

# A Light-weight CNN Model for Efficient Parkinson's Disease Diagnostics

Xuechao Wang\*, Junqing Huang<sup>†</sup>, Marianna Chatzakou<sup>‡</sup>,  
 Kadri Medijainen<sup>§</sup>, Pille Taba<sup>¶</sup>, Aaro Toomela<sup>||</sup>,  
 Sven Nömm<sup>\*\*</sup>, Michael Ruzhansky<sup>††</sup>

\* † ‡ †† Department of Mathematics: Analysis, Logic and Discrete Mathematics, Ghent University, Ghent, Belgium

Email: \*xuechao.wang@ugent.be, †junqing.huang@ugent.be, ‡marianna.chatzakou@ugent.be, ††michael.ruzhansky@ugent.be,

†† School of Mathematical Sciences, Queen Mary University of London, Mile End Road, London E1 4NS, United Kingdom

\*\* Department of Software Science, Faculty of Information Technology, Tallinn University of Technology,  
 Akadeemia tee 15 a, 12618, Tallinn, Estonia

Email: \*\*sven.nomm@ttu.ee

§ Institute of Sport Sciences and Physiotherapy, University of Tartu, Puusepa 8, Tartu 51014, Estonia  
 E-mail: § kadri.medijainen@ut.ee

¶ Department of Neurology and Neurosurgery, University of Tartu, Puusepa 8, Tartu 51014, Estonia  
 ¶ Neurology Clinic, Tartu University Hospital, Puusepa 8, Tartu 51014, Estonia

E-mail: ¶ pille.tabo@klinikum.ee

|| School of Natural Sciences and Health, Tallinn University, Narva mnt. 25, 10120, Tallinn, Estonia  
 E-mail: || aaro.toomela@tlu.ee

**Abstract**—In recent years, deep learning methods have achieved great success in various fields due to their strong performance in practical applications. In this paper, we present a light-weight neural network for Parkinson's disease diagnostics, in which a series of hand-drawn data are collected to distinguish Parkinson's disease patients from healthy control subjects. The proposed model consists of a convolution neural network (CNN) cascading to long-short-term memory (LSTM) to adapt the characteristics of collected time-series signals. To make full use of their advantages, a multilayered LSTM model is firstly used to enrich features which are then concatenated with raw data and fed into a shallow one-dimensional (1D) CNN model for efficient classification. Experimental results show that the proposed model achieves a high-quality diagnostic result over multiple evaluation metrics with much fewer parameters and operations, outperforming conventional methods such as support vector machine (SVM), random forest (RF), lightgbm (LGB) and CNN-based methods.

**Index Terms**—Parkinson's disease, Deep Learning, Hand-drawn tests

## I. INTRODUCTION

Parkinson's disease (PD) is a neurodegenerative disease that is the second most common neurological disorder after Alzheimer's disease [3]. From 1990 to 2016, the estimated global population affected by PD increased more than double (from 2.5 million to 6.1 million), affecting 1-2 in every 1,000 people over 60 years old [5]. PD symptoms such as bradykinesia, tremors, and rigidity can severely affect the patient's quality of life [14], family relationships, and social functioning, placing a heavy financial burden on individuals and society. Although PD is currently incurable, early diagnosis and proper treatment can allow relieving most of the symp-

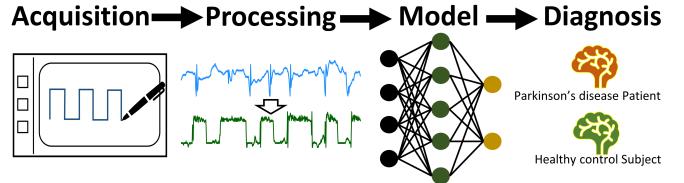


Fig. 1. The proposed workflow for Parkinson's disease diagnostics.

toms. Although some current clinical methods (e.g., optical coherence tomography, magnetic resonance imaging) can help diagnose early PD, they are too harsh and expensive to use, which can lead to patients missing the optimal treatment time. Therefore, researchers from different fields aim to combine knowledge to help people with limited access to medical care have an efficient diagnosis of PD.

Hand drawing tests have been used for almost a century as a convenient way to diagnose PD [2], [9], [21]. In fact, hand drawing is a complex activity that involves sensorimotor and cognitive components, and changes in it can be considered a promising biomarker for the evaluation of early PD [6], [20]. Originally, hand-drawing tests were performed on paper and pen and analysed by the practitioner. This classical setting has two drawbacks, the first is the subjective component introduced by the human practitioner, and the second is that the naked eye could not capture all the parameters of the movements. The work of [15] that proposed the use of a digital tablet opened a new research direction with the aim of addressing both drawbacks. With the proliferation of digital

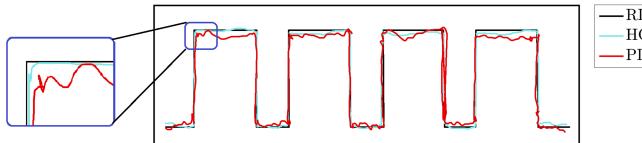


Fig. 2. Comparison of the hand drawings of Parkinson’s disease (PD) patient and healthy control (HC) subject under a reference drawing (RD).

devices, it is now possible to record measurement sequences for hand-drawing tests, provided by tablet and pen devices. The machine can capture the parameters invisible to the human eye, and the digital description of the drawing tests is free of subjective components. Initially, the number of parameters analysed was five, while in [7] and [23] the authors increased this number to hundreds. Furthermore, standard procedures were also described in [22]. Evidently, the development of hand-drawing-based decision is a noninvasive, real-time, and low-cost solution to support standard clinical assessments by human experts.

Deep learning (DL) based approaches have been gaining great attention and popularity in the analysis of drawing and writing tests. The research reported in [19], introduced NewHandPD, a data set of signals extracted from an electronic smart pen, and proposed to pose the problem of distinguishing PD from healthy control (HC) as an image recognition task through CNN. The potential of handwritten visual properties to predict PD was investigated by [16]. In [17], the drawn curves were enhanced with kinematic and pressure characteristics to classify the generated images using CNN. [4] developed a deep RNN model to learn patient similarity in PD that dynamically matched the temporal patterns of patient sequences to learn patient similarity directly from longitudinal and multimodal patient records. RNN with hyperbolic secant in the gates was investigated in [8] to address the same problem. Although the above models could generally diagnose PD, and the classification performance of diagnostic models has been shown to be comparable to that of human practitioners [24], the key step is to select appropriate characteristics through complex processing, and the overall structure of the network is relatively complex.

To address the shortcomings of the above methods, a simple and efficient deep learning method was proposed for the diagnosis of PD. As shown in Fig.1, subtle tremor signals were captured from digital Luria alternating series tests (dLAST) collected by the devices to directly reflect the degree of tremor at adjacent moments when performing fine movements. Meanwhile, a lightweight neural network combining LSTM and CNN, called the LSTM-CNN model, was proposed to learn the temporal and spatial characteristics of each tremor signal and thus effectively diagnose early PD. Furthermore, dLAST could not be obtained in the quantity necessary for deep neural network training, and the application of data segmentation technology alleviated the situation of small data sets to some extent. Our extensive experimental results demon-

strate that even with much fewer parameters and operations, our method achieves more than 90% on multiple evaluation metrics, delivering strong performance on par with or even better than the state-of-the-art.

The paper is organised as follows. Section II provides the reader with the necessary information about the data. Section III presents the research methods and the experimental setting. Section IV presents the main results of the current studies. Discussion of the results achieved, the limitations of the proposed methods, and possible future directions constitute Section V.

## II. MATERIAL

In this research, a data set is considered. The data set, here and later referred to as DraWritePD, was acquired from 49 participants, with a mean age of 74.1 years and a similar gender distribution. Within the group of patients with PD, the age deviation was approximately 3.35 years, while within the group of subjects with HC, the age deviation was 4.55 years, making both groups very similar.

Data acquisition was performed with an iPad Pro 9.7 inch (2016) equipped with an Apple Pencil. As shown in Fig.2, participants were asked to mimic the reference pattern to draw. During this process, the iPad Pro scanned the Apple Pencil signal at 240 points per second. As shown in Table I, for each scan, the device captures six time sequence parameters: azimuth ( $a$ ); altitude ( $l$ ); pressure ( $p$ ); timestamp ( $t$ ); x-Axis ( $x$ ); y-Axis ( $y$ ).

TABLE I  
DESCRIPTION OF THE DRAWRITEPD DATASET

Parameter	azimuth	altitude	pressure	timestamp	x-Axis	y-Axis
Notation	$a$	$l$	$p$	$t$	$x$	$y$
Quantitation	rad	rad	psi	sec	mm	mm

Originally, the dLAST consisted of 3 tests: ΠΑ, Π and *sin wave*, each with 3 exercises: *continue*, *copy* and *trace* applied to each test. The general task was to complete a set of tests consisting of 9 different drawing exercises. To optimise the testing procedure within the frameworks of the present contribution, only the Π test was considered. Specifically, the graphical plots of Π test are demonstrated in Fig. 2.

The data acquisition process was carried out under strict privacy law guidance. The Research Ethics Committee approved the study of the University of Tartu (No.1275T – 9).

## III. METHODOLOGY

In this section, we illustrate the proposed LSTM-CNN model for the diagnosis of PD. The framework starts with preprocessing methods for standardising the data. Regarding the model architecture, we propose a lightweight hybrid model that is composed of an LSTM block cascaded with a CNN-based classifier. We will demonstrate in detail the performance and efficiency of the proposed LSTM-CNN model.

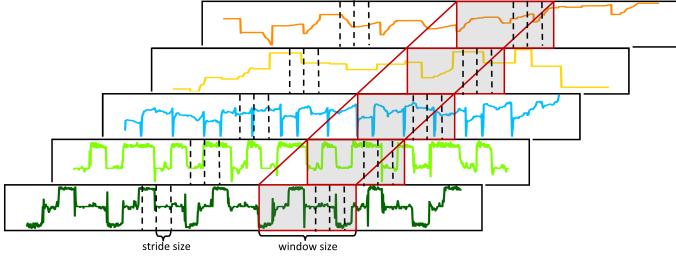


Fig. 3. The scheme of data segmentation with temporal overlap region used in the framework. The *window size* and *stride size* control the length and overlap of the patch.

#### A. Data pre-processing

In many practical applications, the raw data may not be perfectly collected — for example, some data may be missing, or inevitably contain some abnormal features that are not appropriate for direct use in training a regression or a classification model. In this situation, it is always necessary to use some pre-processing methods to standardise the data or improve their quality. Considering the characteristics of the DraWritePD data set, we employ the following preprocessing methods to ensure that our data are suitable for training a diagnostic model.

**Normalisation** As explained in Section II, the DraWritePD data set includes parameters with different ranges. It is necessary to rescale or normalise the data. The Min-Max normalisation technique is used to linearly convert each individual parameter signal in the range from 0 to 1.

**Feature engineering and selection** After the normalising procedure, the feature engineering process proposed in [7] suggests considering the kinematic parameters of the movements of the tip of the pen. Kinematic parameters of fine motor movements (observed during writing and drawing activities) would reflect tremor, freezing, and other symptoms caused by progressing PD [20], and we have demonstrated this conclusion through experiments (see Fig. 4 (a)). Within the framework of the present studies, main attention is paid to the projections of the velocity of x- and y-coordinate parameter signals on the coordinate axis given by Eq.(1), while keeping the other parameters constant.

$$v_x^t = (x_t - x_{t-1})/\delta_t, \quad v_y^t = (y_t - y_{t-1})/\delta_t, \quad (1)$$

where  $(x_{t-1}, y_{t-1})$  and  $(x_t, y_t)$  are the x- and y-coordinate position information of two adjacent time points, and  $\delta_t$  is the time interval between two contiguous sampling data points. The use of velocity also has the advantage of converting the non-stationary  $x, y$  coordinate-based features into stationary ones, leading to a more tractable classification under the proposed model.

**Segmentation** Afterwards, data segmentation is adopted in the proposed framework to generate suitable data samples for model training. The approach of using a sliding window to randomly select a local patch is frequently used in time-series classification and has been demonstrated to be useful

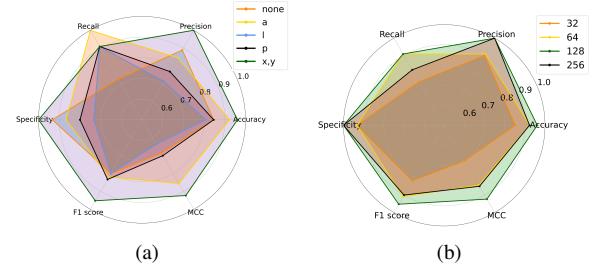


Fig. 4. The effect of hyper-parameters in data pre-processing on the model, where (a) is the effect of transforming different raw parameter signals into velocity features in Feature engineering and selection on the model, and (b) is the effect of *window size* in Segmentation on the model.

in improving model performance. Fig. 3 shows an example based on the DraWritePD data set. The multichannel time-series data are cropped into small patches with a slight overlap to preserve the temporal information. The parameters *window size* ( $w$ ) and *stride size* ( $s$ ) control the length and overlap of the resampling patches. The choice of *window size* and *stride size* depends on the concrete applications. Experiments illustrated by Fig. 4 (b) demonstrate that the  $w=128$  provides balance between the performance and the efficiency under the proposed LSTM-CNN architecture. It is also worth noting that such segmentation generates more data samples for training, which is crucial for limited or small data sampling cases, including the proposed method based on the LSTM-CNN deep learning architecture.

#### B. Model Structure

The proposed model structure is a hybrid of the LSTM and CNN models. Such a combination helps to explore the advantages of both the LSTM and CNN models. In this LSTM-CNN model, convolution operators are also reduced to 1D instances to reduce the computational cost. For clarity, we introduce the model in Fig. 5 according to the characteristics of the LSTM and CNN blocks.

**LSTM Block** The LSTM unit is a subtype of the recurrent neural network (RNN). It was originally proposed as an efficient and scalable building block for analysing complex sequential data or time-series data. The LSTM unit contains a group of special memory cells and is capable of extracting temporal features of data based on the memory of historical information, giving a great advantage over CNN in the extraction of sequence data. For each memory cell, the input data is first sent to different gates, including the input gate, the forget gate, and the output gate, to control the behaviour of each memory cell, and then the output is sent as input at the corresponding moment of the next memory cell. For more details on the structure and properties of LSTM, we refer to [10]. In our LSTM-CNN model, the LSTM block consists of a series of cascaded LSTM units. Experiments show that, in order to improve efficiency while maintaining model performance, we can choose only one LSTM unit to construct the LSTM block. Furthermore, the output of each LSTM unit is concatenated with the original input data and sent to the subsequent CNN

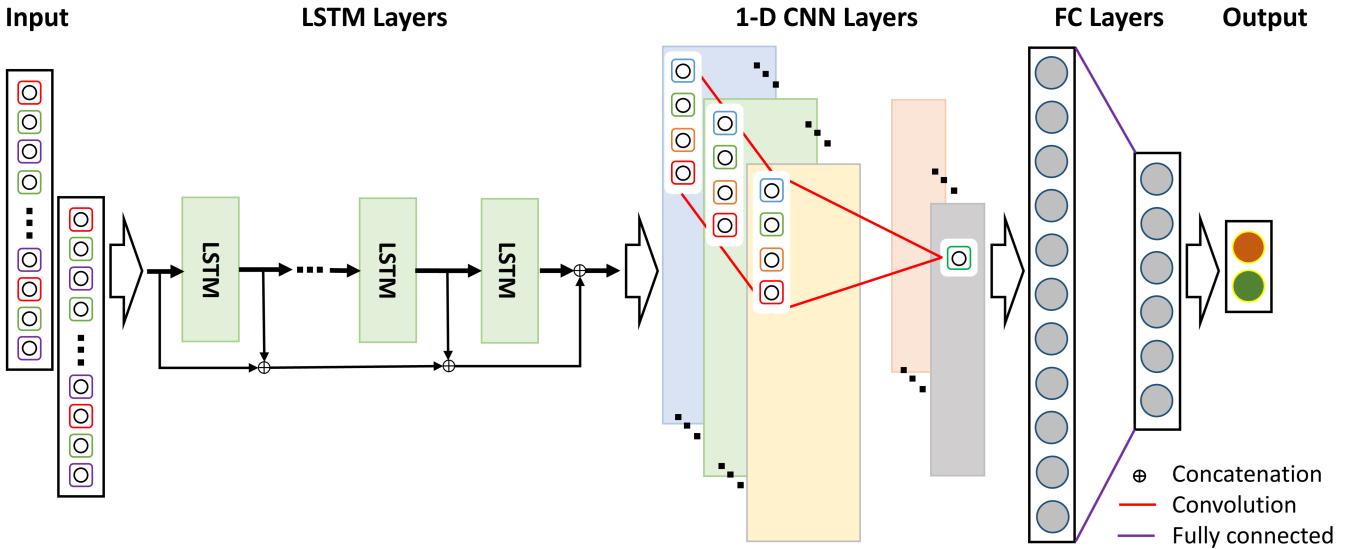


Fig. 5. The network structure of the proposed LSTM-CNN model.

block for robust classification, in which the LSTM output is dimension-expanded (represented as batch size, 1, window size, and feature dimension) to accommodate the shape of the input convolutional layer.

**CNN Block** The CNN unit is suitable for classification or recognition tasks due to its ability to learn discriminative representations. We explore the performance of convolutional networks and follow the architecture and suggestions in [13]. As shown in Fig. 5, the proposed model contains two CNN units, where each consists of a 1D temporal convolutional layer, a rectified linear unit (ReLU) layer and a 1D max pool layer. Specifically, the 1D convolutional layer is the most important component due to its unique feature extraction ability, where in the first convolution unit, the 16 convolution kernels are used for feature extraction, and in the second unit, the 32 convolution kernels are used for deeper feature extraction operations on the output of the feature by the upper layer. The size of each convolution kernel is  $3 \times 1$ , and the sliding step size of the convolution window is fixed to 2 throughout all experiments. Next, a ReLU layer is used to activate its output. A maximum pool layer is used after the ReLU layer to perform the downsampling operation and reduce parameters while maintaining dominant features, with a sampling kernel size of  $2 \times 1$  and a stride size of 2. Notice that all operators are reduced to cases of 1 -D for efficiency.

**FC Block** The fully connected (FC) block — consisting of a fully connected layer, a ReLU activation layer and a dropout layer — is used in the proposed model. During the training phase, the dropout layer temporarily removes nodes from the network with a probability of 0.5. For stochastic gradient descent, since it is randomly dropped, each mini-batch is training a different network to prevent model overfitting and improve model performance. Furthermore, a fully connected layer is deployed after the dropout layer to convert the previous output,

the value of which represents the probabilities belonging to each class.

#### IV. EXPERIMENTAL RESULTS

In this section, we evaluate and analyse the performance of the proposed LSTM-CNN model using the DraWritePD data set. The model runs on the desktop PC with an Intel(R) Core(TM) 3.60 GHz(8 CPU), 32GB RAM, and an NVIDIA RTX3070Ti GPU with 8 GB memory.

##### A. Dataset

The DraWritePD set contains 157 pieces of sequence data from 29 subjects with HC and 20 patients with PD. The raw sequence data needs to be cropped into patches before being fed to the LSTM-CNN model. The class imbalance problem may occur during the segmentation due to the different lengths of the sequence data. As shown in Fig. 3, a nonuniform sampling strategy with varying stride size is adopted to impose the number of generated patch data in each class to be the same. The statistics of the training and testing dataset is listed in Table II.

TABLE II  
THE INFORMATION OF TRAINING SET AND TESTING SET.

	Training set		Testing set	
	HC	PD	HC	PD
Participant	25	16	4	4
Sequence set (S)	80	51	15	11
Patch set (P)	16166	16836	3670	3319

During the training phase, the patch data set was randomly divided in the 8 : 2 ratio into a training patch data set and a validation patch data set. During the testing phase, the proposed model was first evaluated based on the testing patch

data set and then applied to the testing sequence data set, where the predicted result of each raw sequence data set was determined by the majority vote of the prediction result of the patch data. To clarify, we denote the patch data set testing as  $P$  and the raw sequence data set as  $S$ , and independently evaluate the performance of the proposed model on the two cases.

### B. Experimental Setup

In order to fully exploit the performance of the proposed LSTM-CNN model, we use a cross-validation strategy to optimally choose the parameters. Adam [12] optimiser is used to train the model, and the initial learning rate is set to 0.001. Furthermore, the cross-entropy loss function is used for model fitting and the batch size is set to 64. The proposed model is completed in 200 epochs with the loss curve shown in Fig. 6. We use the metrics: accuracy, precision, recall, specificity,  $F_1$  score, and Matthews correlation coefficient (MCC) for evaluation, where the latter has been adopted by many existing methods to describe the different aspects of the performance of a classifier [1]. Once the training phase is completed, the one with the best fitness value is chosen for testing. Moreover, the length of segmented patches and the choice of feature selection are also discussed to interpret their roles in determining the model performance. As shown in Fig. 4, the model achieves the best classification result when the window size is 128 by using the  $(v_x, v_y)$  velocity characteristics. The model performance is eventually tested on both the original sequences dataset ( $S$ ) and the segmented patches dataset ( $P$ ).

### C. Quantitative Evaluation and Comparison

We provide a quantitative comparison to demonstrate the effectiveness and advantages of the proposed LSTM-CNN model. First, we compare it with some traditional machine learning (ML)-based classifiers, which include the Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF), and LightGBM (LGB) [11]. For each classifier, we take 10-fold cross-validations and the grid search algorithm to optimise the parameters and to ensure the robustness of the results. The training and validation of all these ML classifiers are run under Python's scikit-learn library [18]. As shown in Table III, our model has obvious advantages in most classification metrics, in terms of accuracy increased by 3.8%,  $F_1$  score increased by 3.5%, and MCC increased by 6.6%. Regarding the efficiency of the model, our method outperforms 1.8 seconds and 5.3 seconds compared to the SVM model and the RF model, although it is slower than the optimised LR model and the LGB model.

Additionally, we also compare the performance of different neural network models on this task. The multilayer perceptron (MLP) is the basic model and its structure consists of two fully connected layers. AlexNet adopts a convolutional neural network structure similar to AlexNet [13], but to adapt to the size of the input data, the internal parameters are modified. In our work, to improve the efficiency of the LSTM-CNN, in addition to the 1D convolution operation in the CNN block,

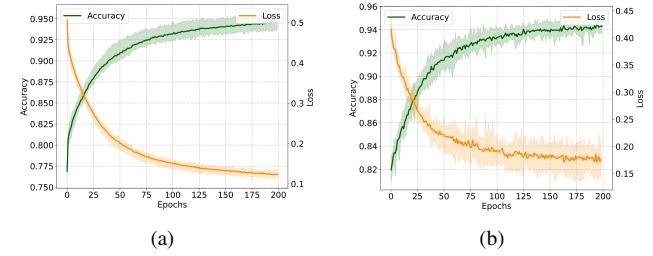


Fig. 6. The accuracy and loss curves of the LSTM-CNN model on the training set(a) and the validation set(b), respectively, where the solid curve represents the average of multiple experiments, and the shaded part represents the range of the results of multiple experiments (N=10).

an LSTM block is added, which contains a concatenation operation. Additionally, as shown in Fig.6, the average of multiple experimental results (N=10) is used as the model result. Finally, let us point out that our methods achieve optimal results in all metrics with the accuracy rate being 96.15%, the  $F_1$  score being 95.24%, and the MCC being 92.32%. Specifically, in the  $S$  testing set, only one sequence data from the PD category is misclassified, and the remaining 25 sequence data are correctly classified.

In summary, the model proposed in this article could not only achieve high recognition accuracy, but also significantly simplify the structure of the model and improve the efficiency of deep learning models in the diagnosis of PD.

## V. CONCLUSIONS

This article proposes a lightweight deep neural network that combines convolutional layers 1D with LSTM for the diagnosis of PD based on hand-drawn tests. In particular, the concatenation operation is added after the LSTM layer, which significantly improves the goodness of the model. Not only can it avoid complex feature extraction, but also it has a high recognition accuracy under the premise of a few model parameters.

## ACKNOWLEDGEMENTS

This study was supported by the PRG Grant957 of the Estonian Research Council. This work in the project “ICT programme” was supported by the European Union through the European Social Fund. It was also partially supported by the FWO Odysseus 1 grant G.OH94.18N: Analysis and Partial Differential Equations, and the Methusalem programme of the Ghent University Special Research Fund (BOF) (Grant number 01M01021). Michael Ruzhansky is also supported by EPSRC grant EP/R003025/2. Marianna Chatzakou is a postdoctoral fellow of the Research Foundation – Flanders (FWO) under the postdoctoral grant No 12B1223N.

## REFERENCES

- [1] P. Baldi, S. Brunak, Y. Chauvin, C. A. Andersen, and H. Nielsen, Assessing the accuracy of prediction algorithms for classification: an overview, *Bioinformatics*, vol. 16, no. 5, pp. 412–424, 2000.

TABLE III  
QUANTITATIVE COMPARISON OF DIFFERENT CLASSIFICATION METHODS.

Model	Inference time (s)	Metric					
		Accuracy (P/S)	Precision (P/S)	Recall (P/S)	Specificity (P/S)	$F_1$ score (P/S)	MCC (P/S)
LR	0.034	0.8061 / 0.9231	0.8559 / 0.8462	0.8565 / <b>1.00</b>	0.7018 / 0.8667	0.8562 / 0.9167	0.5585 / 0.8563
SVM	6.060	0.8371 / 0.8846	0.8657 / 0.7857	0.8977 / <b>1.00</b>	0.7119 / 0.8000	0.8814 / 0.8800	0.6229 / 0.7928
RF	9.526	0.8339 / 0.8462	0.9015 / 0.8889	0.8412 / 0.7273	0.8088 / 0.9333	0.8729 / 0.8000	0.6368 / 0.6860
LGB	0.161	0.7889 / 0.8077	0.9183 / 0.8750	0.7538 / 0.6364	0.8613 / 0.9333	0.8280 / 0.7368	0.5800 / 0.6098
MLP	4.095	0.8274 / 0.8846	0.8598 / 0.8333	0.8921 / 0.9091	0.6891 / 0.8667	0.8756 / 0.8696	0.5950 / 0.7688
AlexNet	4.143	0.7872 / 0.8846	0.9093 / <b>1.00</b>	0.7606 / 0.7273	0.8425 / <b>1.00</b>	0.8284 / 0.8421	0.5662 / 0.7785
LSTM-CNN (Ours)	4.212	0.7994 / <b>0.9615</b>	0.8883 / <b>1.00</b>	0.8039 / 0.9091	0.7900 / <b>1.00</b>	0.8440 / <b>0.9524</b>	0.5720 / <b>0.9232</b>

- [2] M. P. Broderick, A. W. Van Gemmert, H. A. Shill, and G. E. Stelmach, Hypometria and bradykinesia during drawing movements in individuals with parkinson's disease, *Experimental brain research*, vol. 197, no. 3, pp. 223–233, 2009.
- [3] D. B. Calne, Treatment of parkinson's disease, *New England Journal of Medicine*, vol. 329, no. 14, pp. 1021–1027, 1993.
- [4] C. Che, C. Xiao, J. Liang, B. Jin, J. Zho, and F. Wang, An rnn architecture with dynamic temporal matching for personalized predictions of parkinson's disease, in *Proceedings of the 2017 SIAM international conference on data mining. SIAM*, 2017, pp. 198–206
- [5] E. R. Dorsey, A. Elbaz, E. Nichols, N. Abbasi, F. Abd-Allah, A. Abdalim, J. C. Adsuar, M. G. Ansha, C. Brayne, J.-Y. J. Choi et al., Global, regional, and national burden of parkinson's disease, 1990–2016: a systematic analysis for the global burden of disease study 2016, *The Lancet Neurology*, vol. 17, no. 11, pp. 939–953, 2018.
- [6] P. Drotár, J. Mekyska, I. Rektarová, L. Masarová, Z. Smékal, and M. Faundez-Zanuy, A new modality for quantitative evaluation of parkinson's disease: In-air movement, in *13<sup>th</sup> IEEE international conference on bioinformatics and bioengineering. IEEE*, 2013, pp. 1–4
- [7] P. Drotár, J. Mekyska, I. Rektarová, L. Masarová, Z. Smékal, and M. Faundez-Zanuy, Evaluation of handwriting kinematics and pressure for differential diagnosis of parkinson's disease, *Artificial Intelligence in Medicine*, vol. 67, pp. 39 – 46, 2016.
- [8] T. Fujita, Z. Luo, C. Quan, K. Mori, and S. Cao, Performance evaluation of rnn with hyperbolic secant in gate structure through application of parkinson's disease detection, *Applied Sciences*, vol. 11, no. 10, p. 4361, 2021.
- [9] F. Goodenough, *Measurement of Intelligence by Drawings*, ser. Classics in child development. World book Company, 1926. [Online]. Available: <https://books.google.ee/books?id=lrmcAAAAMAAJ>
- [10] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [11] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, Lightgbm: A highly efficient gradient boosting decision tree, *Advances in neural information processing systems*, vol. 30, 2017.
- [12] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980*, 2014.
- [13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, vol. 25, 2012.
- [14] E. D. Louis and D. G. Machado, Tremor-related quality of life:a comparison of essential tremor vs. parkinson's disease patients, *Parkinsonism & related disorders*, vol. 21, no. 7, pp. 729–735, 2015.
- [15] C. Marquardt and N. Mai, A computational procedure for movement analysis in handwriting, *Journal of Neuroscience Methods*, vol. 52, no. 1, pp. 39 – 45, 1994.
- [16] M. Moletesum, I. Siddiqi, N. Vincent, and F. Cloppet, Assessing visual attributes of handwriting for prediction of neurological disorders—a case study on parkinson's disease,*Pattern Recognition Letters*, vol. 121, pp. 19–27, 2019.
- [17] S. Nömm, S. Zaremba, K. Medijainen, P. Taba, and A. Toomela, Deep cnn based classification of the archimedes spiral drawing tests to support diagnostics of the parkinson's disease, *IFAC-PapersOnLine*, vol. 53, no. 5, pp. 260–264, 2020.
- [18] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg et al., Scikit-learn: Machine learning in python, *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.
- [19] C. R. Pereira, S. A. Weber, C. Hook, G. H. Rosa, and J. P. Papa, Deep learning-aided parkinson's disease diagnosis from handwritten dynamics, in *2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*. Ieee, 2016, pp. 340–346
- [20] S. Rosenblum, M. Samuel, S. Zlotnik, I. Erikh, and I. Schlesinger, Handwriting as an objective tool for parkinson's disease diagnosis, *Journal of neurology*, vol. 260, no. 9, pp. 2357–2361, 2013.
- [21] R. Saunders-Pullman, C. Derby, K. Stanley, A. Floyd, S. Bressman, R. B. Lipton, A. Deligtisch, L. Severt, Q. Yu, M. Kurtis et al., Validity of spiral analysis in early parkinson's disease, *Movement disorders: official journal of the Movement Disorder Society*, vol. 23, no. 4, pp. 531–537, 2008.
- [22] E. Smits, A. Tolonen, L. Cluitmans, M. Gils, B. Conway, R. C Zietsma, K. Leenders, and N. Maurits, Standardized Handwriting to Assess Bradykinesia, Micrographia and Tremor in Parkinson's disease, *PloS one*, vol. 9, 05 2014.
- [23] E. Valla, S. Nömm, K. Medijainen, P. Taba, and A. Toomela, Tremor-related feature engineering for machine learning based parkinson's disease diagnostics, *Biomedical Signal Processing and Control*, vol. 75, p. 103551, 2022.
- [24] S. Zaremba, S. Nömm, K. Medijainen, P. Taba, and A. Toomela, Cnn based analysis of the luria's alternating series test for parkinson's disease diagnostics, in *Asian Conference on Intelligent Information and Database Systems*. Springer, Singapore, 2021, pp. 3–13