Visual Representation of Online Handwriting Time Series for Deep Learning Parkinson's Disease Detection

Catherine Taleb, Maha Khachab, and Chafic Mokbel
University of Balamand
Balamand El-Koura, Lebanon
catherine.taleb@std.balamand.edu.lb
Maha.Khachab@balamand.edu.lb
chafic.mokbel@balamand.edu.lb

Laurence Likforman-Sulem
LTCI/Telecom Paris/Institut polytechnique de Paris
Paris, France
likforman@telecom-paristech.fr

Abstract- Parkinson's disease (PD) is a neurological disorder associated with a progressive decline in motor skills, speech, and cognitive processes. Since the diagnosis of Parkinson's disease is difficult, researchers have worked to develop a support tool based on algorithms to separate healthy controls from PD patients. Online handwriting dynamic signals can provide more detailed and complex information for PD detection task. Existing techniques often depended on hand-crafted features that required expert knowledge of the field. In this paper, it is suggested to learn pen-based features by means of deep learning for automatic classification of PD. For this purpose, a visual representation of the time series can be computed and used at the input of a convolutional neural network (CNN) as in [4]. Classically, the time series is transformed into a fixed dimension image applying normalization on the time dimension. In this work we have experimented several visual representations, including the spectrogram where normalization of the time scale is applied after short term information has been extracted locally. We have been able to show that considering the local short term information allows the deep learning models to provide better classification results compared to a globally normalized fixed dimension visual representation. For validation purpose, a CNN-BLSTM was directly applied on the time series, without any normalization of the time scale which led to best performance equivalent to the one obtained on spectrogram representation.

Keywords—PDMultiMC dataset, Parkinson's Disease, CNN, CNN-BLSTM, Spectrogram images, Gramian Angular Field images.

I. INTRODUCTION

Parkinson's disease (PD) is a neurodegenerative disorder caused by a decreased dopamine level in the brain [1]. Diagnosis of PD is difficult especially in early stages when motor symptoms (hypokinesia, tremor, micrographia and muscle rigidity) are not yet severe [2]. Some neurological testing beside brain scans are usually applied in PD diagnosis [1]. However, these methods need a high level of professional skills [1]. Hence, the need to define a reliable system that assists in the decision-making process leading to the diagnosis of PD. We have presented in our previous work the PDMultiMC multimodal Parkinson's disease database, that consists of online handwriting samples collected from 21 PD patients and 21 healthy controls using the digitizing tablet Wacom Intuos 5 [3]. The tablet pen is composed of sensors that measure information captured during handwritten exams. These measurements include spatial displacement (x, y, and z positions), pen pressure, time stamp, and pen-tip angle (altitude, azimuth) [3]. A SVM model trained on pre-engineered features was built in [3]. Based on this model, we have found that handwriting can be a tool for PD diagnosis with a 97% prediction performance when features related to the correlation between kinematic and pressure are used [3]. However, hand-crafted features model required expert knowledge of the field; which motivate us to learn pen-based features by means of deep learning [5]. We have also noticed that only a few research studies uses deep learning to learn features from online handwritten exams, taken from HandPD database, for automatic PD identification [4]. In [4] the online handwriting signals, considered as time series, are first transformed into an image and then a deep learning model has been used to perform the PD detection. In the present work, several approaches to represent the handwriting signals as images are explored and compared. Thus, the main contribution of the present work is to build an efficient model for PD detection via visual representation of online handwriting and deep learning.

When dealing with online handwriting time series, the variation over the time axis defines a challenge for models requiring a fixed dimension input. One approach would consists in normalizing the time series leading to a fixed dimension visual representation as suggested in [4]. An alternative approach explicitly considers the time dimension, especially that the variation over this dimension is nonlinear. This paper proposes two deep learning classes of models in order to tackle the time variation in time series classification. In the first one, spectrogram visualization is performed by extracting the local short information before normalization onto a fixed size image using Lanczos technique [11] and classification. The second one integrates the time variation within a Long-Short-Term Memory Networks model which is directly applied on the time series. The paper is built as follows. A description of the dataset is presented in Section II. Section III provides a description of the deep learning models studied. Numerical results and the discussion are presented in Section IV, and the conclusion and future work are in Section V.

978-1-7281-5054-3/19/$31.00 ©2019 IEEE
DOI 10.1109/ICDARW.2019.50111

II. DATABASE

PDMultiMC is a multimodal database that includes handwritings, speech samples, and eye movements recordings collected from PD patients in two phases (medication on and medication off) and from control subjects [3]. In this work, we focus on handwriting analysis. The seven handwriting tasks recorded for each of the 21 controls and 21 PD patients in their “on-state” were studied and analyzed. A more detailed description of the dataset is provided in [3]. A complete task sheet is shown in Fig. 1; where the repetitive cursive letter (letter ℓ), the triangular wave, the rectangular wave, the repetitive “Monday”, the repetitive “Tuesday”, the repetitive subject’s name, and the repetitive subject’s last name represent the seven tasks *respectively*. The first 3 tasks demand continuous pen movement; which emphasis hypokinesia and tremor [15]. In addition, micrographia needs long writing task to manifest, that’s why we chose the words repetition in the other 4 tasks [15]. Since each sensor outputs the whole signal acquired during the handwriting task, we can represent such data as a Time series, as depicted in Fig. 2 that represents the output of task1 from a healthy subject and a PD patient. The drawings are somehow close, but the signals extracted from PD patient are noisier than those of the control subject. Here comes the importance of studying the extracted signals instead of the drawings [4].

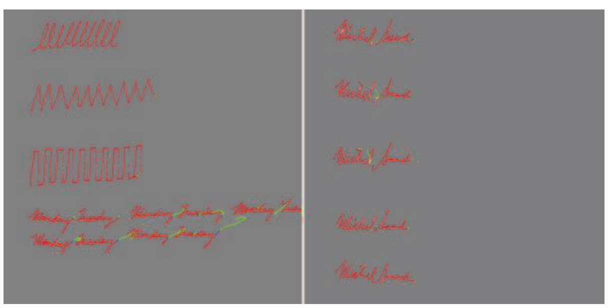

Figure 1. Template used to assess the handwritten skills of a given individual.

III. DEEP LEARNING FOR TIME-SERIES CLASSIFICATION

In recent years, convolutional neural networks have shown excellent performance on image classification tasks [5]. In order to benefit from this, some researchers proposed to transform a Time series into an image and use it as an input to CNN, which will be able to learn features that are used to distinguish healthy individuals from PD patients [4]. From the other side, Long-Short-Term Memory Networks (LSTM) is a family of neural networks that excels in learning from sequential data and can cope with variable length time series [6]. LSTMs are quite popular in dealing with text based data, and have been quite successful in language translation and text generation [7]. Since LSTMs can store information during long time intervals, they are thus also appropriate for processing time series representing handwriting signals [7].

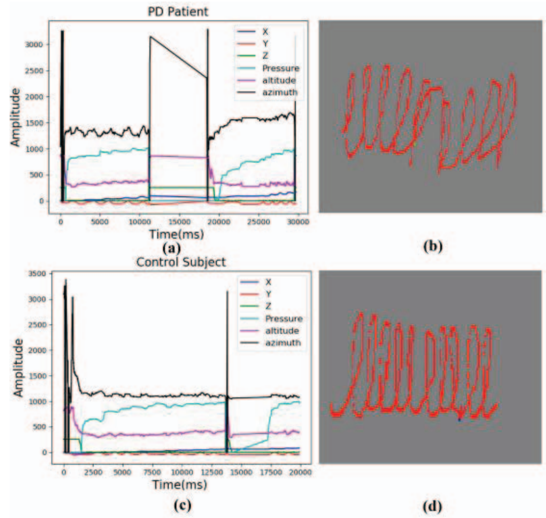

Figure 2. Cursive repetitive ‘ℓ’ samples from: (b) PD patient and (d) control, and their respective signals recorded by the pen in (a) and (c).

Several approaches were used to encode the raw time series into images before being fed into the CNN. In this work, a comparison between CNN-BLSTM and CNN models is done; where each CNN model differs from the others by the approach used to convert time series into image.

A. Pre-processing

As mentioned in [3], PDMultiMC database includes Arabic, French, and English samples. In order to have the same writing direction, the X coordinates of the Arabic samples are flipped. After that, the X and Y coordinates are normalized to achieve a uniform range across all subjects. Each X and Y in the input dataset is normalized by subtracting the minimum and the mean, *respectively*. In addition, for the CNN model, all images are normalized to the range (0, 1) to achieve a uniform contrast and intensity range. The same procedure is applied to the LSTM model; where all time series are also normalized to (0, 1).

B. Encoding time series as images

Each handwriting task is composed of n rows (exam time in ms) and 7 columns (stand for X, Y, Z, pressure, altitude, azimuth, and time stamp) [4]. In order to get the best signals combination, the number of time series features used is a hyper-parameter varying between 1 and 7 (total number of extracted signals) and defined by k. Three frameworks for encoding time series as images were used; where one framework transforms the whole data (n×k matrix) into one image, and the two others convert each feature signal into a separate image. In the 3 frameworks, the converted image size depends on n; when the length of time series is n. The size of time series n differs from one person to another, because a person may take longer than another to perform the exam [4]. To keep the number of input feature maps identical for the CNN, images sizes should be same for all the subjects. In this study, all images are grayscale of size 64×64.

1) Time series-based images: This approach was proposed in [4], it transforms the whole data (n×k matrix) into one image by concatenating the n rows into one vector and then reshaping it into a square matrix. This squared matrix is resized to 64×64 pixels resolution using Lanczos resampling method [11]. The time series-based image of a patient and a healthy subject concerning the 7 tasks are shown in Fig. 3. It is clear that there is a difference between the control subject and the PD patient, and a difference between tasks; which means that each task is important to capture distinct information.

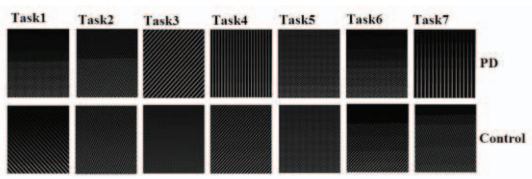

Figure 3. The Time series-based images of a PD patient (first row) and a control subject (second row) concerning the 7 tasks.

2) Gramian angular field: Gramian Angular Field is a framework used to encode each feature time series as image. The purpose here is to represent the time series in a polar coordinate system instead of the Cartesian coordinates [8]. Given a time series X={x_1, x_2, …..,x_n} of n observations, X is rescaled to $\tilde{X}$ so all the values fall in the interval [-1,1] then transformed to polar coordinate using these equations:

$$\varphi = \arccos(\tilde{x}_i), -1 \leq \tilde{x}_i \leq 1, \tilde{x}_i \in \tilde{X} \quad (1)$$

$$r = \frac{t_i}{C}, t_i \in \mathbb{N} \quad (2)$$

Where t_i is the time stamp and C is a constant factor used to regularize the span of the polar coordinate system [8]. After transforming the rescaled time series into the polar coordinate system, the angular perspective is exploiting by considering the trigonometric sum/difference between each point to identify the temporal correlation within different time intervals [8]. The Gramian Summation Angular Field (GASF) and Gramian Difference Angular Field (GADF) are defined as follows:

$$GASF_{i,j} = \left[\cos(\varphi_i + \varphi_j)\right] \quad (3)$$

$$GADF_{i,j} = \left[\sin(\varphi_i - \varphi_j)\right] \quad (4)$$

The size of Gramian matrix is n×n. In order to get the same image size (64×64) for all samples, Piecewise Aggregation Approximation (PAA) is used to smooth the time series while keeping trends by dividing the time series of length n into 64 equi-sized "frames" [8]. The mean value of the time series falling within a frame is calculated and a vector of these values becomes the data reduced representation [8]. GADF and GASF are applied to each time series separately; this means that for each exam we will get 7 different images . GADF and GASF of a PD patient and a control subject concerning task1 for each of the 7 signals are shown in Fig. 4. In this study, we have focused on the GADF images.

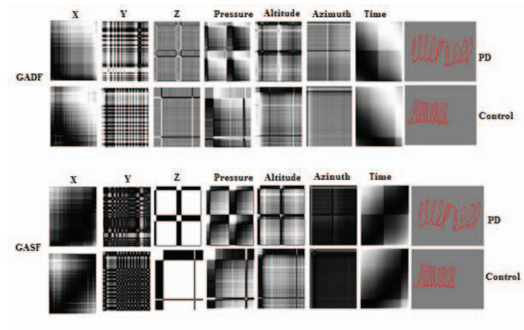

Figure 4. GADF and GASF of a PD patient and a control subject in task1 for each of the 7 signals.

The idea of PAA is to replace each segment by its mean. However, this method can remove by smoothing some important information that can play a role in PD detection. In this study each time series is divided into M segments of equal length 64; where each segment is converted into image using GAF method. The number of segments M depends on the original time series length; which means that for each subject we will get different number of images.

3) Spectrogram images: Handwritten dynamics signals are considered as nonstationary signals that are difficult to separately analyze in time or in frequency domain. Time-frequency representations method is specified to analyze such signals [9]. The Short Time Fourier Transform (STFT) is another framework used for encoding each feature time series as image. The time-frequency resolution of the spectrogram is dependent on the window size and type [12]. Different window sizes and types were tried, and the ones returning the best spectrogram resolution were selected. A blackman windowing function was used where both the window length and the number of FFT point were 256 and the overlapping rate was 50%.

In line with the idea of treating a spectrogram like an image, the number of frequencies and the number of time bins in the spectrogram representation refer to the height and the width of the output image in pixels. In addition, the numerical "brightness" value of each pixel of this two dimensional image is then equal to the output value of the spectrogram at the particular time and frequency corresponding to that pixel [10]. These values should be converted to a logarithmic scale (decibels) then normalized to [0, 1] generating a grayscale image [11]. The width of the image depends on the length of the signal. To keep the number of input feature maps identical, the area of spectrogram should be the same for all subjects. While we believe that the variation over the time axis is non-linear, we assume here that this variation is linear and we apply Lanczos technique to resize the spectrogram images to 64×64 pixels resolution [11]. The spectrograms of a patient and a control concerning task1 for each of the 7 signals are shown in Fig. 5.

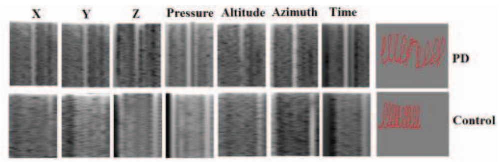

Figure 5. The spectrograms of a PD patient and a control subject in task1 for each of the 7 signals.

C. Deep learning architectures

Deep learning approaches have been successfully used in computer vision. It is therefore tempting to represent the time series as images and apply these networks for the PD detection. The image presented at the input of a CNN undergoes several convolutional layers that would extract useful information to be finally presented to one or multiple classification layer at the output stage. In the following the different architectures are described as well as the CNN-BLSTM used for detection from direct time series.

1) CNN architecture for a single input image: The CNN model architecture used with a single image input (time series-based image) is represented in Fig. 6. The overall architecture consists of 2 main parts, the feature extractor and the classifier. The feature extractor layers consist of 2 convolution layers (Relu unit) and 2 pooling layers. Starting with a 64×64 pixel image with one channel (Grayscale), all the convolutional layers employ kernels of size 5×5 with stride of 1 pixel, and all the pooling layers are 2×2 max pooling. The convolutional layers convert the image to 64 feature maps of size 16×16. After using convolution layers to extract the spatial features of an image, we apply fully connected layers for the final classification. The output of the convolution layers is flattened, then a hidden layer is used before performing the final classification.

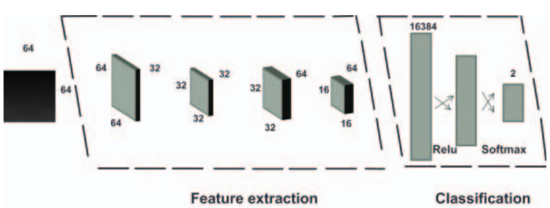

Figure 6. Convolutional neural network model architecture used in case of a single image input.

2) k-input CNN (1≤ k ≤ 7) architecture: As stated in Section III-B, GAF and spectrogram frameworks encode each feature time series into separate image. This means that for each subject we have multi-image as inputs to the CNN model. The k-input CNN model is summarized in Fig. 7; where different convolution (Relu unit) and pooling operations are applied on each image in order to learn different features independently. At a later stage, the outputs are combined and flattened and then fed into densely connected layer to make a prediction. The number of images is a hyper-parameter; where k varies between 1 and 7.

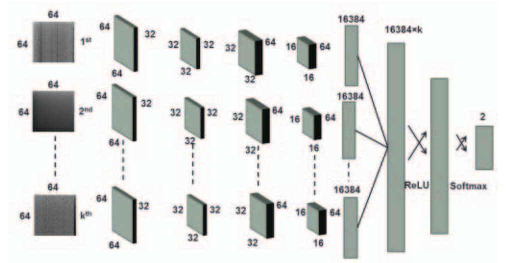

Figure 7. Architecture of the k-input CNN model. The k inputs come from distinct features.

3) Slicing and combination approaches: As described in Section III-B-2, each subject has a different number of GAF images per task. When doing training using the k-input CNN model represented in Fig. 7, all the training window slices images are considered independent training instances. Window slicing is also applied when predicting the label of a testing time series. The trained k-input CNN model predicts the label of each of the window slices. To make the final prediction for each subject in the test set, 2 operations were used:

- Using a majority vote among all these slices
- Using bidirectional-LSTM summarized in Fig. 8; where the M probability vectors outputs of the CNN models are considered as a Multivariate sequence (which have two or more variables observed at each time) of length M, and are used as input to a dynamic BLSTM to decide the final prediction.

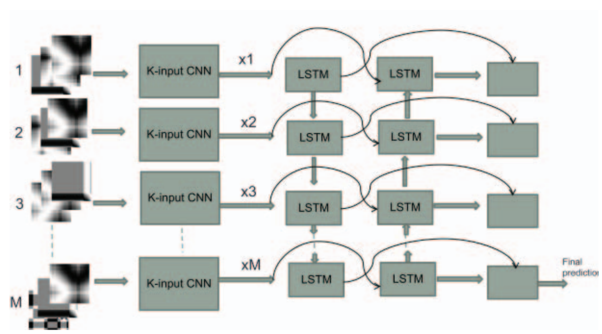

Figure 8. Slicing combination approach. k time series are cut into M slices. In each slice, k input images are built. The decisions (probability distribution for each class) provided by each slice images are input as a sequence of length M to a BLSTM.

4) k-input CNN-BLSTM (1≤k≤7) architecture: For the sake of comparison, we also suggest to apply a CNN-BLSTM directly on the time series without visualizing them as images. The CNN-BLSTM architecture involves using CNN layers for feature extraction on input data combined with BLSTMs to support sequence prediction. Instead of converting the time series into images, the entire raw time series are used here as input to the model. The convolutional layers are constructed using one-dimensional kernels that move through the sequence. A CNN- BLSTM model on multivariate time series is represented in Fig. 9. The output of the CNN is a sequence of length n/4 of vectors of size 32; where n represents the time series length. This sequence is then used as input to a dynamic BLSTM. The number of time series k is a hyper-parameter, and it varies between 1 and 7.

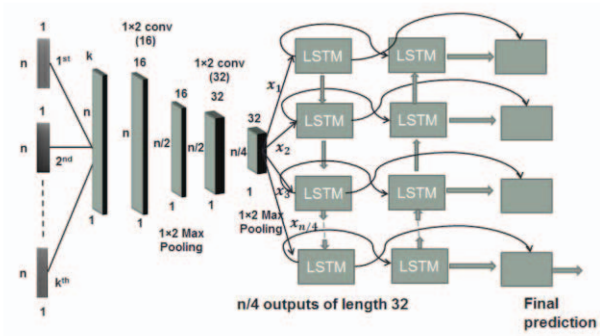

Figure 9. A CNN-BLSTM model on multivariate time series.

D. *Combination approach*

The experiments are divided into two rounds: a single assessment and a combined assessment [4]. In the single assessment, we analyze each task separately, while in the combined assessment we combine the output of each CNN/CNN-BLSTM model using majority voting in order to obtain the final result as shown in Fig. 10. The hyper-parameter k described in Section III is selected in a way of returning the highest overall accuracy.

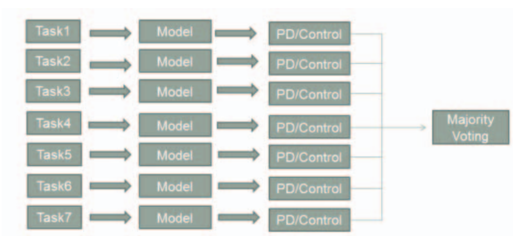

Figure 10. Combination approach to evaluate CNNs/CNN-biLSTMs in the context of PD identification.

IV. Results and Discussions

The performance was figured out in term of accuracy, sensitivity, and specificity. The models described in Section III are tested; where 3-folds cross validation was applied with stratified sampling method in order to insure the same class distribution in all the subsamples. The performance measures in TABLE I represent the average of 3 runs considering the maximum voting combination approach, where the best results are in bold.

According to the results presented in TABLE I, we can see that both the CNN model with spectrogram images as input, and the CNN-BLSTM model with raw time series as input return the best PD detection accuracy; where best features combination was chosen in a way of returning best overall accuracy. Based on these features, it's important to compare and rank the performance of each task separately to select tasks with the highest features relevance. Task-wise system accuracies for different model are represented in TABLE II; where D1, D2, D3, D4, and D5 models refer to the ones used in entries 1, 2, 3, 4 and 5 in TABLE I. It can be observed from that "all tasks" reports highest accuracies across all the 5 models. Additionally, we can observe that Task 2 (triangular wave), and Task 3 (rectangular wave) report highest accuracies across all the 5 models. These two tasks are considered long and somehow complex. Copying these cursive tasks needs higher cognitive force and explains the effect of disease on handwriting [3].

The main finding from the direct comparison of our CNN-BLSTM and 2D spectrogram CNN models against the model proposed by Pereira et al. [4] (see Section III-B-1) is that the CNN classifier fed with 2D spectrograms, and the CNN-BLSTM fed with time series perform better than a single CNN fed with time series-based images as in [4]. This finding can be explained by the fact that the signals generated by the pen are multi- frequency, non-periodic, and arbitrary [13]. In addition, the BLSTM-based model takes advantage of learning the temporal feature activation dynamics, which the CNN model is not capable to model [14].

V. Conclusion and Future Work

The main contribution of this paper is to employ deep learning approach to aid in PD detection. We proposed 2 based learning models for end to-end time series classification: the CNN and the CNN-BLSTM. For the CNN model, two different frameworks were proposed to encode time series into images: Gramian angular field images, and spectrogram images. We compared these 2 frameworks with the one proposed by Pereira et al. [4]: the direct encoding of time series into images. The advantage of using spectrogram images consists in computing local short term information that exists in the non-stationary online handwriting signals before normalization, while the other two approaches normalize the time series into a fixed dimension image without extracting local information. For the CNN-BLSTM model, the raw time series are directly used with no need to convert them into images. This approach has been experimented to validate the importance of considering the local information before integrating on the time scale. We demonstrated the importance of both: a deep architecture based on the combination of 1D CNN and BLSTM recurrent layers, and a CNN model with spectrograms as input in PD detection. Our results clearly show that when explicitly considering the local short term information on the time axis of the non-stationary online handwriting signals the deep learning models provides the best performance.

Compared with these deep learning models, the SVM model trained on pre-engineering features and described in [3] shows better accuracy because SVM is not sensitive to the number of training samples, in contrast to deep learning models which require a large number of training samples to work well.

TABLE I. 3-FOLD CV PERFORMANCE MEASURES CONSIDERING THE MAXIMUM VOTING COMBINATION APPROACH

Exp.	Model	Data Input	Window slicing combination approach	Perf. (%)	Best Features Combination
1.	Single input CNN	Time series-based images		Acc :80.95 Sens:85.71 Spec:76.19	Pressure
2.	k-input CNN	GADF images using window slicing	BLSTM	Acc :80.95 Sens:71.43 Spec:90.48	X+Y+Z+Pressure+Altitude+Azimuth
3	k-input CNN	GADF images using window slicing	Max. Voting	Acc :78.57 Sens:80.95 Spec:76.19	X+Y+Z+Pressure
4.	k-input CNN	Spectrogram images		**Acc :83.33** Sens:85.71 Spec:80.95	X+Y+Z+Pressure+Altitude
5.	k-input CNN-BLSTM	Raw time series		**Acc :83.33** Sens:71.43 Spec:95.24	X+Y+Z+Pressure+Altitude+Azimuth

TABLE II. TASK-WISE SYSTEM ACCURACIES FOR DIFFERENT MODELS (IN %)

Task	D1	D2	D3	D4	D5
Repetitive cursive letter ‘l’	**71.43**	64.29	61.90	54.76	57.14
Triangular wave	69.05	69.05	64.29	50.00	**76.19**
Rectangular wave	33.33	**76.19**	73.81	64.29	73.81
Repetitive “Monday”	61.90	54.76	61.90	61.90	**64.29**
Repetitive “Tuesday”	50.00	59.52	**66.67**	64.29	45.24
Repetitive “Name”	**69.05**	54.76	35.71	64.29	47.62
Repetitive “Family Name”	66.67	61.90	57.14	71.43	**73.81**
All tasks	80.95	80.95	78.57	**83.33**	**83.33**

This means that PD classification using deep learning is a challenging task due to the limited data availability. As future work, we will investigate transfer learning and data augmentation approaches based on these models to perform PD detection on large-scale data.

ACKNOWLEDGMENT

This study has been approved by the institutional review board (IRB) of the University of Balamand and Saint George Hospital University Medical Center.

REFERENCES

[1] R. K. Sharma and A. K. Gupta, "Voice analysis for telediagnosis of Parkinson disease using artificial neural networks and support vector machines," International Journal of Intelligent Systems and Applications, vol. 7, no. 6, pp. 41–47, May 2015, doi:10.5815/ijisa.2015.06.04.

[2] M. Nilashi, O. Ibrahim and A. Ahani, “Accuracy Improvement for Predicting Parkinson’s Disease Progression,” *Scientific Reports*, vol. 6, no. 1, 2016, doi:10.1038/srep34181.

[3] C. Taleb, L. Likforman, M. Khachab and C. Mokbel, "Feature Selection for an Improved Parkinson’s Disease Identification Based on Handwriting", in Arabic Script Analysis and Recognition (ASAR), 2017 1st International Workshop on, Nancy, France, doi:10.1109/ASAR.2017.8067759.

[4] C. R. Pereira, D. R. Pereira, G. H. Rosa, V. H. Albuquerque, S. A. Weber, C. Hook, and J. P. Papa, “Handwritten dynamics assessment through convolutional neural networks: An application to Parkinson’s disease identification,” *Artificial Intelligence in Medicine*, vol. 87, pp. 67–77, 2018, doi:10.1016/j.artmed.2018.04.001.

[5] Gamboa and J. C. Borges, “Deep Learning for Time-Series Analysis,” Kaiserslauter Univ., Kaiserslautern, Germany ,2017.

[6] R. Atienza, “LSTM by Example using Tensorflow,” *Towards Data Science*, 17-Mar-2017. [Online]. Available: https://towardsdatascience.com/lstm-by-example-using-tensorflow-feb0c1968537.

[7] B. Himmetoglu, “Time series classification with Tensorflow,” 19-Sep-2017. [Online]. Available: https://burakhimmetoglu.com/2017/08/22/time-series-classification-with-tensorflow/.

[8] Z. Wang, T. Oates, “Imaging time-series to improve classification and imputation,” in proceedings of the 24th International Join Conference on Artificial Intelligence(IJCAI), 2015.

[9] N. A. Khan, M. N. Jafri, and S. A. Qazi, “Improved resolution short time Fourier transform,” *2011 7th International Conference on Emerging Technologies*, doi:10.1109/ICET.2011.6048476.

[10] E. Naumov, “A convolutional network on EEG spectrograms for sleep staging,” M.S.thesis, College of CS, McGill Univ., Montreal, 2017.

[11] M. Huzaifah, “Comparison of time-frequency representations for environmental sound classification using convolutional neural networks,” Jun. 2017, [online] Available: https://arxiv.org/abs/1706.07156.

[12] Nisar, S., Khan, O. and Tariq, M. (2016). An Efficient Adaptive Window Size Selection Method for Improving Spectrogram Visualization. Computational Intelligence and Neuroscience, pp.1-13, doi:10.1155/2016/6172453.

[13] M. A. Alsheikh, A. Selim, D. Niyato, L. Dayle, S. Lin, H-P. Tan, “Deep Activity Recognition Models with Triaxial Accelerometers” , in proceedings of the AAAI Workshop: Artificial Intelligence Applied to Assistive Technologies and Smart Environments, 2016.

[14] F. Ordóñez and D. Roggen, “Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition,” *Sensors*, vol. 16, no. 1, p. 115, 2016, doi:10.3390/s16010115.

[15] J. Alty, J. Cosgrove, D. Thorpe, and P. Kempster, “How to use pen and paper tasks to aid tremor diagnosis in the clinic,” *Practical Neurology*, vol. 17, no. 6, pp. 456–463, 2017, doi: 10.1136/practneurol-2017-001719.