Feature Mapping and Deep Long Short Term Memory Network based Efficient Approach for Parkinson’s disease Diagnosis

FATIH DEMIR¹, ABDULKADIR SENGUR², ALI ARI³, KAMRAN SIDDIQUE⁴, (SENIOR MEMBER, IEEE), AND MOHAMMED ALSWAITTI⁵, (MEMBER, IEEE)

¹Firat University, Vocational School of Technical Sciences, Biomedical Dept., Elazig, Turkey
²Firat University, Technology Faculty, Electrical-Electronics Engineering Dept., Elazig, Turkey
³Inonu University, Engineering Faculty, Computer Engineering Dept., Malatya, Turkey
⁴School of Electrical and Computer Engineering, Department of Information and Communication Technology, Xiamen University Malaysia, Sepang, 43900, Malaysia
⁵School of Electrical and Computer Engineering, Department of Information and Communication Technology, Xiamen University Malaysia, Sepang, 43900, Malaysia

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

This research was supported by Xiamen University Malaysia Research Fund (Grant No: XMUMRF/2019-C4/IECE/0012)

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].

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 et al. [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 present 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 et al. [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 “houseness” 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 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.

III. 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 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.
Des, the LSTM layer. Thus, Rectified Linear Unit (ReLU) was initially normalized into 0 to 255 ranges by using the Equation 1:

\[ x_i^{\text{normalized}} = 255 \times \frac{(x_i - \min(x_i))}{\max(x_i) - \min(x_i)} \]  

where \( x_i \) showed the \( i \)th 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 420x560x3 color image. The obtained images were then resized to 180x180x3 color images. Fig. 2 shows all steps in the pre-processing stage of the proposed.

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:

\[ x_i^{\text{normalized}} = 255 \times \frac{(x_i - \min(x_i))}{\max(x_i) - \min(x_i)} \]  

and learnable filter are used in the convolution operation process [20]–[22]. The learnable filter can be in different sizes such as 3x3 and 5x5, 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:

\[ (I \ast F)(i,j) = \sum_m \sum_n F(m,n)X(i - m, j - n) \]  

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:

\[ \hat{x}_i = \frac{x_i - b_m}{\sqrt{b_v} + \epsilon} \]  
\[ y_i = b\hat{x}_i + a \]  

Here \( \hat{x}_i \) is the normalized activation, and Constant \( \epsilon \) 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:

\[ f(x) = \max(0, x) \]

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 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:

\[
\text{ACC} = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}
\]

\[
Sn = \frac{TP}{TP + FN} \tag{13}
\]

\[
Sp = \frac{TN}{TN + FP} \tag{14}
\]

\[
Pr = \frac{TP}{TP + FP} \tag{15}
\]

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

### IV. 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.

\[
S^k = \frac{e^{x^k}}{\sum_{i=1}^{N} e^{x^i}} \tag{11}
\]
| Order | Layer               | Layer info                                      | Layer output size |
|-------|---------------------|-------------------------------------------------|-------------------|
| 1     | Sequence Input      | Sequence input with 180x180x3 dimensions        | 180x180x3         |
| 2     | Sequence Folding    | -                                               | 180x180x3         |
| 3     | Convolution         | Convolutions with 20 filters, 5x5x3, filter size, stride [1 1] and padding [0 0 0 0] | 176x176x20        |
| 4     | Batch Normalization | Batch normalization with 20 channels            | 176x176x20        |
| 5     | ReLU                | First ReLU (ReLU1)                              | 176x176x20        |
| 6     | Sequence Unfolding  | Cell data                                       | 176x176x20        |
| 7     | Flatten             | Cell data                                       | (619520x1)        |
| 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      | crossentropy with classes 'normal' and 'parkinson' | 1                 |

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 5x5x3. After the convolution layer, the input size became 176x176x20 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 61950x1. 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.

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.
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.

| 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   |

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.
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 (25x30)
3: Create feature mapping based-RGB images (420x560x3) from 2D data with the Colormap technique
4: Resize RGB images to the input layer size of deep LSTM (180x180x3)
5: Split dataset randomly at 70% training 30% test ratio (training :N×0.7 and test: N×0.3)
6: Train the proposed approach from scratch
7: Test the proposed approach with the trained model and test dataset

ACKNOWLEDGMENT

The authors acknowledge that this research was supported by Xiamen University Malaysia Research Fund (Grant No: XMUMRF/2019-C4/IECE/0012).

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. Little, P. McSharry, and L. Ramig, “Accurate telemonitoring of Parkinson’s disease progression by non-invasive speech tests,” Nat. Preced., p. 1, 2009.
[3] C. G. Goetz et al., “Testing objective measures of motor impairment in early Parkinson’s disease: Feasibility study of an at-home testing device,” Mov. Disord., vol. 24, no. 4, pp. 551–556, 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. Am. Dent. Assoc., vol. 140, no. 6, pp. 658–669, 2009.
[6] B. E. Sakar et al., “Collection and analysis of a Parkinson speech dataset with multiple types of sound recordings,” IEEE J. Biomed. Heal. Informatics, vol. 17, no. 4, pp. 828–834, 2013.
[7] C. O. Sakar et al., “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, 2019.
[8] C. R. Pereira, D. R. Pereira, S. A. T. 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, 2019.
[9] 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, 2019.
[10] S. Lahmiri 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. Gürel, “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 et al., “An intelligent Parkinson’s disease diagnostic system based on a chaotic bacterial foraging optimization enhanced fuzzy KNN approach,” Comput. Math. Methods Med., vol. 2018, 2018.

[14] D. Gupta et al., “Optimized cuttlefish algorithm for diagnosis of Parkinson’s disease,” Cogn. Syst. Res., vol. 52, pp. 36–48, 2018.

[15] J. R. Orozo-Arroyave et al., “Automatic detection of Parkinson’s disease in running speech spoken in three different languages,” J. Acoust. Soc. Am., 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. Am., 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,” Nat. Proced., p. 1, 2007.

[18] A. Tsalas, M. A. Little, P. E. McSharry, J. Spielman, and L. O. Ramljak, “Novel speech signal processing algorithms for high-accuracy classification of Parkinson’s disease,” IEEE Trans. Biomed. Eng., vol. 59, no. 5, pp. 1264–1271, 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. Online, vol. 20, no. 1, pp. 1–54, 2021.

[20] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv Prepr. arXiv1409.1556, 2014.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Advances in neural information processing systems, 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, 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,” 32nd Int. Conf. Mach. Learn. ICML 2015, vol. 1, pp. 448–456, 2015.

[25] F. Demir, M. Turkgolu, M. Aslan, and A. Sengur, “A new pyramidal concatenated CNN approach for environmental sound classification,” Appl. Acoust., vol. 170, p. 107520, 2020, doi: 10.1016/j.apacoust.2020.107520.

[26] D. Şengür and S. Sulyqi, “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. Selimfendigil, Y. Akbulut, A. Sengur, and H. F. Ortop, “MHD conjugate natural convection in a porous cavity involving a curved conductive partition and estimations by using Long Short-Term Memory Networks,” J. Therm. Anal. Calorim., vol. 140, no. 3, pp. 1457–1468, 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. Çömert, 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, pp. 106580, 2020, 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. Alsagaf, 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.