2008.
- [22] S. B. Ahmed, S. Naz, S. Swati, and M. I. Razzak, "Handwritten Urdu character recognition using one-dimensional blstm classifier," *Neural Computing & Applications*, vol. 31, no. 4, pp. 1143–1151, 2019.
- [23] N. Das, J. M. Reddy, R. Sarkar et al., "A statistical-topological feature combination for recognition of handwritten numerals," *Applied Soft Computing*, vol. 12, no. 8, pp. 2486–2495, 2012.
- [24] M. Husnain, S. Mumtaz, M. Coustaty, M. Luqman, J.-M. Ogier, and S. Malik, "Urdu handwritten text recognition: a survey," *IET Image Processing*, vol. 14, no. 11, 2020.
- [25] M. W. Sagheer, C. L. He, N. Nobile, and C. Y. Suen, "A new large Urdu database for off-line handwriting recognition," in

- Proceedings of the International Conference on Image Analysis and Processing*, pp. 538–546, Vietri sul Mare, Italy, September 2009.
- [26] C.-Y. Liou, W.-C. Cheng, J.-W. Liou, and D.-R. Liou, “Autoencoder for words,” *Neurocomputing*, vol. 139, pp. 84–96, 2014.
- [27] M. Husnain, M. M. Saad Missen, S. Mumtaz et al., “Recognition of Urdu handwritten characters using convolutional neural network,” *Applied Sciences*, vol. 9, no. 13, p. 2758.
- [28] A. Ali, M. Ahmad, N. Rafiq, J. Akber, U. Ahmad, and S. Akmal, “Language independent optical character recognition for hand written text,” in *Proceedings of the 8th International Multitopic Conference INMIC 2004*, pp. 79–84, Lahore, Pakistan, December 2004.
- [29] I. Haider and K. U. Khan, “Online recognition of single stroke handwritten Urdu characters,” in *Proceedings of the IEEE 13th International Multitopic Conference, 2009. INMIC 2009*, pp. 1–6, Islamabad, Pakistan, December 2009.
- [30] N. Shahzad, B. Paulson, and T. Hammond, “Urdu QAEDA: recognition system for isolated Urdu characters,” in *Proceedings of the IUI Workshop on Sketch Recognition*, Sanibel Island, Florida, February 2009.
- [31] S. B. Ahmed, S. Naz, S. Swati, and M. I. Razzak, “Handwritten Urdu character recognition using one-dimensional BLSTM classifier,” *Neural Computing & Applications*, pp. 1–9, 2017.
- [32] M. A. Ko and S. Poruran, “OCR-nets: variants of pre-trained CNN for Urdu handwritten character recognition via transfer learning,” *Procedia Computer Science*, vol. 171, pp. 2294–2301, 2020.
- [33] H. Ali, A. Ullah, T. Iqbal, and S. Khattak, “Pioneer dataset and automatic recognition of Urdu handwritten characters using a deep autoencoder and convolutional neural network,” *SN Applied Sciences*, vol. 2, no. 2, pp. 1–12, 2020.
- [34] R. Borse and I. Ansari, *Offline Handwritten and Printed Urdu Digits Recognition Using Daubechies Wavelet*, ER Publication, New Delhi, India, 2017.
- [35] M. I. Razzak, S. Hussain, A. Beläid, and M. Sher, “Multi-font numerals recognition for Urdu script based languages,” *International Journal of Recent Trends in Engineering*, 2009.
- [36] M. I. Razzak, S. Hussain, and M. Sher, “Numeral recognition for Urdu script in unconstrained environment,” in *Proceedings of the International Conference on Emerging Technologies 2009 ICET 2009*, pp. 44–47, Islamabad, Pakistan, October 2009.
- [37] R. Sarkhel, N. Das, A. Das, M. Kundu, and M. Nasipuri, “A multi-scale deep quad tree based feature extraction method for the recognition of isolated handwritten characters of popular indic scripts,” *Pattern Recognition*, vol. 71, pp. 78–93, 2017.
- [38] M. Takruri, R. Al-Hmouz, A. Al-Hmouz, and M. Momani, “Fuzzy c means based hybrid classifiers for offline recognition of handwritten indian (Arabic) numerals,” *International Journal of Applied Engineering Research*, 2015.
- [39] F. N. Said, A. Yacoub, and C. Y. Suen, “Recognition of English and Arabic numerals using a dynamic number of hidden neurons,” in *Proceedings of the Fifth International Conference on Document Analysis and Recognition. ICDAR’99 (cat. no. pr00318)*, pp. 237–240, Bangalore, India, September 1999.
- [40] A. Mowlaei, K. Faez, and A. T. Haghghat, “Feature extraction with wavelet transform for recognition of isolated handwritten farsi/Arabic characters and numerals,” in *Proceedings of the 2002 14th International conference on Digital Signal Processing DSP 2002 (cat. no.02 th8628)*, vol. 2, pp. 923–926, Piscataway, NJ, USA, July 2002.