Feature Mapping and Deep Long Short Term Memory Network-Based Efficient Approach for Parkinson's Disease Diagnosis

FATIH DEMIR[ID]1, **ABDULKADIR SENGUR**[ID]2, **ALI ARI**3,
KAMRAN SIDDIQUE[ID]4, **(Senior Member, IEEE)**,
AND MOHAMMED ALSWAITTI[ID]4, **(Member, IEEE)**

[1]Biomedical Department, Vocational School of Technical Sciences, Firat University, 23119 Elaziğ, Turkey
[2]Electrical-Electronics Engineering Department, Technology Faculty, Firat University, 23119 Elaziğ, Turkey
[3]Computer Engineering Department, Engineering Faculty, Inonu University, 44000 Malatya, Turkey
[4]Department of Information and Communication Technology, School of Electrical and Computer Engineering, Xiamen University Malaysia, Sepang 43900, Malaysia

Corresponding authors: Kamran Siddique (kamran.siddique@xmu.edu.my) and Mohammed Alswaitti (alswaitti.mohammed@xmu.edu.my)

ABSTRACT In this paper, a novel approach was developed for Parkinson's disease (PD) diagnosis based on speech disorders. When the literature about the speech disorders-based PD diagnosis was reviewed, it was seen that the most of approaches were concentrated on the feature selection as the datasets contained a huge number of features. In contrast, in the proposed approach, instead of eliminating some of the features by using any feature selection method, all features were initially used for forming a mapping procedure where the input feature vectors were converted to the input images. Then, a deep Long Short Term Memory (LSTM) network was employed for PD detection where the obtained images were used. The deep LSTM network carried out both feature extraction and classification processes and its training was carried out in an end-to-end fashion. The activations in the convolutional layer were converted to sequence data through the sequence-folding and sequence-unfolding layers. The activations in the LSTM output with learning parameters were conveyed to the Softmax layer for the classification process. A publically available PD dataset was used in the experimental works and classification accuracy, sensitivity, specificity, precision, and F-score metrics were used for performance evaluation. The obtained accuracy, sensitivity, specificity, precision and F-score values were 94.27%, 0.960, 0.960, 0.910 and 0.930, respectively. The obtained results were also compared with some of the published results and it had seen that most of the achievements of the proposed method are better than the compared methods.

INDEX TERMS Convolutional structure, deep LSTM network, feature mapping, PD diagnosis, speech disorders.

I. INTRODUCTION

Parkinson's disease (PD) is defined as a neurodegenerative disorder that affects some specific parts of the brain [1]. The PD symptoms, which include memory, depression, and movement problems, reveal slowly over the years. Besides the movement problems, including slowness, stiffness, and tremor, balance problems including walking, are revealed at the last stage of PD [2]–[6].

The associate editor coordinating the review of this manuscript and approving it for publication was Jon Atli Benediktsson[ID].

In the last two decades, signal processing and machine learning-based methods have been proposed for the diagnosis of PD. These approaches are generally relied on measuring the motor system disorders caused by the disease. Vocal disturbances from continuous vocal phonations or flowing speech decline are important indicators for most PD patients in the early stage of the disease. To this end, Sakar *et al.* [7] used tunable Q-factor wavelet transform and ensemble learning for voice-based PD detection. The authors constructed a dataset where 756 PD voice signals were collected. Before classification, the authors used the Minimum Redundancy

Maximum Relevance (mRMR) feature selection method for the detection of the most efficient features. The K-Nearest Neighbors (KNN), Multilayer Perceptron (MLP), Random Forest (RF), Linear Regression (LR), Naïve Bayes (NB), and Support Vector Machine (SVM) classifiers were considered in ensemble learning. The reported accuracy score was 86.0%. Pereira *et al.* [8] reviewed the works which were related to the PD in a systematic way. Authors concerned various recent technologies that were important for PD treatment. Some well-known datasets and the methodologies related to the PD were analyzed and the future trends in the treatment of PD were investigated. The authors concluded that in most of the reviewed works, signal-based datasets were considered. Moreover, it was observed that virtual reality and e-health monitoring systems were increasingly used in recent works to increase the life quality of PD patients. Mostafa *et al.* [9] used a general pattern recognition framework for voice disorders-based PD detection. The authors considered multiple feature evaluation approach and classification based method to detect the PD. To this end, a multi-agent system which was composed of five machine-learning methods namely Decision Tree (DT), NB, Neural Network (NN), RF, and SVM and an achievement evaluation system was proposed. Authors mentioned that with that proposed approach, the DT, NB, NN, RF, and SVM classifiers achievements were improved by 10.51%, 15.22%, 9.19%, 12.75%, and 9.13%, respectively. Lahmiri and Shmuel [10] proposed a voice pattern-based approach for PD detection. The authors used eight different feature selection approaches for assessing the impact of the voice patterns for PD detection. In the classification stage of the proposed method, a non-linear SVM technique, which was tuned by a Bayesian optimization, was considered. The receiver operating characteristic (ROC) and the Wilcoxon-based ranking techniques provide the highest sensitivity and specificity. Lahmini *et al.* [11] conducted a study where various machine-learning techniques such as Linear Discriminant Analysis (LDA), KNN, NB, Regression Trees (RT), Radial Basis Function Neural Network (RBFNN), SVM, and Mahalanobis distance classifier (MDC) were examined in the detection of the PD. 22 voice disorder features were considered in the presented work to discriminate the healthy and PD cases. Accuracy, G-mean, and the area under the ROC metrics were used for the performance evaluations of the considered machine learning methods and it was reported that the SVM classifier achieved the best performance. Gürüler [12] developed a hybrid system for the diagnosis of PD. The authors combined the k-means clustering and complex-valued NN for PD detection. More specifically, the k-means algorithm was used for constructing a feature weighting mechanism before NN classification. A dataset that was obtained from speech/sound signals was used in experiments and classification accuracy was used for performance evaluation. The reported accuracy score was 99.52%. Cai *et al.* [13] developed an early PD detection approach that was based on the modified fuzzy k-nearest neighbor method. In the proposed method, vocal measurements were used for PD recognition.

The parameters of the modified fuzzy KNN were tuned by using an optimization technique namely, the evolutionary learning approach. In the evolutionary learning approach, the chaotic bacterial foraging optimization with Gauss mutation was considered. The authors reported that the proposed method outperformed the other compared approaches. Gupta *et al.* [14] presented an approach, which was based on feature selection, for PD diagnosis. The feature selection was accomplished via an enhanced cuttlefish algorithm. After the feature selection operation, the selected features were classified into Parkinson's and normal classes by using DT and KNN classifiers. The experimental dataset was obtained via sound signals and the handwriting samples. According to the classification accuracy, the proposed method obtained an approximately 94% score for PD diagnosis. Sakar *et al.* [6] carried out a comprehensive work where various speech-based features were extracted to construct a dataset for PD. The features in the dataset were obtained from the vowels, words, and sentences. The authors intended to determine which quantity such as the vowels and words or sentences were the most effective in determining the PD. The authors also investigated the central tendency and dispersion metrics for evaluation of the extracted features., traditional machine learning approaches were used in the classification stage. The experimental works indicated that vowel-based features had more effect in PD diagnosis. Orozco-Arroyave *et al.* [15] proposed a study for PD diagnosis which was based on the analysis of continuous speech signals. Four different languages were considered in the proposed work. The speech signals were segmented into voiced and unvoiced frames and twelve Mel-frequency cepstral coefficients and twenty-five bands Bark scales were considered for feature extraction. The reported accuracy scores were in the range of 85% and 99%. Rusz *et al.* [16] used varied speech data which were collected from fourth six Czech native speakers. Twenty-three subjects of the dataset were labeled as PD and the rest were labeled as healthy subjects. Nineteen features were selected, and Wald sequential method was applied to validate the efficiency of each selected feature. Experimental works showed that the fundamental frequency variation was one of the efficient features for early diagnosis of the PD. Moreover, the reported accuracy was 78% for PD versus healthy discrimination. Little *et al.* [17] introduced an approach which was called as "*hoarseness*" diagram. The proposed approach was basically depended on the recurrence and fractal scaling. The proposed method overcame the range limitations of existing speech based PD diagnosis methods by addressing directly the recurrence and fractal scaling together. A bootstrapped classifier was employed and average accuracy score of 91.8% was reported. Tsanas *et al.* [18] proposed an approach based on speech signal processing for discrimination of the PD and healthy cases. Authors extracted 132 dysphonia features from the vowels. Then, four parsimonious subsets of these dysphonia features were selected by using four feature selection methods. The selected features were classified into PD and the healthy classes by using the RF and SVM
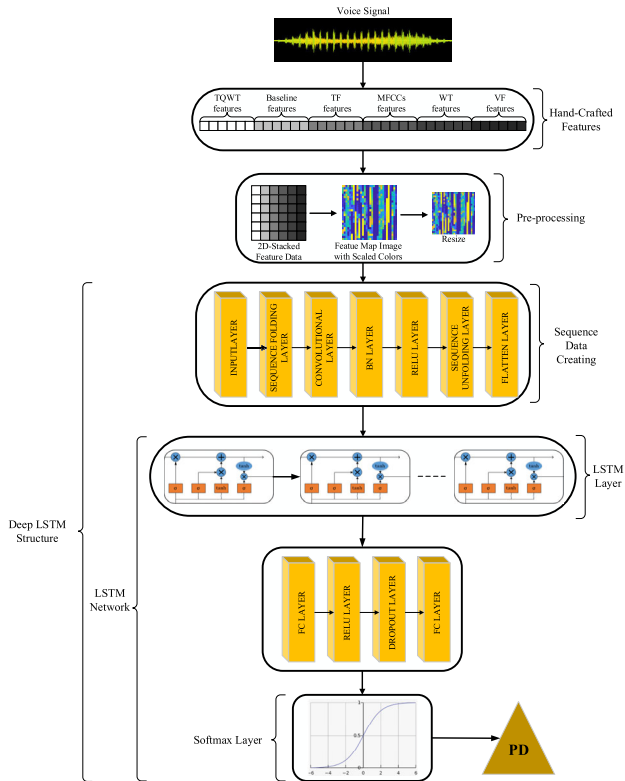
FIGURE 1. Proposed method framework.

classifiers, respectively. By considering ten dysphonia features, the reported accuracy score was 99%.

In this study, a new model was constituted for the detection of PD based on speech disorders. A hand-crafted feature set, which consisted of TQWT features, baseline features, TF features, MFCCs features, WT features, and vocal fold (VF) features, was initially normalized to 0 and 255 ranges to change the values of each feature vector in the dataset to a common scale, without distorting differences in the ranges of values. The normalized feature values were then used to form a feature map image by stacking these features in the columns of a matrix. Thus, the input numerical dataset was converted to the feature map images with the scaled colors technique [19]. Later, the feature map images were resized to 180×180 for sake of convenience with the input of the deep LSTM architecture. The proposed model, which was trained in an end-to-end fashion with the constructed feature map images, was constituted with a sequence data creating structure and an LSTM network respectively. The sequence data creating structure consisted of a convolutional layer, batch normalization layer, and ReLU layer. The activations in the convolutional layer were converted to sequence data through the sequence-folding and sequence-unfolding layers. These triple layers were considered for feature extraction. The LSTM layer was used for learning dependencies in the sequence data. A fully connected layer, ReLU layer, dropout layer, and another fully connected layer were followed the LSTM layer, respectively. The activations in the

LSTM output with learning parameters were conveyed to the Softmax layer for the classification process. The main contributions of this study are as follows;

- In contrast to the related studies, in this study, instead of eliminating some of the features by using any feature selection method, all input features were used for the construction of input images.
- Deep LSTM model with sequence data creation could directly operate image data instead of time-series signals that are used as input data in conventional LSTM models.
- The proposed deep LSTM network was not a combination of CNN and LSTM structures. In studies constituted with CNN and LSTM combination, a pre-trained CNN was used for feature extraction and the LSTM was utilized for the classification task. However, in this study, the proposed deep network was trained in an end-to-end fashion.

The remainder of this paper is as follows. The next section introduces the dataset that is used in this study. The techniques and the methodology of the proposed study are introduced in Section 3 and Section 3's subsections, respectively. The experimental works and results are given in Section 4. The paper is concluded in Section 5.

II. METHODOLOGY
A. DATASET
The used dataset was composed of voice records collected from 188 PD sufferers and 64 healthy subjects in a certain group. Voice records of all subjects were recorded 3 times, and the sample number of the dataset was boosted to 756. To detect speech disorders in the PD sufferers, the speech-based 753 hand-crafted features were extracted from 756 samples in the dataset. 21 baseline features were extracted from techniques such as fundamental frequency parameters and harmonicity parameters [7]. The rest features were constituted by using speech signal techniques containing wavelet transform (WT) based features, time-frequency (TF) features, tunable Q-factor wavelet transform (TQWT) features, Mel frequency Cepstral coefficients (MFCCs), and vocal fold (VF) features extracted from the voice records of PD sufferers. The features based on softening, monotonous, brittle and rapid expression of the voice were extracted from these records.

B. PROPOSED METHOD
A novel and robust approach was proposed to categorize speech disorders of PD sufferers from their voice signals. The illustration of the proposed method was shown in Fig. 1. In this study, we used a dataset that was released by Sakar *et al.* [7]. The dataset was composed of 89 features where there were TQWT features, baseline features, TF features, MFCCs features, WT features, and vocal fold (VF) features, respectively. In some previous works, the authors used various classifications and feature selection approaches
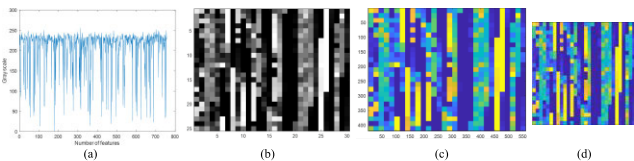
FIGURE 2. The preprocessing steps. (a) Data normalization, (b) Construction of 25 × 30 matrix, (c) Conversion of the matrix into a 420 × 560 × 3 color image, (d) Resized to 180 × 180 × 3 color image.

FIGURE 3. The representation of an LSTM unit.

on the dataset. Different from the previous works, we opted to convert the dataset into images. Then, the obtained images were resized to 180 × 180. Thus, the input to the deep-LSTM network was constructed. In the second step of the proposed approach, the deep-LSTM network was trained from obtained images. The convolutional model in the sequence data creating structure was internally used for the pre-feature extraction. Also, the activations in the convolutional layer were converted to sequence data through the sequence-folding and sequence-unfolding layers. The LSTM network was used for learning dependencies in the sequence data. The activations in the LSTM network output with learning parameters were conveyed to the Softmax layer for the classification process. In the next subsections, the details of the each block are given.

C. PRE-PROCESSING

In the pre-processing, each of the hand-crafted feature vectors was initially normalized into 0 to 255 ranges by using the Equation 1;

$$x_i^{normalized} = 255 \times \frac{(x_i - \min(x_i))}{\max(x_i) - \min(x_i)} \quad (1)$$

here x_i showed the ith feature vector in the given dataset. The normalized feature vectors were then converted to the 8 bits unsigned integers. As the PD dataset contained a total of 750 features for each sample, a matrix of size 25 × 30 was constructed and the converted 8 bits unsigned integer values were stacked into the columns of constructed 25 × 30 matrix. Thus, a sequence of 750 features was represented by a 25 × 30 matrix. The scaled colors technique was used to convert the matrix into a 420 × 560 × 3 color image. The obtained images were then resized to 180 × 180 × 3 color images. Fig. 2 shows all steps in the pre-processing stage of the proposed.

D. DEEP STRUCTURE

The sequence-folding layer turns the sequence image into an image set and convolutional processes are implemented to these images with certain periods. After convolutional activations, the sequence-unfolding layer turns output data of the convolutional layer into sequence data for application to the LSTM network. Convolution operation denoted as "*" is the core function of the convolutional layer. The input data and learnable filter are used in the convolution operation process [20]–[22]. The learnable filter can be in different sizes such as 3 × 3 and 5 × 5, and the padding process
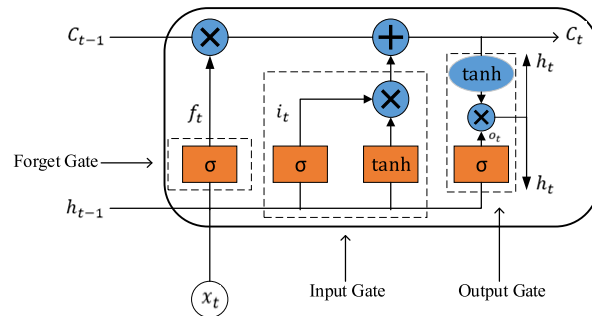
can be also used as an option in the convolutional layer for tuning the convolutional area. The main target of convolution operation is to constitute features by finding similar regional parts of samples in the dataset and assign them into a feature mapping matrix [23]. The convolution function for 2D data at the discrete-time is as follows:

$$(I * F)(i, j) = \sum_m \sum_n F(m, n) X(i - m, j - n) \quad (2)$$

Here, I and F symbolizes the inputs and learnable filters, respectively.

In the deep learning models, the batch normalization (BN) layer is utilized to decrease training time and enhance network initialization performance [24]. Besides, the vanishing gradient problem is reduced with operations in the BN layer. By using the mean of mini-batch b_m and variance mini-batch b_v of the input data, the BN layer output y_i can be calculated as follows:

$$\hat{x}_i = \frac{x_i - b_m}{\sqrt{b_v^2 + \epsilon}} \quad (3)$$

$$y_i = b\hat{x}_i + a \quad (4)$$

Here $\hat{x}_i$ is the normalized activation, and Constant ϵ is used to balance the numerical result if b_v is very small. Scale variable a and balance variable b, which are learnable parameters are updated for the best y_i during optimization.

An activation function such as sigmoid and tangent is frequently applied to the ANN models for obtaining a nonlinearity characteristic in the network. However, a sigmoid and tangent function can cause gradient vanishing and explosion problems in big scaled networks such as deep learning models [25], [26]. Thus, Rectified Linear Unit (ReLU) layer is frequently used as an activation function. In the ReLU layer, the calculation is as follows:

$$f(x) = \max(0, x) \quad (5)$$

According to Equation 4, the input data equals zero if it is negative, otherwise, the input is equalized to the output.

In the flatten layer, 2D data conveyed from the previous layer is turned into 1D data for transmitting to the FC layer.

LSTM containing the units which have a controlled structure consisting of three gates (input, output and forget), is a recurrent neural network (RNN) model [27]. The LSTM

unit, which is given in Fig. 3, keeps data determined in a prior period through these gates, and these gates control the data transmission in the units [28]. Besides, the LSTM layer significantly reduces the gradient vanishing and explosion problems.

The forget gate structure resembles a single-layer neural network (SLNN). According to Equation 6, the forget gate activates if the output is equal to 1.

$$f_t = \sigma(W[x_t, h_{t-1}, C_{t-1}] + b_f) \tag{6}$$

where, x_t represents the input of the existing LSTM unit, h_{t-1} represents output vector of a prior LSTM unit, C_{t-1} represents the memory of prior LSTM unit, b_f represents the biased values, σ represents the sigmoid activation function and W represents the weighted vector. The input gate consists of a structure where the existing recollection is composed with an SLNN and the prior recollection unit data. These computations are expressed in Equations 7 and 8.

$$i_t = \sigma(W[x_t, h_{t-1}, C_{t-1}] + b_i) \tag{7}$$

$$C_t = f_t.C_{t-1} + i_t.\tanh([x_t, h_{t-1}, C_{t-1}]) + b_c \tag{8}$$

The output gate collects data and information transmitting from the existing LSTM unit. The computations in the output gate are given in Equations 9 and 10.

$$\sigma_t = \sigma(W[x_t, h_{t-1}, C_{t-1}] + b_o) \tag{9}$$

$$h_t = o_t.\tanh(C_t) \tag{10}$$

The structure of the FC layer is similar to a multilayer perceptron (MLP). The neurons in the FC layer provide clues about how well a value fits any class [29]. The dropout layer randomly sets input units to 0 with a frequency of rate at each step during training time, which aids obstruct overfitting [30]. The softmax layer with the class possibility values classifies data in the last FC layer. The softmax function is used in the classification stage for CNNs. The softmax function is expressed as follows:

$$S^k = \frac{e^{x^k}}{\sum_{i=1}^{n} e^{x^n}} \tag{11}$$

The output vector S^k is calculated for each input value (x^k), and the sums of all output values are equal to 1 [25].

E. PERFORMANCE EVALUATION METRICS

The true positive (TP), false positive (FP) true negative (TN), false negative (FN) numbers in the confusion matrix were utilized to evaluate the proposed approach. The evaluation metrics were constituted by using the accuracy (ACC), sensitivity (Sn), specificity (Sp), precision (Pr), and F-score value. The evaluation metrics were calculated as follows:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

$$Sn = \frac{TP}{TP + FN} \tag{13}$$
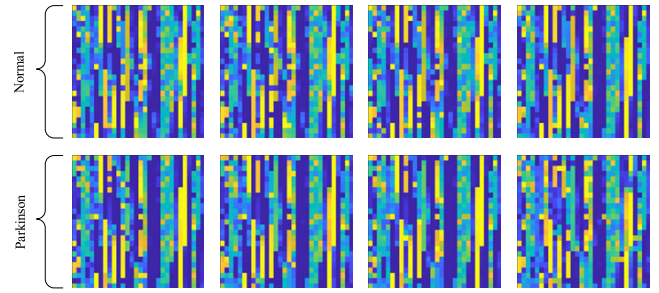
$$Sp = \frac{TN}{TN + FP} \tag{14}$$

FIGURE 4. Some example images from the dataset. The first row shows the normal cases and the second row shows the PD cases.

TABLE 1. The layer properties of the proposed deep LSTM network.

Order	Layer	Layer info	Layer output size
1	Sequence Input	Sequence input with 180×180×3 dimensions	180×180×3
2	Sequence Folding	-	180×180×3
3	Convolution	Convolutions with 20 filters, 5x5x3 filter size, stride [1 1] and padding [0 0 0 0]	176×176×20
4	Batch Normalization	Batch normalization with 20 channels	176×176×20
5	ReLU	Fisrt ReLU (ReLU1)	176×176×20
6	Sequence Unfolding	Cell data	{176×176×20}
7	Flatten	Cell data	{619520×1}
8	LSTM	LSTM with 100 hidden units	100
9	Fully Connected	First fully connected layer (fc1)	350
10	ReLU	Second ReLU (ReLU2)	350
11	Dropout	40% dropout	350
12	Fully Connected	Last fully connected layer (fc2)	2
13	Softmax		2
14	Classification Output	crossentropyex with classes 'normal' and 'parkinson'	1

$$Pr = \frac{TP}{TP + FP} \tag{15}$$

$$F-score = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{16}$$

III. EXPERIMENTAL STUDIES

All experimental studies were carried out on MATLAB. The image processing toolbox was used for the conversion of the numerical data to image data. The deep learning toolbox was used for the construction of the proposed deep LSTM model. As the dataset contains 756 samples, 756 images were used in the training/testing of the proposed deep LSTM model. Randomly selected 70% and 80% datasets were used in training and the rest 30% and 20% were used for performance testing of the proposed approach. The proposed deep LSTM structure was trained in an end-to-end fashion. Fig. 4 shows some sample images from both PD and normal cases.

Since there are enough samples in each class, supervised learning, not one-shot learning, was used in the proposed method. The ground truth labels were embedded to the

training options since the proposed deep-LSTM contained an end-to-end learning strategy. In the proposed approach during training and test, the prediction results obtained at the classification output were logically ("and" operator) compared with the ground truth results using the cross entropy loss function in the classification layer.

The first row of Fig. 4 shows the samples of the normal cases and the second row indicates the sample PD images. The layer properties of the proposed deep LSTM network were given in Table 1. As seen in Table 1, the sequence folding process was applied to the input images initially, and then obtained structures were conveyed to the network through the set of layers containing Convolution, BN, and ReLU layers. Sequence unfolding, flatten, LSTM, fc1, ReLU2, dropout, fc2, softmax, and classification layers came after the ReLU1 for the construction of the whole network. The stochastic gradient descent with momentum (SGDM) algorithm was used as optimization solver. At the training options, the mini-batch size was selected as maximum 32 due to the hardware capacity. Also, the initial learning rate and the epoch number was tuned as 0.001 and 200, respectively.

From Table 1, it is seen that the convolution layer has 20 filters of size $5 \times 5 \times 3$. After the convolution layer, the input size became $176 \times 176 \times 20$ for each input image. Batch normalization and ReLu layers did not change the dimensions of the input. After the flatten layer, the 3-dimensional input data was converted to the 1-dimensional data, where the dimension was 61950×1. The LSTM layer has 100 hidden units and the first FC layer contained 350 units. After ReLU, dropout, and last FC layer, the softmax, and classification output layers were located in the proposed deep LSTM network model. Besides accuracy, sensitivity, specificity, precision, and F-score measures, the receiver operating characteristic (ROC) curve was also used in the performance evaluation of the proposed method. Fig. 5 shows the training and the loss curves of the proposed deep LSTM model. As seen in Fig. 5(a), the training accuracy was started around the 30% accuracy level and gradually increased above to 90% accuracy around the 75th iteration. Around the 175th iteration, the training accuracy was reached to almost 100% training accuracy. The loss of the proposed deep LSTM network was around 0.7 when the training operation has just started.

And, it gradually came under 0.1 around the 175th iteration. Around the 275th iteration, the loss curve settled to around 0.05 value. Fig. 5(b) shows the training accuracy and loss plots when 80% of the dataset was used for training. The training accuracy was started around the 40% accuracy level and gradually increased above 90% accuracy around the 50th iteration. Around the 100th iteration, the training accuracy was reached to almost 100% training accuracy. The loss curve was around 0.7 when the training operation has just started. And, it gradually came under 0.1 around the 100th iteration. Around the 200th iteration, the loss curve settled to around 0.01 value. When Figs. 5(a) and 5(b) were compared, it was seen that the 80%-20% training-test set achieved better training than the 70%-30% training-test set.
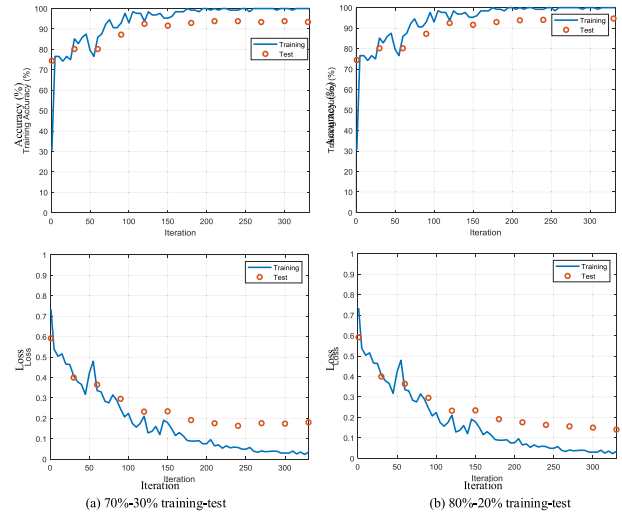

FIGURE 5. The training-test and the loss curves of the proposed deep LSTM architecture. (a) Accuracy curve, (b) Loss curve.
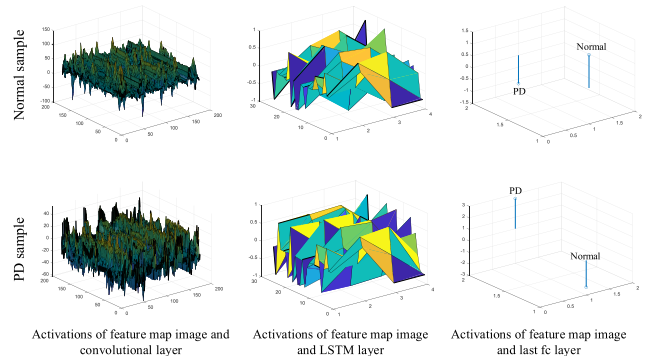

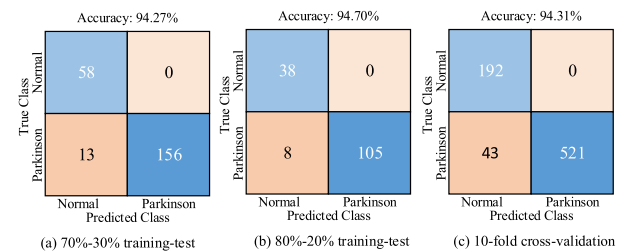**FIGURE 6.** Optimized activations of deep-LSTM structure.

FIGURE 7. Confusion matrix test results.

Fig. 6 shows the optimized activation outputs of the proposed deep LSTM structure. While the first column of Fig. 6 shows the activations of the feature map image and the convolution layers, respectively, the second and third columns of Fig. 6 show the activations of the feature map image and the LSTM layer and the activations of the feature map image and the last fully connected layer (fc2), respectively.

Fig. 7 shows the obtained confusion matrix and accuracy score of the proposed method for 70%-30% training-test, 80%-20% training-test, and 10-fold cross-validation, respectively. As seen the overall classification accuracy scores were
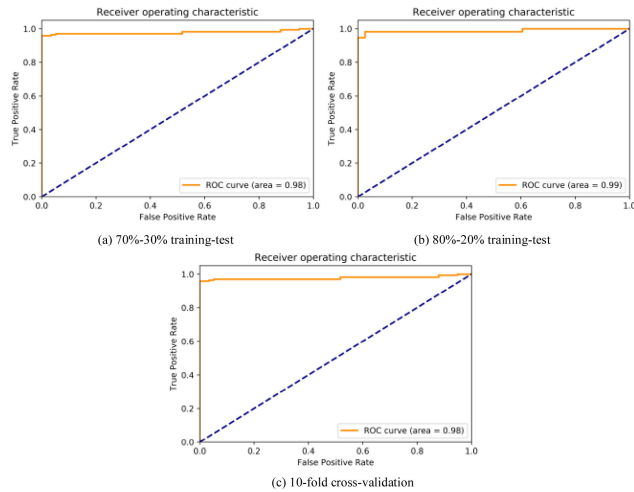
FIGURE 8. ROC curve representations of the proposed method.

TABLE 2. Other evaluation metrics.

Training-test rate (%)	Class	Sn	Sp	Pr	F-score
70-30	Normal	1.000	0.923	0.817	0.900
	Parkinson	0.923	1.000	1.000	0.960
80-20	Normal	1.000	0.929	0.826	0.904
	Parkinson	0.929	1.000	1.000	0.963
10-fold cross-validation	Normal	1.000	0.817	0.817	0.899
	Parkinson	0.817	1.000	1.000	0.960

TABLE 3. Performance comparison of the proposed method with other methods.

Methods	ACC (%)	Sn	Sp	Pr	F-score
Baseline method [7]	86.00	-	-	-	0.840
Other method [31]	93.80	0.840	0.970	0.915	-
Proposed Method	94.27	0.960	0.960	0.910	0.930

94.27%, 94.70%, and 94.31% for 70%-30% training-test and 80%-20% training-test, and 10-fold cross-validation, respectively. From the confusion matrix of 70%-30% training-test set, it was seen that all normal cases were correctly classified and 13 PD cases were classified as PD. For the confusion matrix of 80%-20% training-test set, while all normal cases were correctly classified, 8 PD cases were classified as PD. For 10-fold cross-validation, while all normal cases were correctly classified, 43 PD cases were classified as PD. Table 2 shows the sensitivity, specificity, precision, and F-score values for each class for both training and test sets. For the normal class when a 70%-30% training-test set was used, the obtained sensitivity, specificity, precision, and F-score values were 1.000, 0.923, 0.817, and 0.9, respectively. Similarly, the obtained sensitivity, specificity, precision, and F-score values for the PD class were 0.923, 1.000, 1.000, and 0.96, respectively. When 80%-20% training-test set was considered, for normal cases, while 1.000, 0.929, 0.826, and 0.904 values were obtained for the sensitivity, specificity, precision, and F-score, for PD cases, 0.929, 1.000, 1.000, and 0.963 scores were obtained for the sensitivity, specificity, precision, and F-score, respectively. When 10-fold cross-validation was considered, for normal cases, while 1.000, 0.817, 0.817, and 0.899 values were obtained for the sensitivity, specificity, precision, and F-score, for PD cases, 0.817, 1.000, 1.000, and 0.960 scores were obtained for the sensitivity, specificity, precision, and F-score, respectively.

The ROC curve representations of the proposed method were also given in Fig. 8. The x-axis of the ROC curve shows the false positive rate and the y-axis shows the true positive rate. In Fig. 8(a), it was seen that the true positive rate was about 0.92 when the false positive rate was 0. Besides, the true positive rate was 1 when the false positive rate was 1 when the 70%-30% training-test set was used. The area under the ROC curve (AUC) was 0.98. The ROC curve for the 80%-20% training-test set was given in Fig. 8(b). The

obtained ROC curve for the 80%-20% training-test set was quite similar to the ROC curve of the 70%-30% training-test set. The calculated AUC for the 80%-20% training-test set was 0.99. The ROC curve for 10-fold cross-validation was given in Fig. 8(c). The obtained ROC curve for 10-fold cross-validation was quite close to the ROC curve of the 70%-30% training-test set. The calculated AUC for 10-fold cross-validation was 0.98.

The comparison of the obtained results with some of the previously published results was given in Table 3. The dataset of other existing methods for the training and testing process is split in 70% and 30%, respectively. Therefore, the same training test rates of the proposed approach were used in Table 3. The first row of Table 3 shows the baseline results that were reported by Sakar *et al.* [7]. In [7], the authors only calculated the accuracy and F-score metrics. The calculated accuracy and F-score metrics were 86.0% and 0.84, respectively.

In [31], the authors proposed a two-level feature selection approach for PD detection. The reported achievement scores were given in the second row of Table 3. As can be seen in Table 3, the reported accuracy sensitivity, specificity, and precision values were 93.8%, 0.84, 0.97, and 0.915, respectively. The performance evaluation scores for the proposed method were also given in the third row of the Table 3. The obtained accuracy sensitivity, specificity, precision and F-score values were 94.27%, 0.960, 0.960, 0.910 and 0.930, respectively. So, when accuracy, sensitivity, precision and F-score metrics were considered, it was seen that the proposed method outperformed other methods. Only the specificity score of the reference [31] was higher than the proposed method's specificity score. However, since the training-test samples are

randomly selected, it cannot be said that the approaches that produce similar results are completely superior to each other.

IV. CONCLUSION

In this paper, a novel approach was developed for PD diagnosis based on speech disorders. The proposed approach is based on feature mapping and deep LSTM structure, respectively. The input dataset, which contains 750 features and 756 samples, is used to obtain 756 images of size $180 \times 180 \times 3$. 70% of these images are used in training of the proposed deep LSTM structure and the rest images are used in testing. The obtained accuracy, sensitivity, specificity, precision and F-score values are 94.27%, 0.960, 0.960, 0.910 and 0.930, respectively. When the 80%-20% training-test set was used, the obtained accuracy, sensitivity, specificity, precision, and F-score values were 94.70%, 0.9645, 0.9645, 0.9130, and 0.9335, respectively. Besides, for 10-fold cross-validation, the obtained accuracy, sensitivity, specificity, precision and F-score values are 94.31%, 0.908, 0.908, 0.908 and 0.929, respectively. The obtained scores are compared with some of the existing results and it is seen that the proposed method produces promising scores. This is because the proposed approach used a specific deep learning architecture that trains the convolutional model and the LSTM model together. Hand-crafted features are used in both of the compared models. Deep learning architectures are often better at extracting features than traditional methods. In addition, the LSTM strategy in the proposed method increased the classification performance by keeping the more important ones in memory among the extracted features. However, the robust hardware for the proposed approach is needed to operate with high-resolution image data and a bigger network.

APPENDIX

The pseudocode of the proposed approach

Input: 1D-data with 750 size (hand-crafted features) N = The number of sample (756)

1: Normalize feature vector between 1 and 255
2: Convert 1D data to 2D data for each samples (25×30)
3: Create feature mapping based-RGB images ($420 \times 560 \times 3$) from 2D data with the Colormap technique
4: Resize RGB images to the input layer size of deep LSTM ($180 \times 180 \times 3$)
5: Split dataset randomly at 70% training 30% test ratio (training:$N \times 0.7$ and test: $N \times 0.3$)
6: Train the proposed approach from scratch
7: Test the proposed approach with the trained model and test dataset

---

REFERENCES

[1] J. Jankovic, "Parkinson's disease: Clinical features and diagnosis," *J. Neurol., Neurosurg. Psychiatry*, vol. 79, no. 4, pp. 368–376, 2008.

[2] A. Tsanas, M. A. Little, P. E. McSharry, and L. O. Ramig, "Accurate telemonitoring of Parkinson's disease progression by noninvasive speech tests," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 4, pp. 884–893, Apr. 2010.

[3] C. G. Goetz, G. T. Stebbins, D. Wolff, W. DeLeeuw, H. Bronte-Stewart, R. Elble, M. Hallett, J. Nutt, L. Ramig, T. Sanger, A. D. Wu, P. H. Kraus, L. M. Blasucci, E. A. Shamim, K. D. Sethi, J. Spielman, K. Kubota, A. S. Grove, E. Dishman, and C. B. Taylor, "Testing objective measures of motor impairment in early Parkinson's disease: Feasibility study of an at-home testing device," *Movement Disorders*, vol. 24, no. 4, pp. 551–556, Mar. 2009.

[4] P. Calabresi, B. Picconi, L. Parnetti, and M. Di Filippo, "A convergent model for cognitive dysfunctions in Parkinson's disease: The critical dopamine–acetylcholine synaptic balance," *Lancet Neurol.*, vol. 5, no. 11, pp. 974–983, 2006.

[5] A. H. Friedlander, M. Mahler, K. M. Norman, and R. L. Ettinger, "Parkinson disease: Systemic and orofacial manifestations, medical and dental management," *J. Amer. Dental Assoc.*, vol. 140, no. 6, pp. 658–669, 2009.

[6] B. E. Sakar, M. E. Isenkul, and C. O. Sakar, "Collection and analysis of a Parkinson speech dataset with multiple types of sound recordings," *IEEE J. Biomed. Health Inform.*, vol. 17, no. 4, pp. 828–834, Jul. 2013.

[7] C. O. Sakar, G. Serbes, A. Gunduz, H. C. Tunc, H. Nizam, B. E. Sakar, M. Tutuncu, T. Aydin, M. E. Isenkul, and H. Apaydin, "A comparative analysis of speech signal processing algorithms for Parkinson's disease classification and the use of the tunable Q-factor wavelet transform," *Appl. Soft Comput.*, vol. 74, pp. 255–263, Jan. 2019.

[8] C. R. Pereira, D. R. Pereira, S. A. Weber, C. Hook, V. H. C. de Albuquerque, and J. P. Papa, "A survey on computer-assisted Parkinson's disease diagnosis," *Artif. Intell. Med.*, vol. 95, pp. 48–63, Apr. 2019.

[9] S. A. Mostafa, A. Mustapha, M. A. Mohammed, R. I. Hamed, N. Arunkumar, M. K. A. Ghani, M. M. Jaber, and S. H. Khaleefah, "Examining multiple feature evaluation and classification methods for improving the diagnosis of Parkinson's disease," *Cogn. Syst. Res.*, vol. 54, pp. 90–99, May 2019.

[10] S. Lahmiri and A. Shmuel, "Detection of Parkinson's disease based on voice patterns ranking and optimized support vector machine," *Biomed. Signal Process. Control*, vol. 49, pp. 427–433, Mar. 2019.

[11] S. Lahmiri, D. A. Dawson, and A. Shmuel, "Performance of machine learning methods in diagnosing Parkinson's disease based on dysphonia measures," *Biomed. Eng. Lett.*, vol. 8, no. 1, pp. 29–39, 2018.

[12] H. Gürüler, "A novel diagnosis system for Parkinson's disease using complex-valued artificial neural network with k-means clustering feature weighting method," *Neural Comput. Appl.*, vol. 28, no. 7, pp. 1657–1666, 2017.

[13] Z. Cai, J. Gu, C. Wen, D. Zhao, C. Huang, H. Huang, C. Tong, J. Li, and H. Chen, "An intelligent Parkinson's disease diagnostic system based on a chaotic bacterial foraging optimization enhanced fuzzy KNN approach," *Comput. Math. Methods Med.*, vol. 2018, Jun. 2018, Art. no. 2396952.

[14] D. Gupta, A. Julka, S. Jain, T. Aggarwal, and A. Khanna, "Optimized cuttlefish algorithm for diagnosis of Parkinson's disease," *Cogn. Syst. Res.*, vol. 52, pp. 36–48, Dec. 2018.

[15] J. R. Orozco-Arroyave, F. Hönig, J. D. Arias-Londoño, and J. F. Vargas-Bonilla, "Automatic detection of Parkinson's disease in running speech spoken in three different languages," *J. Acoust. Soc. Amer.*, vol. 139, no. 1, pp. 481–500, 2016.

[16] J. Rusz, R. Cmejla, H. Ruzickova, and E. Ruzicka, "Quantitative acoustic measurements for characterization of speech and voice disorders in early untreated Parkinson's disease," *J. Acoust. Soc. Amer.*, vol. 129, no. 1, pp. 350–367, 2011.

[17] M. Little, P. McSharry, S. Roberts, D. Costello, and I. Moroz, "Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection," *Nature Precedings*, vol. 436, no. 5, p. 135, 2007.

[18] A. Tsanas, M. A. Little, P. E. McSharry, J. Spielman, and L. O. Ramig, "Novel speech signal processing algorithms for high-accuracy classification of Parkinson's disease," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 5, pp. 1264–1271, May 2012.

[19] M. Paul, S. C. Behr, C. Weiss, K. Heimann, T. Orlikowsky, and S. Leonhardt, "Spatio-temporal and -spectral feature maps in photoplethysmography imaging and infrared thermograph," *Biomed. Eng. OnLine*, vol. 20, no. 1, p. 8, 2021.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

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

[22] F. Demir, D. A. Abdullah, and A. Sengur, "A new deep CNN model for environmental sound classification," *IEEE Access*, vol. 8, pp. 66529–66537, 2020, doi: 10.1109/ACCESS.2020.2984903.

[23] F. Demir, V. Bajaj, M. C. Ince, S. Taran, and A. Şengür, "Surface EMG signals and deep transfer learning-based physical action classification," *Neural Comput. Appl.*, vol. 31, no. 12, pp. 8455–8462, Dec. 2019, doi: 10.1007/s00521-019-04553-7.

[24] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. 32nd Int. Conf. Mach. Learn. (ICML)*, vol. 1, 2015, pp. 448–456.

[25] F. Demir, M. Turkoglu, M. Aslan, and A. Sengur, "A new pyramidal concatenated CNN approach for environmental sound classification," *Appl. Acoust.*, vol. 170, Dec. 2020, Art. no. 107520, doi: 10.1016/j.apacoust.2020.107520.

[26] D. Şengür and S. Siuly, "Efficient approach for EEG-based emotion recognition," *Electron. Lett.*, vol. 56, no. 25, pp. 1361–1364, 2020, doi: 10.1049/el.2020.2685.

[27] F. Selimefendigil, Y. Akbulut, A. Sengur, and H. F. Oztop, "MHD conjugate natural convection in a porous cavity involving a curved conductive partition and estimations by using long short-term memory networks," *J. Thermal Anal. Calorimetry*, vol. 140, no. 3, pp. 1457–1468, May 2020.

[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[29] M. Nour, Z. Cömert, and K. Polat, "A novel medical diagnosis model for COVID-19 infection detection based on deep features and Bayesian optimization," *Appl. Soft Comput.*, vol. 97, Dec. 2020, Art. no. 106580, doi: 10.1016/j.asoc.2020.106580.

[30] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[31] A. S. Ashour, M. K. A. Nour, K. Polat, Y. Guo, W. Alsaggaf, and A. El-Attar, "A novel framework of two successive feature selection levels using weight-based procedure for voice-loss detection in Parkinson's disease," *IEEE Access*, vol. 8, pp. 76193–76203, 2020.

FATIH DEMIR received the B.Sc., M.Sc., and Ph.D. degrees in electrical and electronics engineering from Firat University, Turkey, in 2007, 2010, and 2020, respectively. His research interests include environmental sound classification, biomedical signals, biomedical images, and deep learning.

ABDULKADIR SENGUR received the B.Sc. degree in electronics and computers education, the M.Sc. degree in electronics education, and the Ph.D. degree in electrical and electronics engineering from Firat University, Turkey, in 1999, 2003, and 2006, respectively. He became a Research Assistant at the Technical Education Faculty, Firat University, in February 2001. He is currently a Professor with the Technology Faculty, Firat University. His research interests include signal processing, image segmentation, pattern recognition, medical image processing, and computer vision.

ALI ARI received the bachelor's degree in electronics and computers education, the bachelor's degree in computer engineering, and the master's degree in electronics and computers education from Firat University, Turkey, in 2010, 2013, and 2014, respectively, and the Ph.D. degree in computers engineering from Inonu University, Turkey, in 2019. He became a Lecturer with the Surgu Vocational School, Inonu University, in 2013. He is currently an Assistant Professor with the Computer Engineering Department, Inonu University. His research interests include image segmentation, pattern recognition, medical image processing, computer vision, deep learning, and signal processing.

KAMRAN SIDDIQUE (Senior Member, IEEE) received the Ph.D. degree in computer engineering from Dongguk University, South Korea. He is currently an Associate Professor with Xiamen University Malaysia. His research interests include big data, cybersecurity, machine learning, the IoT, and cloud computing.

MOHAMMED ALSWAITTI (Member, IEEE) received the B.Eng. degree in computer engineering from the Islamic University of Gaza (IUG), Palestine, in 2010, and the M.Sc. degree in electronic systems design engineering from Universiti Sains Malaysia (USM), Malaysia, in 2011. He worked as a Lecturer at the University of Palestine and an Instructor at the Educational Technology Center, IUG. He pursued his Ph.D. degree under the Malaysian International Scholarship (MIS) Scheme and worked as a Graduate Assistant at the Electrical and Electronic Engineering Department, USM. Currently, he is an Assistant Professor with the Electrical and Computer Engineering Department, XMUM. His research interests include nature-inspired optimization-based clustering techniques, machine learning, and pattern recognition.

• • •