State-Of-The-Art Techniques For Exploiting Loop-Like Tasks In The Classification Of Parkinson’s Disease Patients

Content

Moises Diaz et al, 2021

Sequence-based dynamic handwriting analysis for Parkinson's disease detection with one-dimensional convolutions and BiGRUs

Catherine Taleb et al, 2019

Improving Deep Learning Parkinson's Disease Detection Through Data Augmentation Training

Catherine Taleb et al, 2020

Detection of Parkinson's disease from handwriting using deep learning: a comparative study

Maria Angelillo et al, 2019

Performance-driven Handwriting Task Selection for Parkinson's Disease Classification

Gazda Matej et al, 2022

Ensemble of convolutional neural networks for Parkinson's disease diagnosis from offline handwriting

Donato Impedovo et al, 2018

Dynamic Handwriting Analysis for Supporting Earlier Parkinson's Disease Diagnosis

Momina Moetesum et al, 2019

Assessing visual attributes of handwriting for prediction of neurological disorders—A case study on Parkinson's disease

Christian Kahindo et al, 2018

Characterizing Early Stage Alzheimer through Spatiotemporal Dynamics of Handwriting

Propositions

Sequence-based dynamic handwriting analysis for Parkinson’s disease detection with one-dimensional convolutions and BiGRUs

Moises Diaz et al, 2021

Introduction

- Goal : The classification of PD patients using on 1D convolutions and BiGRUs (Bidirectional gated recurent units), to assess the potential of sequential information (the sequences of motion events and their spatio-temporal properties), using raw sequences and derived features.
- Results : The proposed method outperformed state-of-the-art approaches on the PaHaW dataset, and achieved competitive results on the NewHandPD dataset.

The PaHaW Dataset

- The “Parkinson’s disease handwriting database” (PaHaW) collects hand-writing data of Participants were enrolled at the First Department of Neurology, Masaryk University, and the St. Anne’s University Hospital, Brno, Czech Republic :
- 37 PD patients and 38 age and gender-matched healthy controls (HC).
- Right-handed.
- At least 10 years of education.
- Czech as their native language. No significant between-group difference regarding age or gender was found.
- No history or presence of any psychiatric symptom or disease affecting the central nervous system, with the exception of Parkinsonism in the PD group.
- Patients were only examined in their ON-state while taking dopaminergic medication.
- PDs were evaluated by a qualified neurologist.
- HCs underwent a thorough examination to ensure that no movement disorder or injury could have significantly affected handwriting.

The tasks performed :

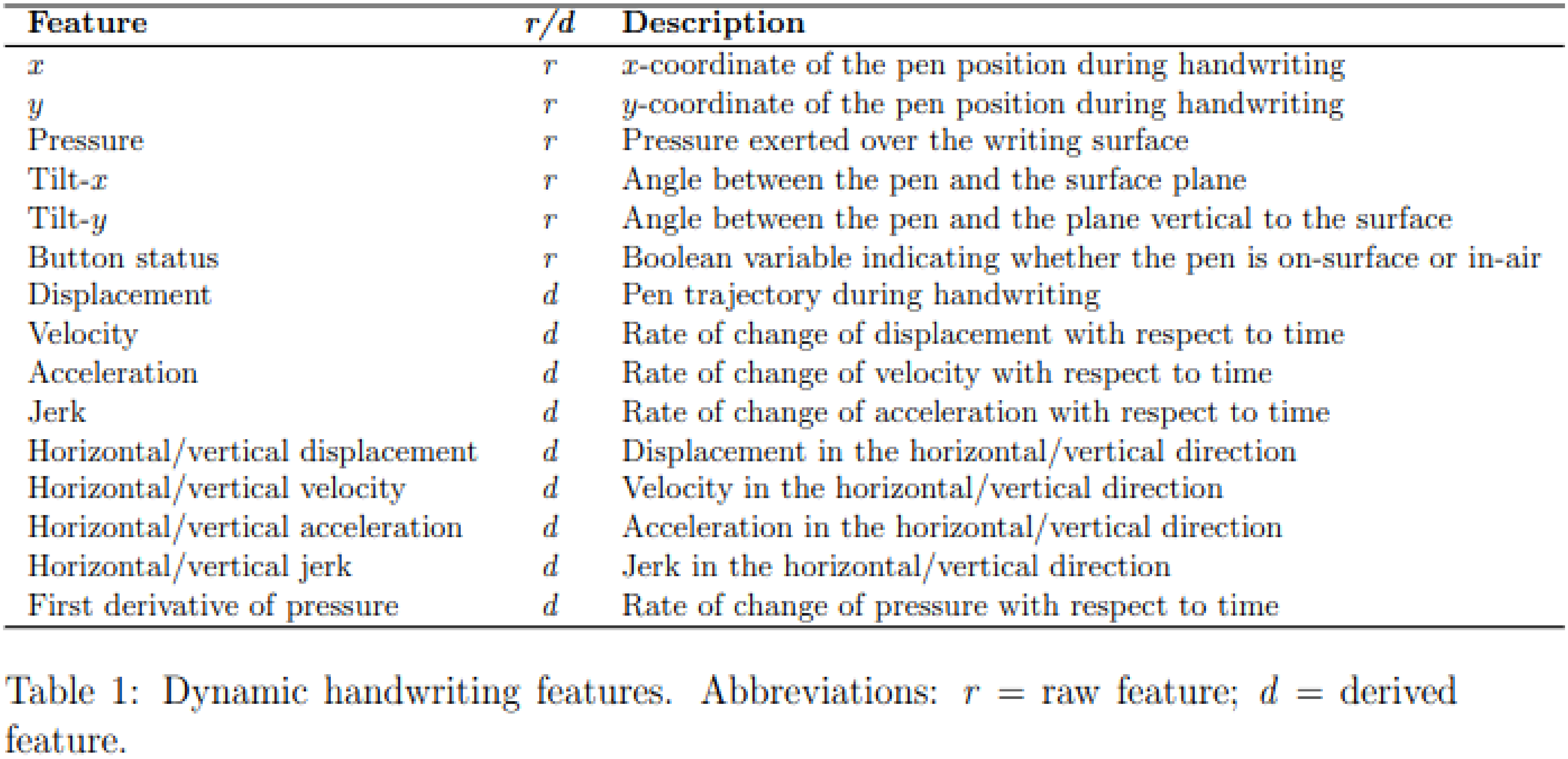
- Drawing an Archimedes spiral
- Writing in cursive the letter ***l***
- The biagram ***le***
- The triagram ***les***
- Writing in cursive the word ***lektorka*** (“female teacher” in Czech)
- ***porovnat*** (“to compare”)
- ***nepopadnout*** (“to not catch”)
- Writing in cursive the sentence ***Tramvaj dnes ûz nepojede*** (“The tram won’t go today”).

- The handwriting signals were recorded using a Wacom Intuos digitizing tablet, overlaid with a blank sheet of paper. Like many other professional tablets, the raw data acquired are the x- and y-coordinates of the pen tip, the corresponding time stamps, measures of pen inclination, i.e. tilt-x and tilt-y, and pen pressure. The button status is also available, which is a binary variable with value 0 for pen-ups ("in-air movement") and 1 for pen-downs ("on-surface movement"). The sampling rate was 200 samples per second.
- Since not all participants completed each task, we considered only those subjects who completed each of the eight tasks, i.e. 36 PD and 36 HC.

The NewHandPD Dataset

- The NewHandPD database is an extension of the previous HandPD corpus. The first database consisted of images from two drawing tasks, i.e. the typical spiral cognitive test and a modified spiral (“meander”) test performed by healthy individuals and people with Parkinson’s disease. However, the new corpus, NewHandPD, contains both offline images and online signals (time-based sequences) of the two groups, and other tasks, such as circle tasks and diadochokinesis tests.
- The handwriting signals were acquired through a technology other than a tablet, i.e. an electronic smart pen (BiSP).
- Since the NewHandPD doesn’t contain any tasks related to repititive cursive ***l*** or ص , all the studies and methods performed on it and their results are not included in this presentation.

Feature Extraction



Feature	r/d	Description
x	r	x-coordinate of the pen position during handwriting
y	r	y-coordinate of the pen position during handwriting
Pressure	r	Pressure exerted over the writing surface
Tilt-x	r	Angle between the pen and the surface plane
Tilt-y	r	Angle between the pen and the plane vertical to the surface
Button status	r	Boolean variable indicating whether the pen is on-surface or in-air
Displacement	d	Pen trajectory during handwriting
Velocity	d	Rate of change of displacement with respect to time
Acceleration	d	Rate of change of velocity with respect to time
Jerk	d	Rate of change of acceleration with respect to time
Horizontal/vertical displacement	d	Displacement in the horizontal/vertical direction
Horizontal/vertical velocity	d	Velocity in the horizontal/vertical direction
Horizontal/vertical acceleration	d	Acceleration in the horizontal/vertical direction
Horizontal/vertical jerk	d	Jerk in the horizontal/vertical direction
First derivative of pressure	d	Rate of change of pressure with respect to time

Table 1: Dynamic handwriting features. Abbreviations: r = raw feature; d = derived feature.

- Note that other commonly used spatio-temporal variables were not considered, such as stroke size and duration, overall time, etc., as they are expressed as a single-valued feature rather than a time-dependent vector feature.
- Each handwriting sample S_n can therefore be represented as a multidimensional vector of m dynamic features as follows:

$$S_{n=1}^{N} = \{X_1^n, X_2^n, \dots X_m^n\}$$

$$X_{i=1}^{m} = \{x_i^{t_1}, x_i^{t_2}, \dots x_i^{T}\},$$

N: The size of the dataset.
m: The number of features.
T: The number of time-steps in a sequence.

- Since T is arbitrary for each sequence, a cut-off was set to the mean of the number of time-steps for all the sequences to avoid negatively impacting the training time for long sequences, and underfitting for short sequences. On shorter sequences zero-padding is applied.

One-Dimensional Convolutions

- 1D convolutions are a type of operation used in deep learning to process sequential data, such as time series, audio, or text. They involve sliding a small filter (also known as a kernel) along the input sequence to capture local patterns and dependencies. The filter's learnable weights are shared across different positions, allowing the model to efficiently extract relevant features from various parts of the sequence. By using a stride greater than 1, 1D convolutions can downsample the data and reduce its spatial dimensions. This technique is especially useful for tasks that require analyzing the order of elements in the data while benefiting from parameter sharing, scalability, and time-invariant feature extraction.

In this study 1D-Convolutions with a stride greater than 1 were used for 2 reasons:

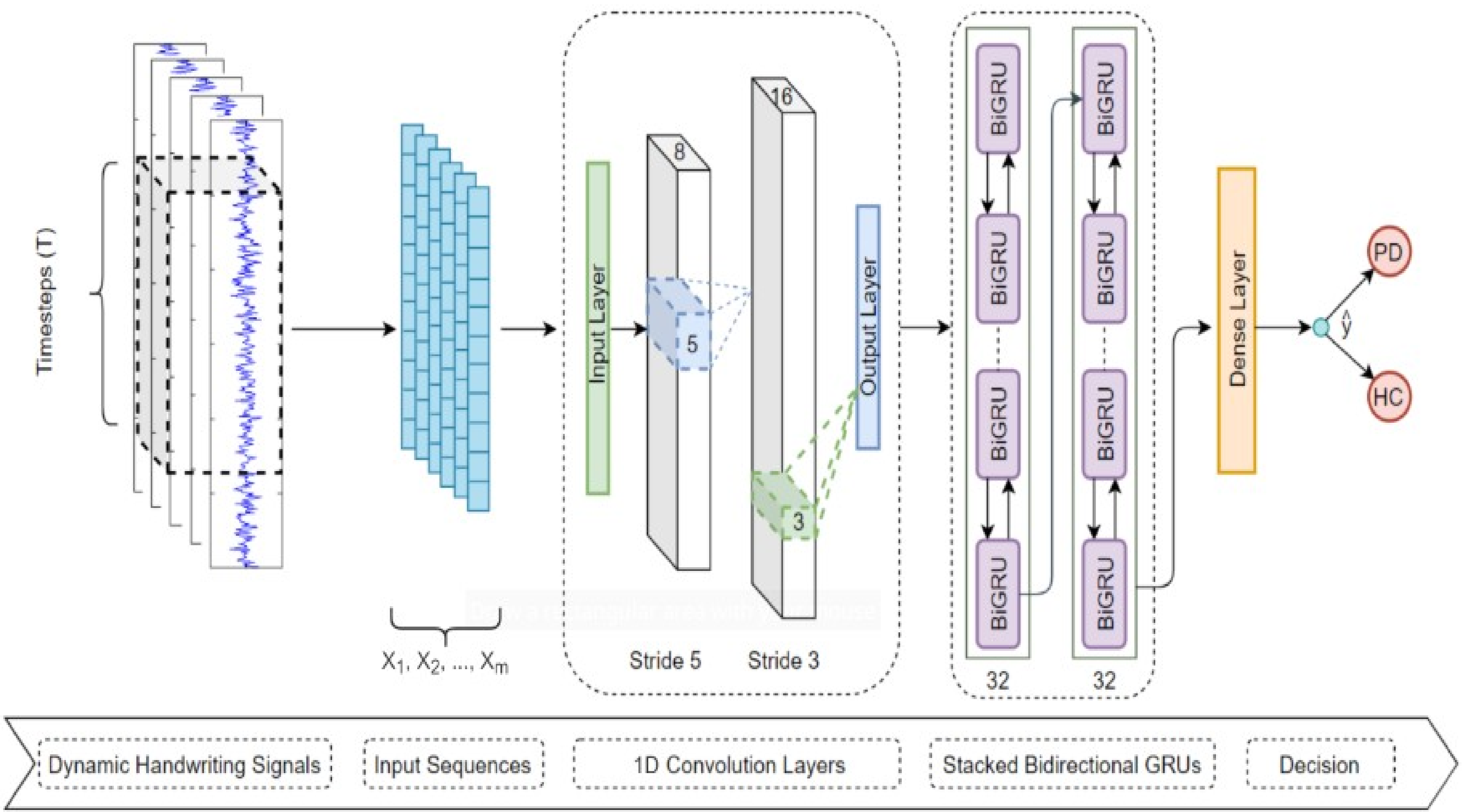
- To turn the long input sequence into much shorter (down-sampled) pieces of higher-level, locally invariant features, that can be processed easily when inputed to the RNN.
- 1D convolutions can extract local temporal information from the input sequences, thus performing a pre-training step towards learning meaningful temporal dependencies.

Bidirectional Gated Recurent Units (Bi-GRUs)

- Bi-GRUs are a type of recurrent neural network (RNN) architecture designed to capture information from both past and future contexts in sequential data. Unlike traditional GRUs, which process sequences in a forward direction only, Bi-GRUs process sequences in both forward and backward directions simultaneously. This means that at each time step, a Bi-GRU has two hidden states: one representing the information from the past and the other representing the information from the future.
- By considering information from both directions, Bi-GRUs can capture long-range dependencies (that are often lost in RNNs due to the problem of vanishing gradient) and contextual information effectively, making them well-suited for tasks like natural language processing, speech recognition, and any other tasks that benefit from bidirectional context modeling.

- In order to guarantee the capturing of long-term dependencies, which can fail in RNNs due to the vanishing gradient problem, either the Long-Short Term Memory (LSTM) or the Gated Recurrent Units (GRUs) are used.
- In this study, a GRU based model was chosen because it’s less computationally expensive than LSTMs due to the lower number of gates and therefore fewer parameters to learn.
- Furthermore, Bi-GRUs are used to process a sequence in both directions (forward and backward) to capture patterns that a unidirectional model might overlook.

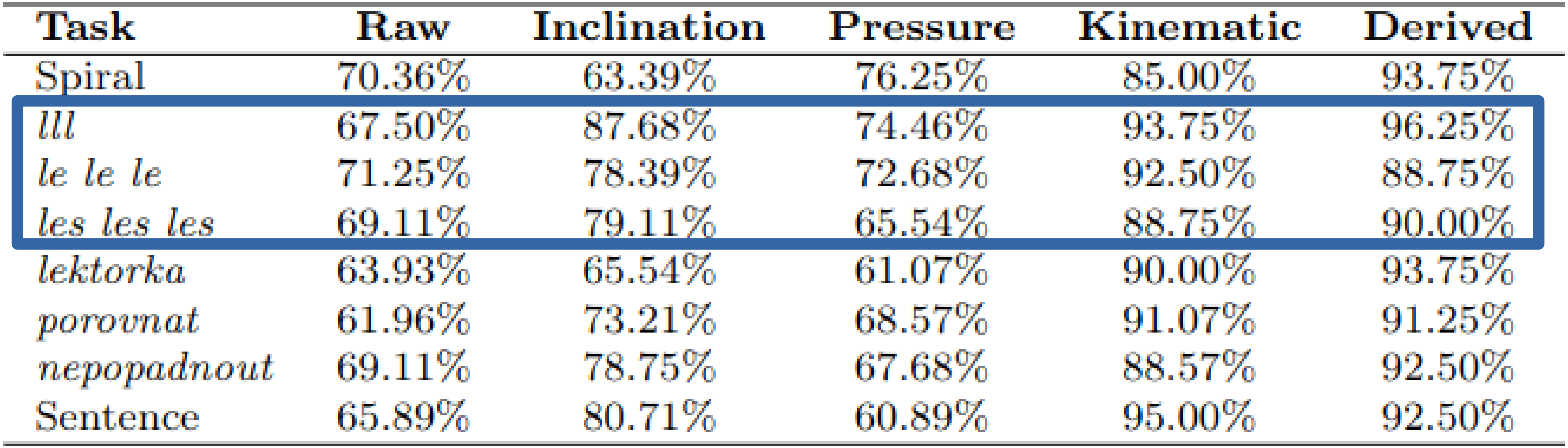
Model Architecture



Raw data from handwriting signals are converted into feature sequences. The input to the model is a sequence of length m and each time-step is a vector of the aformentioned features. There are two 1D convolutional layers with 8 and 16 filters and strides of 5 and 3, respectively. Two Bi-directional GRU layers, each with 32 units. The output layer has a single neuron with sigmoid activation to predict one of the two classes (PD or HC).

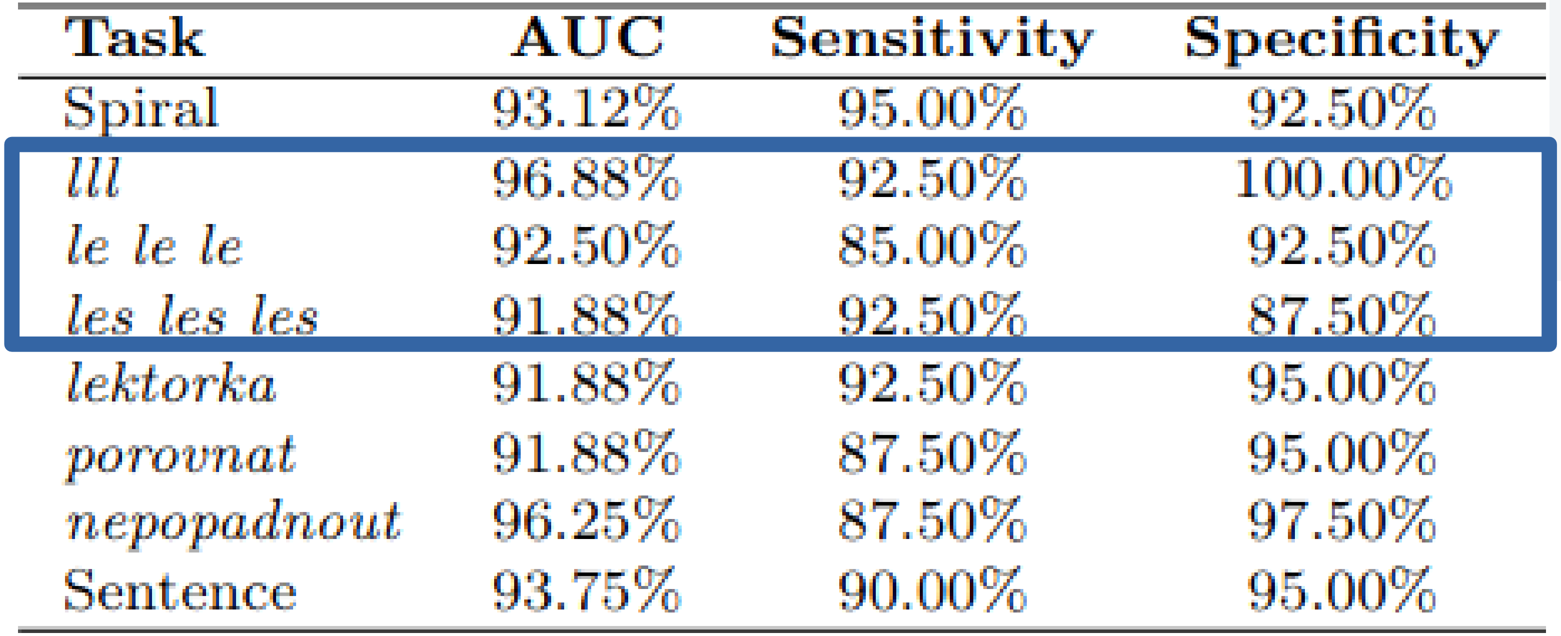
Classification on PaHaW

- To assess the performance of the model, mean accuracy values are reported, averaged over all the iterations of a 10-fold cross-validation scheme on each of the following feature sets :
- Raw
- Inclination
- Pressure
- Kinematic
- Derived



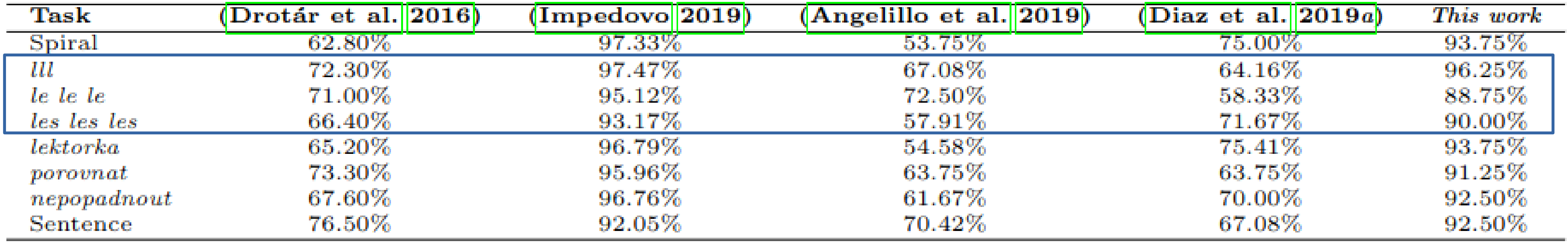
Task	Raw	Inclination	Pressure	Kinematic	Derived
Spiral	70.36%	63.39%	76.25%	85.00%	93.75%
lll	67.50%	87.68%	74.46%	93.75%	96.25%
le le le	71.25%	78.39%	72.68%	92.50%	88.75%
les les les	69.11%	79.11%	65.54%	88.75%	90.00%
lektorka	63.93%	65.54%	61.07%	90.00%	93.75%
porovnat	61.96%	73.21%	68.57%	91.07%	91.25%
nepopadnout	69.11%	78.75%	67.68%	88.57%	92.50%
Sentence	65.89%	80.71%	60.89%	95.00%	92.50%

- Mean accuracy for the individual subsets of features for each task.
- Best performing feature set: Derived and kinematic both with exactly the same mean score of **91.66666666666667%** in the mean accuracy for the 3 tasks.
- Best performing task of all tasks: '*lll*' on the derived feature set **(96.25%)**, which clearly shows the impairment of PD patients in fine motor control during loop-like movements.



Task	AUC	Sensitivity	Specificity
Spiral	93.12%	95.00%	92.50%
lll	96.88%	92.50%	100.00%
le le le	92.50%	85.00%	92.50%
les les les	91.88%	92.50%	87.50%
lektorka	91.88%	92.50%	95.00%
porovnat	91.88%	87.50%	95.00%
nepopadnout	96.25%	87.50%	97.50%
Sentence	93.75%	90.00%	95.00%

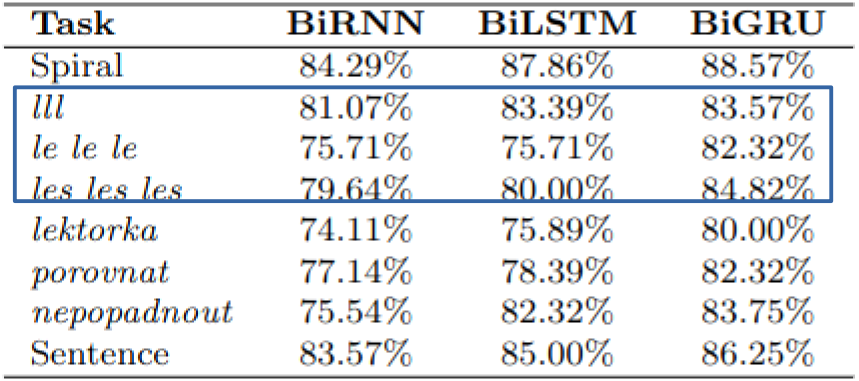
- Other accuracy measures for the best performing subset for all tasks (Derived).
- ‘*lll*’ is still clearly the best performing task in all the accuracy measures.
- The method appears to be slightly biased in favor of specificity over sensitivity. This suggests that a screening test based on this model will be better at correctly classifying healthy subjects.



Task	(Drotár et al. 2016)	(Impedovo 2019)	(Angelillo et al. 2019)	(Diaz et al. 2019*a*)	*This work*
Spiral	62.80%	97.33%	53.75%	75.00%	93.75%
lll	72.30%	97.47%	67.08%	64.16%	96.25%
le le le	71.00%	95.12%	72.50%	58.33%	88.75%
les les les	66.40%	93.17%	57.91%	71.67%	90.00%
lektorka	65.20%	96.79%	54.58%	75.41%	93.75%
porovnat	73.30%	95.96%	63.75%	63.75%	91.25%
nepopadnout	67.60%	96.76%	61.67%	70.00%	92.50%
Sentence	76.50%	92.05%	70.42%	67.08%	92.50%

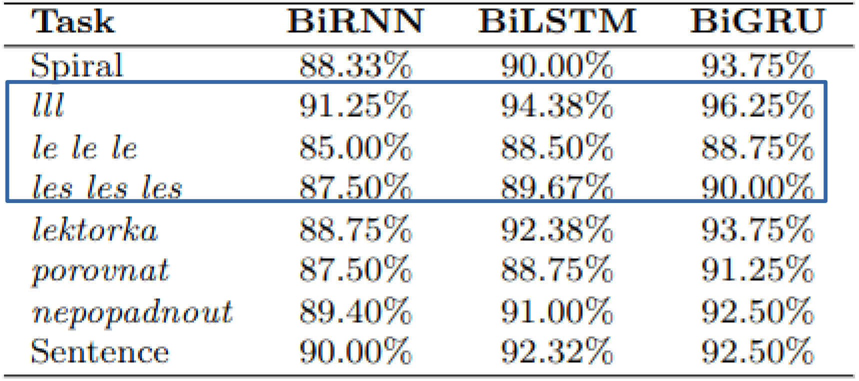
Performance comparison with state-of-the-art approaches on PaHaW.

Ablation Study



Task	BiRNN	BiLSTM	BiGRU
Spiral	84.29%	87.86%	88.57%
lll	81.07%	83.39%	83.57%
le le le	75.71%	75.71%	82.32%
les les les	79.64%	80.00%	84.82%
lektorka	74.11%	75.89%	80.00%
porovnat	77.14%	78.39%	82.32%
nepopadnout	75.54%	82.32%	83.75%
Sentence	83.57%	85.00%	86.25%

Comparison between different RNN models without convolution.



Task	BiRNN	BiLSTM	BiGRU
Spiral	88.33%	90.00%	93.75%
lll	91.25%	94.38%	96.25%
le le le	85.00%	88.50%	88.75%
les les les	87.50%	89.67%	90.00%
lektorka	88.75%	92.38%	93.75%
porovnat	87.50%	88.75%	91.25%
nepopadnout	89.40%	91.00%	92.50%
Sentence	90.00%	92.32%	92.50%

Comparison between different RNN models with 1D convolution.

Conclusion

- In this study, a new model was proposed based on one-dimensional convolutions and Bidirectional GRUs to identify distinctive patterns in the handwriting sequences of PD patients and controls.
- Different sets of dynamic features acquired from on-line graphomotor samples of both groups were fed to the model as input.
- Convolutional layers perform sub-sampling and learn effective feature representations before sending sequences to the Bidirectional GRU part of the network.
- The results of this experimental study indicate the effectiveness of the proposed technique with respect to the state-of-the-art. The proposed method, in fact, outperformed other “holistic” approaches, thus confirming the effectiveness of the sequence learning paradigm for processing sequential handwriting data.
- A significant limitation of the present study is the small size of the datasets employed, nevertheless, despite this constraint, the reported performance values are indeed very promising and the results of this study are expected to make way for a working system in the clinical settings.

Improving Deep Learning Parkinson’s Disease Detection Through Data Augmentation Training

Detection of Parkinson’s disease from handwriting using deep learning: a comparative study

Catherine Taleb et al

Introduction

- Goal: Investigate transfer learning and data augmentation approaches in order to train deep learning models for PD detection on large-scale data using the convolutional neural network (CNN) and the convolutional neural network- bidirectional long short term memory network (CNN-BLSTM).
- Results: Experimental results show that the CNN-BLSTM model used with the combination of Jittering and Synthetic data augmentation methods provides promising results in the context of early PD detection, with accuracy reaching 97.62% for all the combined tasks, but poor mono-task performance was observed.

The HandPDMultiMC dataset

- 7 handwriting tasks.
- 21 HC controls and 21 PD patients in their “on- state” (with dopaminergic medication).
- Participants were required to write on a sheet of paper laid on the tablet.
- The handwriting tasks are separated into two parts. Part I includes the free writing tasks. Part II includes the copying tasks.
- In the copying tasks, participants are asked to copy patterns and words which were preprinted into 3 different languages (Arabic, French and English) on the left of the sheet paper placed on the tablet.

The writing tasks are:

- Task 1: Drawing repetitive cursive letter ℓ
- Task 2: Drawing a triangular wave
- Task 3: Drawing a rectangular wave. For tasks 1–3, subjects were asked to proceed copying the patterns from left to right until 10 cycles.
- Task 4: Repetitive writing of word ‘Monday’ within the word sequence Monday–Tuesday. These words may be written in the subject’s familiar language. Subjects were asked to write this sequence 5 times.
- Task 5: Repetitive writing of word ‘Tuesday’. (See Task 4 remarks).
- Task 6: Repetitive writing of first name.
- Task 7: Repetitive writing of last name. For Tasks 6 and 7, subjects were asked to write their full name 5 times, each time on a different line.

Data have been registered using a Wacom Intuos 5 tablet and a special pen device, with a sample rate of 197 points/s and high spatial and pressure accuracies. The following measurements are collected per sample point:

- Pen tip position in X-axis, Y-axis, and Z-axis. The Z coordinate is registered when the pen tip is within 0–1 cm above the tablet. When Z equals 0, the pen is on tablet and when Z > 0 an in-air point is registered.
- Pen tip pressure on the surface of the tablet.
- Altitude and Azimuth angle of the pen with respect to the tablet.
- Time stamp.

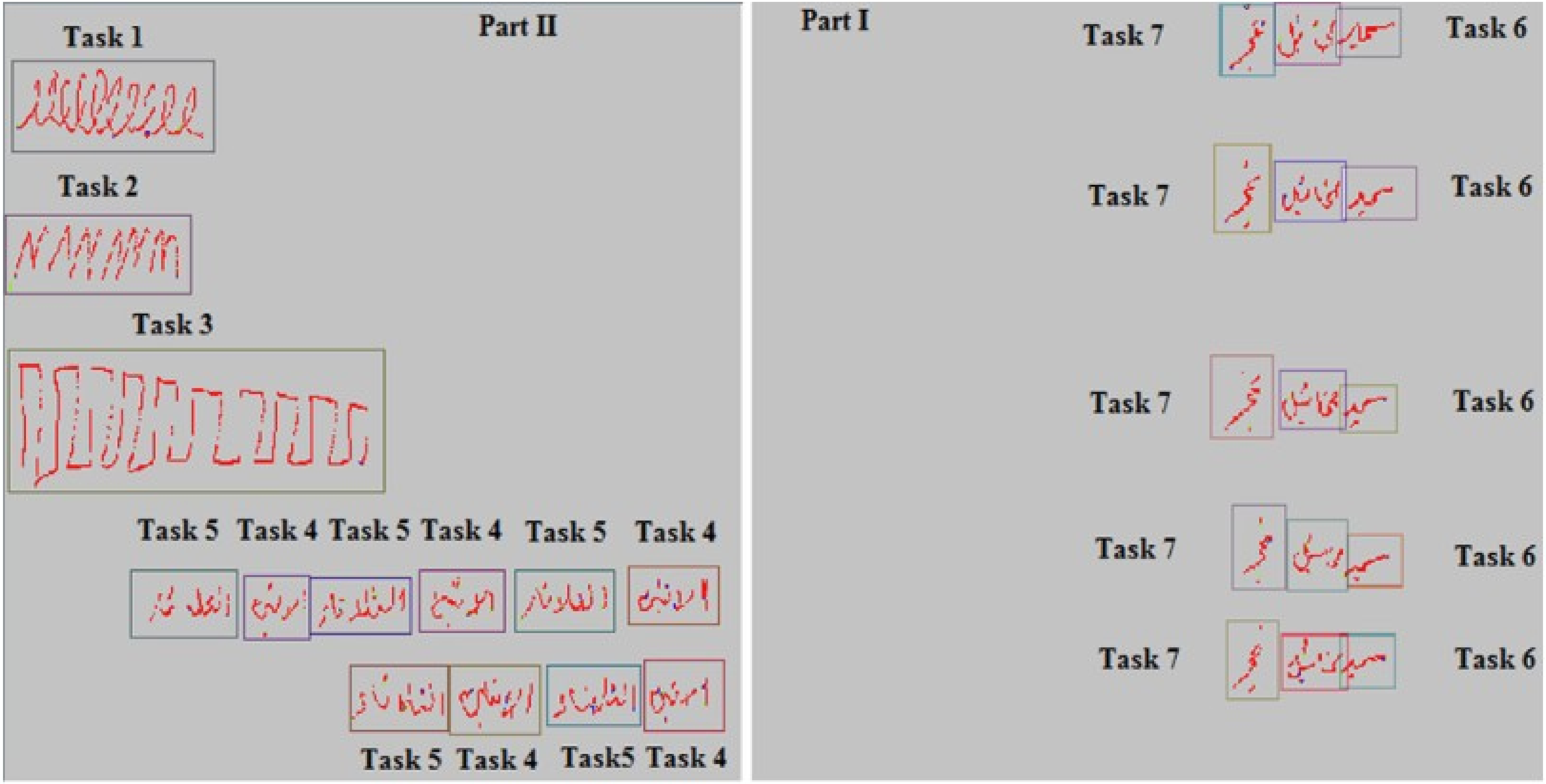


Fig. 1 The seven tasks segmented from the sheet filled by a PD subject

Pre-processing

- Arabic X coordiantes are flipped to have the same direction as english and french.
- X and Y coordinates are normalized to a range of (0, 1)

2D representation of time series

- Each handwriting task is composed of n rows (time) and 7 columns (X, Y, Z, pressure, altitude, azimuth, and time stamp).
- An optimal selection of time series features to be used is performed and a hyper-parameter k between 1 and 7 must be determined.
- 2 approaches are proposed:

  The first approach for 2D representation consists in concatenating the k time series and reshaping the result as an image.
- The second approach computes spectrograms and use them as 2D representations.

The concatenation approach

- This approach consists in transforming k time series of length n into a single image.
- The whole data (n × k matrix) is transformed into one image by concatenating the n rows into one vector and then reshaping it into a square matrix of size (√(n × k) , √(n × k)).
- This square is resized to 64 × 64 pixels resolution using Lanczos resampling method.
- Lanczos resampling is a high-quality image resampling technique used to resize digital images while preserving their visual integrity.

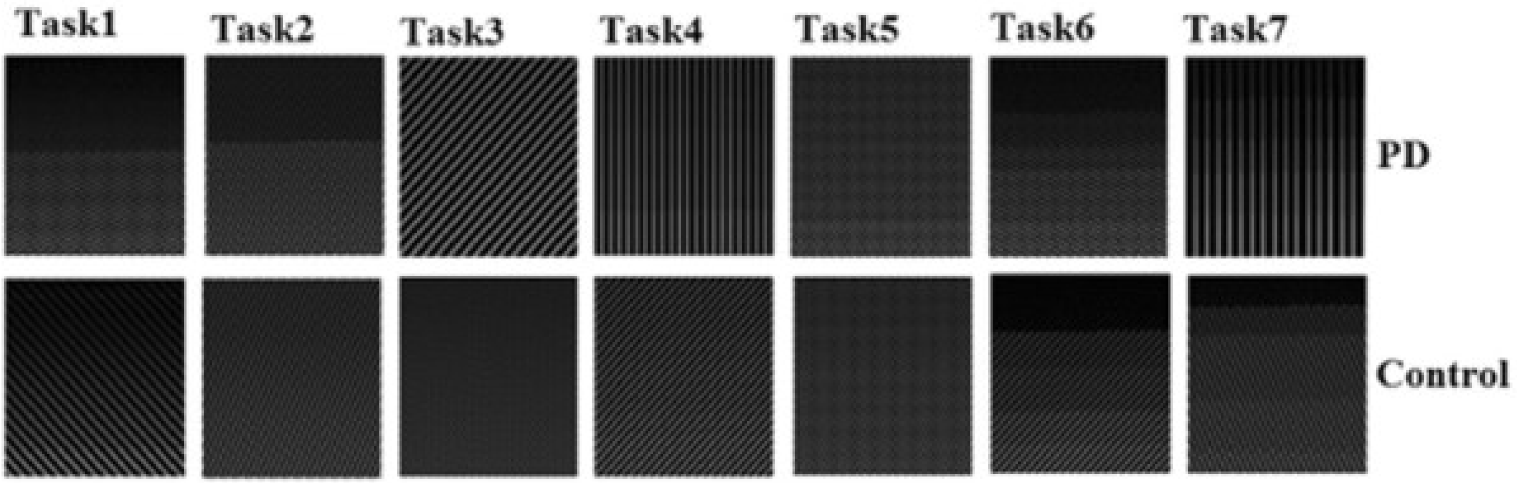
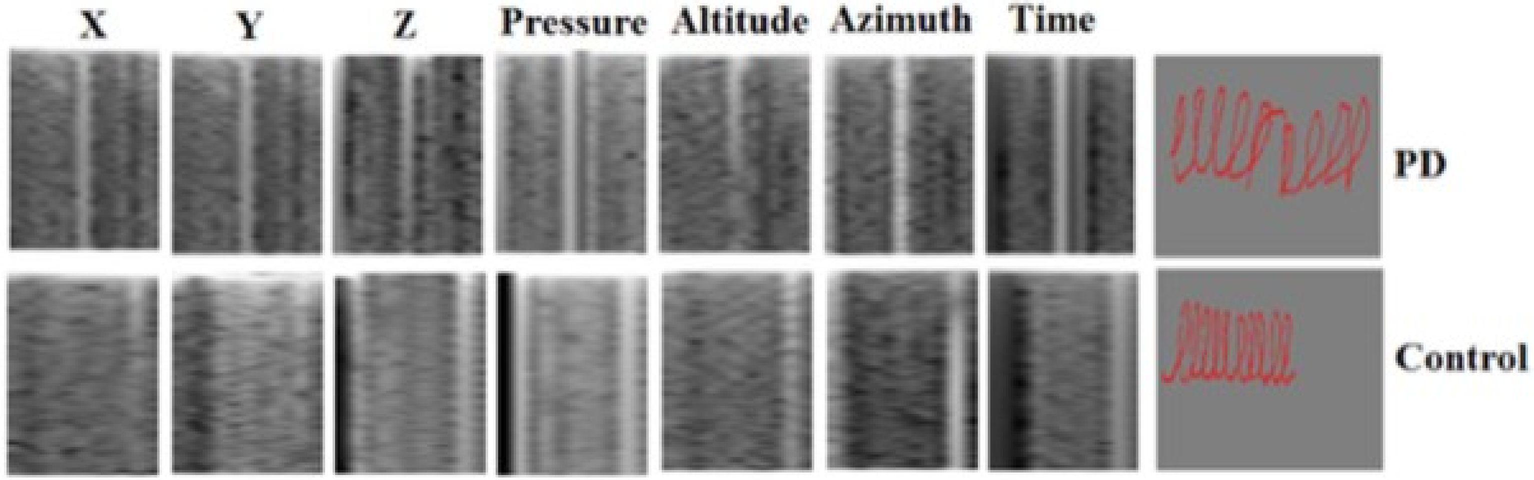


Fig. 3 Time series-based images of PD (top) and control (bottom) subjects for the 7 tasks

The spectrogram approach

- **A spectrogram is a visual representation of the frequency content of a signal over time.**
- Time-frequency representations method is specified to analyze signals where Short Time Fourier Transforms (STFT) are computed on sliding windows of the signal.
- The time-frequency resolution depends on the window size and type.
- **In this study blackman windowing with window length 256 and overlapping rate 50% provide the best spectrogram resolution.**
- When treating a spectrogram like an image, the number of frequencies and the number of time bins in the spectrogram refer to the height and the width of the image in pixels.
- The numerical “brightness” value of each pixel is then equal to the output value of the spectrogram.
- These values are converted to a logarithmic scale then normalized to [0, 1] generating a grayscale image.
- 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, to do that Lanczos technique is used to resize the spectrogram images to size 64 × 64.**

Fig. 4 The spectrograms of PD and control subjects in Task 1 for the 7 signals



The CNN model

- The overall architecture consists of 2 main parts, the feature extractor and the classifier.
- The feature extractor layers consist of two convolution layers, each followed by Relu activation function, and two pooling layers.
- The convolutional layers employ kernels of size 5 × 5 with stride of 1 pixel, and the maxpooling operations are applied on regions of size 2 × 2, with stride 2.
- The convolutional layers convert the 64 × 64 pixel input image into 64 feature maps of size 16 × 16.
- the outputs of the convolution layers are flattened, then concatenated and fed into densely connected layer to make a prediction.
- The number of input images is a hyper-parameter k, ($1 \leq k \leq 7$).
- This CNN model can be used for classification from a single image including k measurements (time-series based), or classification from k measurements, (i.e. k images).

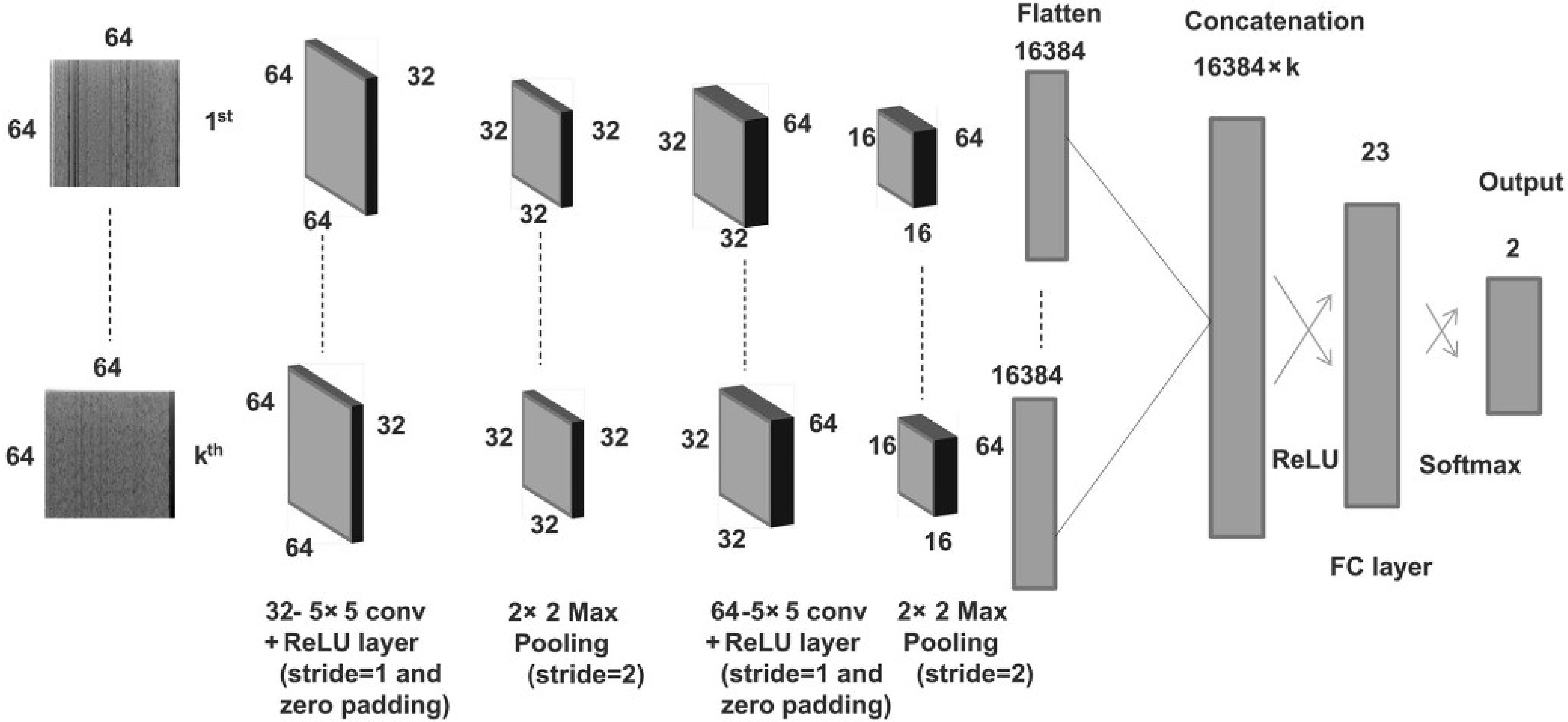


Fig. 5 Single-task CNN architecture with input k 2D representations of 1D time series

The CNN-BLSTM model

- The CNN-BLSTM architecture consists in using CNN layers for feature extraction 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.
- 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 BLSTM. The number of input time series k is a hyper-parameter ($1 \leq k \leq 7$).

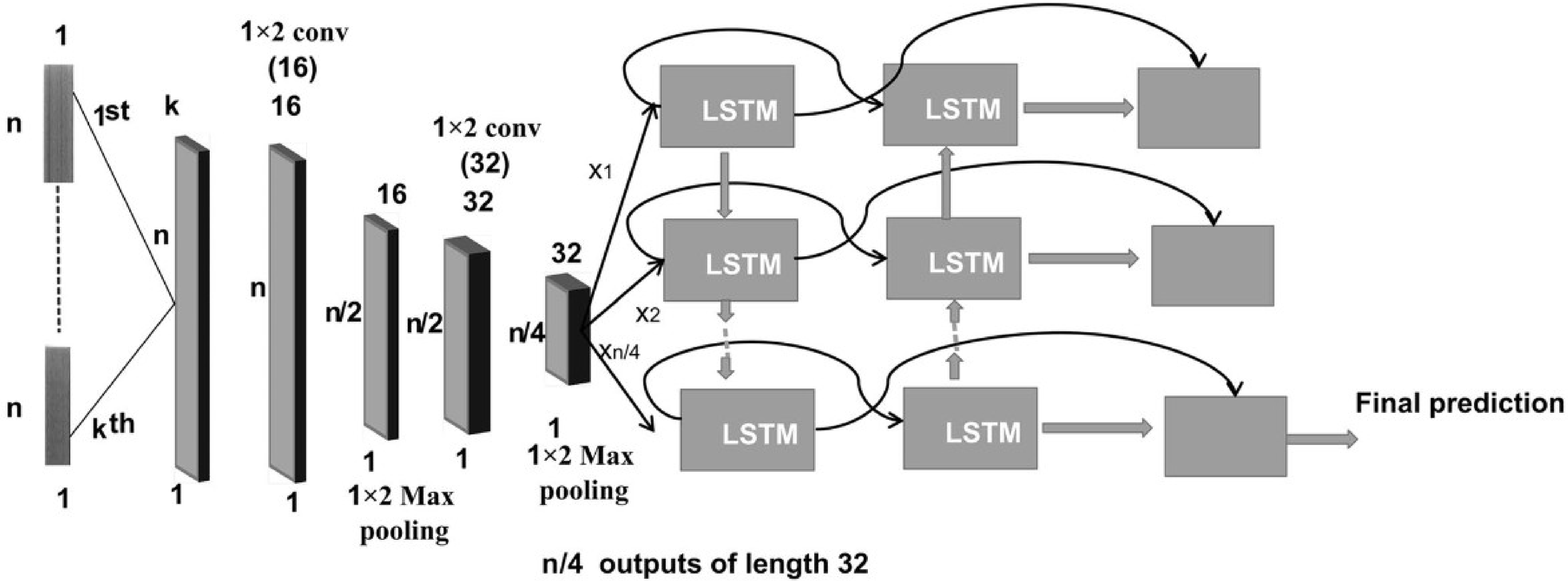


Fig. 6 Single-task CNN-BLSTM architecture on multivariate time series

Strategies to Avoid Overfitting

- In this paper the best deep learning model for time series classification found in previous work is selected with the best features combination (among features x, y, z, pressure, etc…). Based on this result, transfer learning and data augmentation approaches are applied to avoid the overfitting caused by the limited number of patient data in this study.

Transfer learning

- Transfer learning is a machine learning technique where a model trained on one task (a source domain) is re-purposed on a second related task.
- Here the CNN model is trained on a larger handwriting dataset, namely the PaHaW.
- To match the two datasets task 8 in PaHaW is eliminated since PDMultiMC has only 7 tasks and the Z coordinate feature in PDMultiMC is also eliminated.

Different transfer learning freezing strategies were studied and compared to validate the gains of transfer learning over training the CNN model from scratch, as follows:

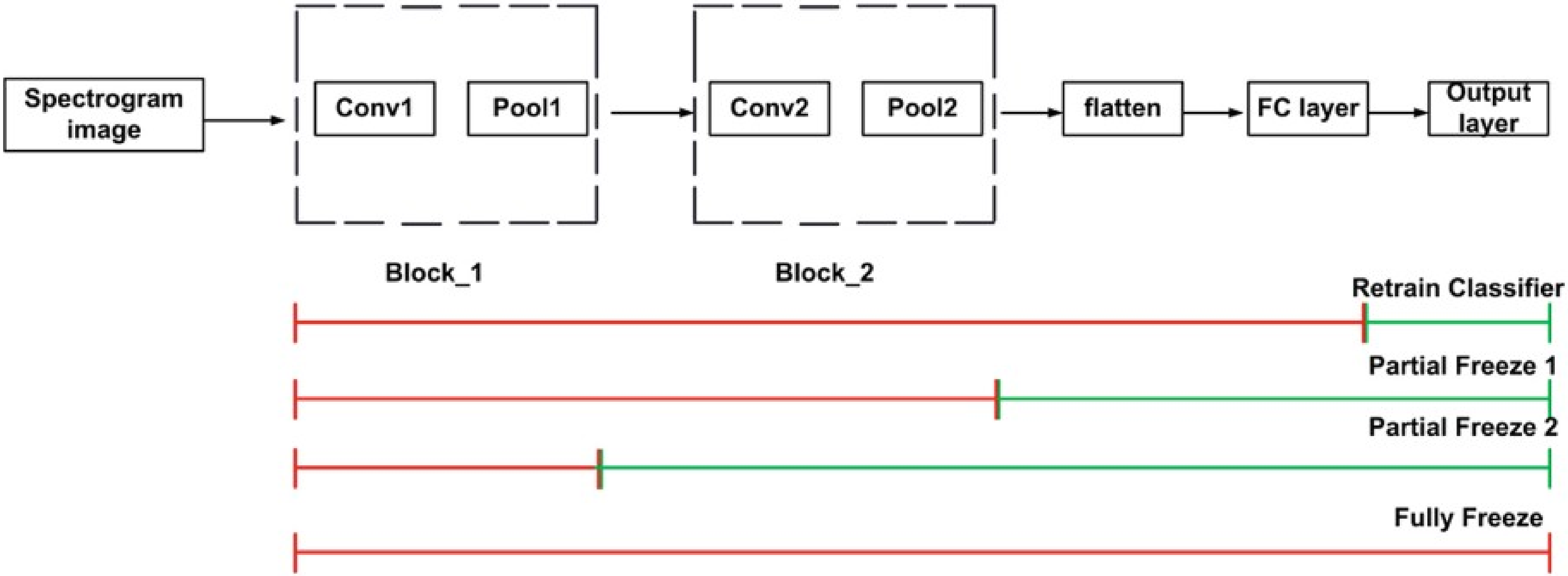


Fig. 4. Different transfer learning strategies are studied. Green indicates that blocks are retrained, and red indicates that blocks are frozen. (Color figure online)

Data augmentation applied to time series

- Data augmentation is a technique used to artificially increase the size of a training dataset by applying various transformations to the existing data. It is commonly used in deep learning to overcome the limitations of having limited labeled training data.
- The following techniques are used: Jittering, scaling, Time-Warping, and synthetic data generation.

Jittering

- Jittering is a way of simulating additive sensor noise.
- In this study Gaussian noise with a mean μ = 0 was added to each feature time series of the original training data.
- Different values for noise intensity (standard deviation (STD)) and the augmented multiple (m) were studied.

Scaling

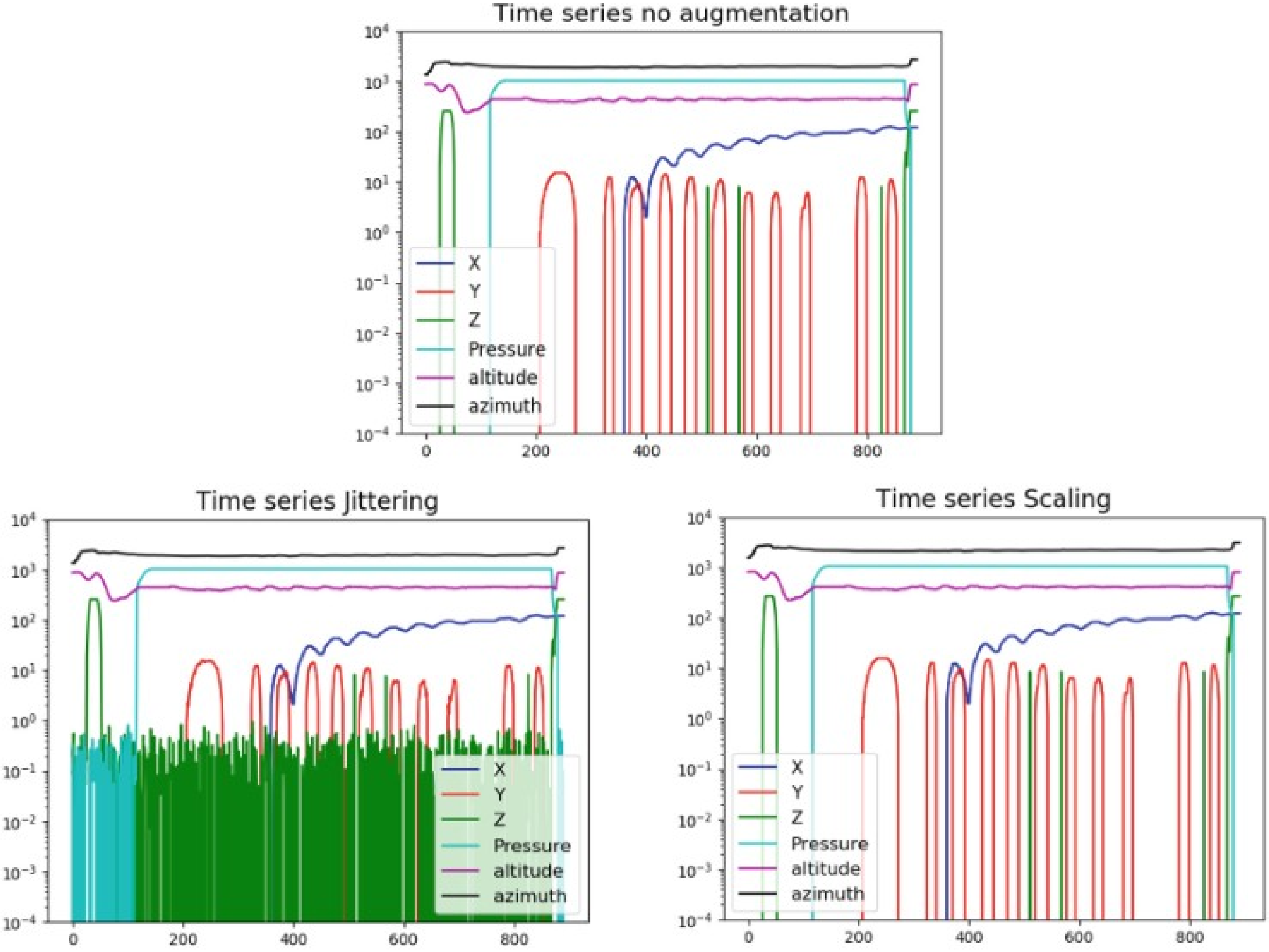
- Scaling changes the magnitude of the data in a window by multiplying by a random scalar. It is considered as a way of simulating multiplicative sensor noise.
- Gaussian noise multiplication (with a non-zero mean) was also performed to each feature time series of the original training data.
- Different values of m and STD are studied.

Time Warping

- Time-warping is a way to perturb the temporal location by smoothly distorting the time intervals between sample.

Generating synthetic data

- To create the synthetic time series, some authors propose to average a set of time series and to use the averaged time series as a newly created example.
- In this work, time series are of variable lengths.
- First of all the training data will be separated into subsets of the same class label, then the size of each subset is calculated and the maximum is selected and defined by G.
- The number of synthetic data per class Sl is equal to 2G-Hl; where Hl refers to the size of class l.
- The next step is to assign the weights for each subset separately based on the following steps: starting with a random initial time series chosen from the subset, it is assigned with a weight equal to 0.5.
- Then the 5 nearest neighbors using the Dynamic Time Warping (DTW) distance are searched. Randomly 2 out of these 5 neighbors are selected and assigned with a weight equal to 0.15 each.
- Therefore, in order to have a normalized sum of weights, the rest of time series in the subset will share the rest of the weight 0.2.
- The new generated time series length is equal to the initial time series chosen.


Time series no augmentation

Time series Jittering

Time series Scaling

Fig. 5. Raw input time series and time series obtained by various data augmentation approaches such as: jittering, scaling, time-warping, and synthetic data generation.

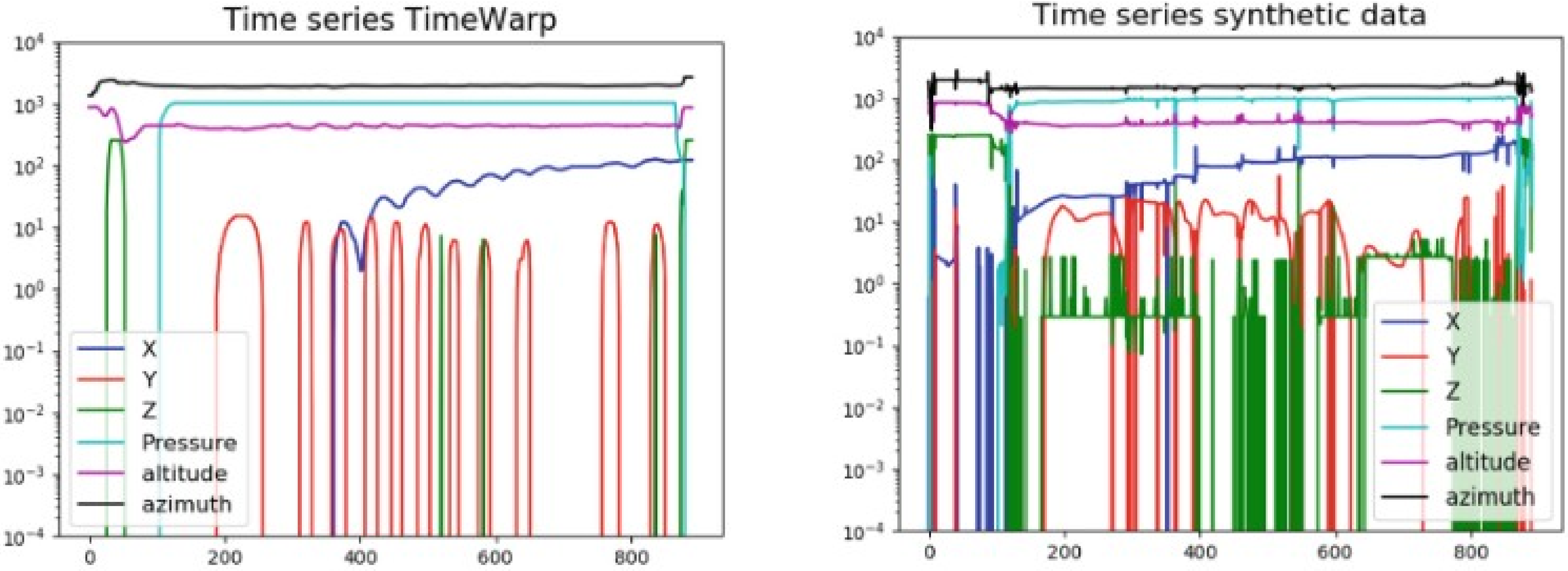


Fig. 5. Raw input time series and time series obtained by various data augmentation approaches such as: jittering, scaling, time-warping, and synthetic data generation.

Combination approach

- Single assessment and combined assessment.
- In single assessment each task is evaluated separately.
- In the combined assessment the outputs of the 7 models are combined to find the final label (also called overall performance). Each model corresponds to one task.
- Each model outputs two values corresponding to the probabilities that the input time series, associated to the given task, are performed by a parkinsonian or a control subject respectively.

- For the transfer learning approach, maximum voting will be used to obtain the final label.
- For data augmentation, a multi-layer perceptron (MLP) model that combines the probability vectors provided by the 7 models will be used instead of majority voting.
- The MLP model is composed of an input layer of 14 nodes, a single hidden layer of 40 nodes with Rectified Linear Unit (ReLU) activation function, and 2 output nodes (corresponding to PD and control) with softmax activation function.

The experiment

- In order to asses the performance of the CNN and CNN-BLSTM models:
- Different parameter values (STD and m) are applied and compared for data augmentation.
- For jittering a STD value sampled from a Gaussian distribution with 0.3 STD.
- For scaling, a random scalar is sampled from a Gaussian distribution with a mean of 1 and 0.1 STD.
- For Time-Warping, random sinusoidal curves are generated using arbitrary amplitude, frequency, and phase value.
- For data augmentation, the best accuracy was achieved when the training data is augmented to 2 times.
- 3-folds cross validation (CV) with stratified sampling method was applied in order to insure the same class distribution and number of samples in all the folds.
- The performance measures in the following table are compiled from the average of the 3 runs of the 3-fold cross validation.

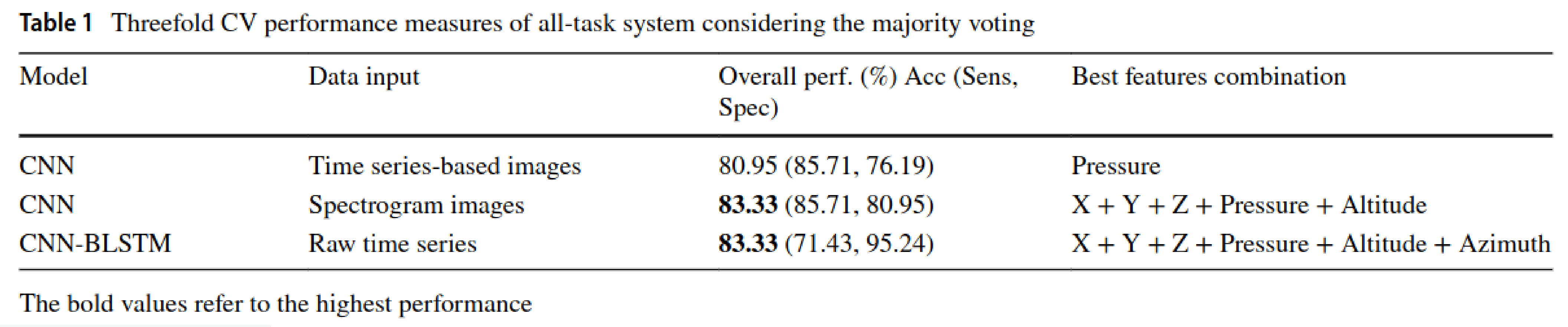


Table 1 Threefold CV performance measures of all-task system considering the majority voting

Model	Data input	Overall perf. (%) Acc (Sens, Spec)	Best features combination
CNN	Time series-based images	80.95 (85.71, 76.19)	Pressure
CNN	Spectrogram images	**83.33** (85.71, 80.95)	X + Y + Z + Pressure + Altitude
CNN-BLSTM	Raw time series	**83.33** (71.43, 95.24)	X + Y + Z + Pressure + Altitude + Azimuth

The bold values refer to the highest performance

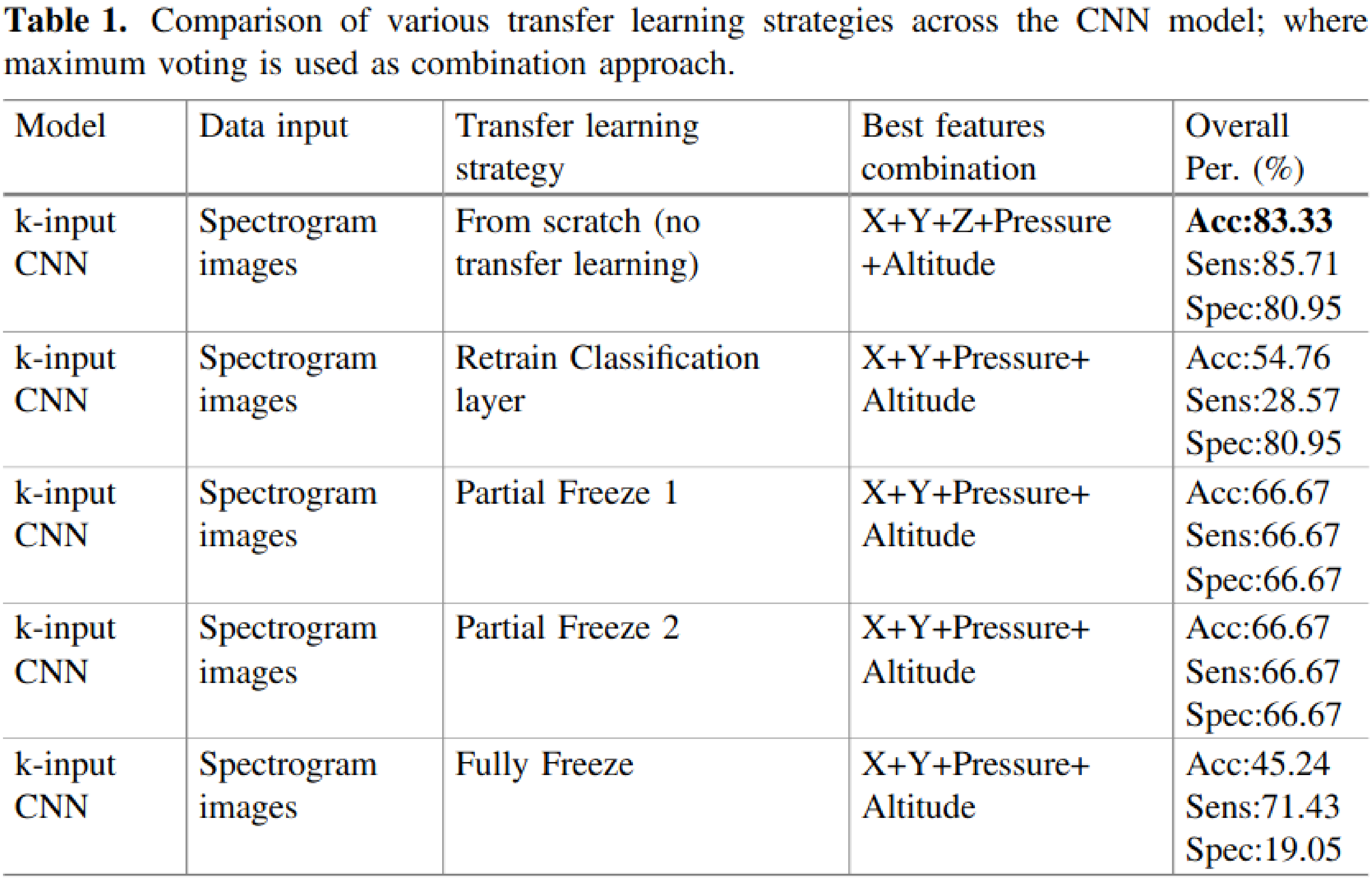


Table 1. Comparison of various transfer learning strategies across the CNN model; where maximum voting is used as combination approach.

Model	Data input	Transfer learning strategy	Best features combination	Overall Per. (%)
k-input CNN	Spectrogram images	From scratch (no transfer learning)	X+Y+Z+Pressure +Altitude	**Acc:83.33** Sens:85.71 Spec:80.95
k-input CNN	Spectrogram images	Retrain Classification layer	X+Y+Pressure+ Altitude	Acc:54.76 Sens:28.57 Spec:80.95
k-input CNN	Spectrogram images	Partial Freeze 1	X+Y+Pressure+ Altitude	Acc:66.67 Sens:66.67 Spec:66.67
k-input CNN	Spectrogram images	Partial Freeze 2	X+Y+Pressure+ Altitude	Acc:66.67 Sens:66.67 Spec:66.67
k-input CNN	Spectrogram images	Fully Freeze	X+Y+Pressure+ Altitude	Acc:45.24 Sens:71.43 Spec:19.05

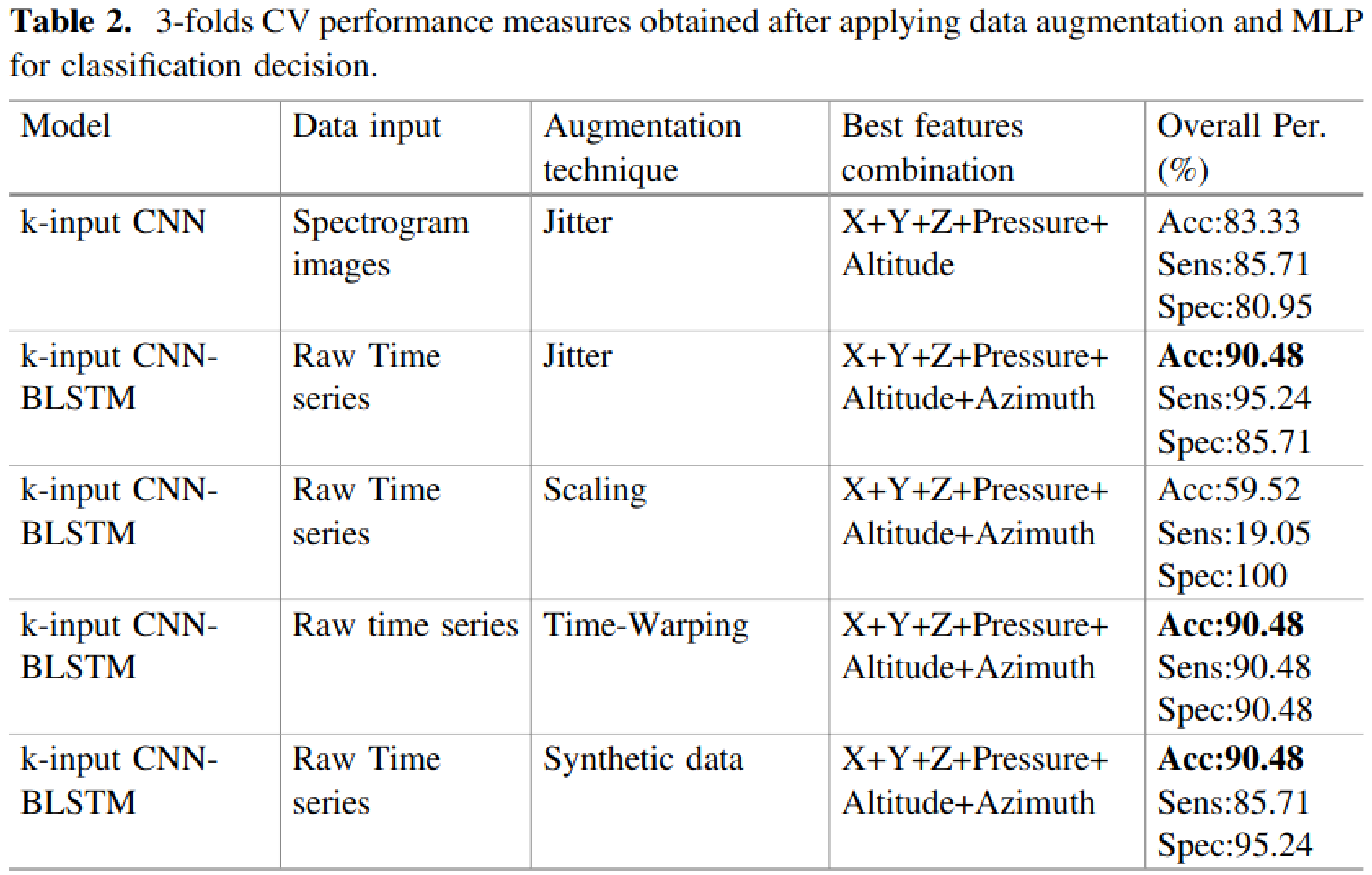


Table 2. 3-folds CV performance measures obtained after applying data augmentation and MLP for classification decision.

Model	Data input	Augmentation technique	Best features combination	Overall Per. (%)
k-input CNN	Spectrogram images	Jitter	X+Y+Z+Pressure+ Altitude	Acc:83.33 Sens:85.71 Spec:80.95
k-input CNN-BLSTM	Raw Time series	Jitter	X+Y+Z+Pressure+ Altitude+Azimuth	**Acc:90.48** Sens:95.24 Spec:85.71
k-input CNN-BLSTM	Raw Time series	Scaling	X+Y+Z+Pressure+ Altitude+Azimuth	Acc:59.52 Sens:19.05 Spec:100
k-input CNN-BLSTM	Raw time series	Time-Warping	X+Y+Z+Pressure+ Altitude+Azimuth	**Acc:90.48** Sens:90.48 Spec:90.48
k-input CNN-BLSTM	Raw Time series	Synthetic data	X+Y+Z+Pressure+ Altitude+Azimuth	**Acc:90.48** Sens:85.71 Spec:95.24

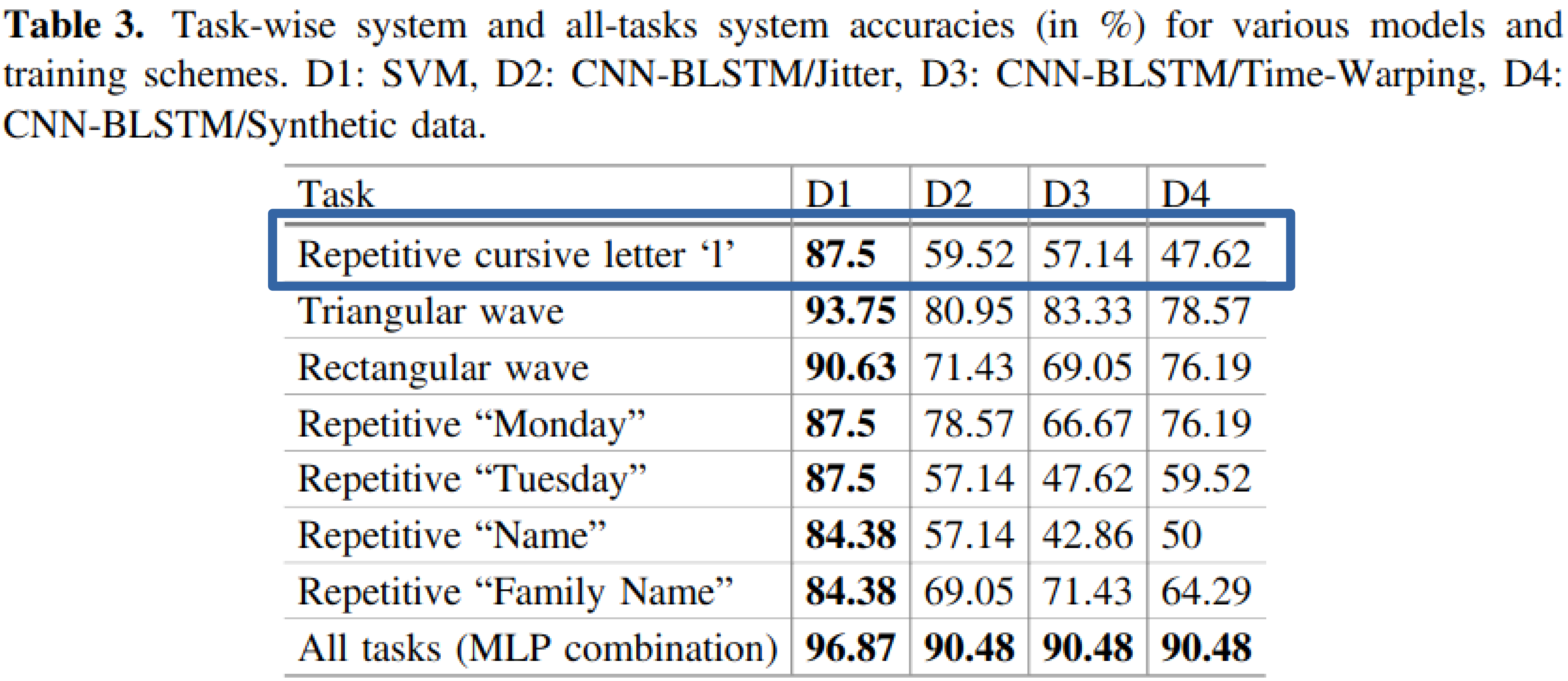


Table 3. Task-wise system and all-tasks system accuracies (in %) for various models and training schemes. D1: SVM, D2: CNN-BLSTM/Jitter, D3: CNN-BLSTM/Time-Warping, D4: CNN-BLSTM/Synthetic data.

Task	D1	D2	D3	D4
Repetitive cursive letter ‘l’	**87.5**	59.52	57.14	47.62
Triangular wave	**93.75**	80.95	83.33	78.57
Rectangular wave	**90.63**	71.43	69.05	76.19
Repetitive “Monday”	**87.5**	78.57	66.67	76.19
Repetitive “Tuesday”	**87.5**	57.14	47.62	59.52
Repetitive “Name”	**84.38**	57.14	42.86	50
Repetitive “Family Name”	**84.38**	69.05	71.43	64.29
All tasks (MLP combination)	**96.87**	**90.48**	**90.48**	**90.48**

Conclusion

- Automatic classification system for Parkinson's disease (PD) detection based on online handwriting.
- Two end-to-end time series classification models proposed: CNN and CNN-BLSTM.
- Deep learning models require a large number of training samples, which is challenging due to limited data availability for PD classification.
- Two main approaches used to cope with limited data: transfer learning with CNN and data augmentation techniques.
- Transfer learning showed limited gains, likely due to the absence of Z coordinate feature in the dataset.
- Data augmentation techniques used: jittering, scaling, Time-Warping, and synthetic data generation.
- CNN-BLSTM model combined with jittering and synthetic data augmentation achieved an improved accuracy of 97.62% from 83.33% for all tasks.
- Observations and conclusions: importance of Z coordinate feature, effectiveness of data augmentation over transfer learning, Time-Warping did not improve PD classification, data augmentation boosts deep learning model performance without converting time series to images.
- The techniques proposed in this study proved to be highly accurate for the ensemble of tasks, however, in single-task classification it performed very poorly for all the available tasks (the l task is the main concern in our case).

Performance-driven Handwriting Task Selection for Parkinson’s Disease Classification

Maria Angelillo et al, 2019

Introduction

- Goal: Investigate the potential of an optimal subset of tasks for a more accurate Parkinson classification.
- Results: The proposed approach improves the baseline results on the PaHaW dataset.

The PaHaW Dataset

- The “Parkinson’s disease handwriting database” (PaHaW) collects hand-writing data of Participants were enrolled at the First Department of Neurology, Masaryk University, and the St. Anne’s University Hospital, Brno, Czech Republic :
- 37 PD patients and 38 age and gender-matched healthy controls (HC).
- Right-handed.
- At least 10 years of education.
- Czech as their native language. No significant between-group difference regarding age or gender was found.
- No history or presence of any psychiatric symptom or disease affecting the central nervous system, with the exception of Parkinsonism in the PD group.
- Patients were only examined in their ON-state while taking dopaminergic medication.
- PDs were evaluated by a qualified neurologist.
- HCs underwent a thorough examination to ensure that no movement disorder or injury could have significantly affected handwriting.

The tasks performed :

- Drawing an Archimedes spiral
- Writing in cursive the letter ***l***
- The biagram ***le***
- The triagram ***les***
- Writing in cursive the word ***lektorka*** (“female teacher” in Czech)
- ***porovnat*** (“to compare”)
- ***nepopadnout*** (“to not catch”)
- Writing in cursive the sentence ***Tramvaj dnes ûz nepojede*** (“The tram won’t go today”).

- The handwriting signals were recorded using a Wacom Intuos digitizing tablet, overlaid with a blank sheet of paper. Like many other professional tablets, the raw data acquired are the x- and y-coordinates of the pen tip, the corresponding time stamps, measures of pen inclination, i.e. tilt-x and tilt-y, and pen pressure. The button status is also available, which is a binary variable with value 0 for pen-ups (“in-air movement”) and 1 for pen-downs (“on-surface movement”). The sampling rate was 200 samples per second.
- Since not all participants completed each task, we considered only those subjects who completed each of the eight tasks, i.e. 36 PD and 36 HC.

Feature extraction

- The feature calculation stage resulted in either a single value or a vector feature.
- For all the resulting vector features the following basic statistical measures were computed: mean; median; standard deviation; 1st percentile; 99th percentile; 99th percentile – 1st percentile, which is an outlier robust range.
- All features were normalized before classification so as to have zero mean and unit variance.

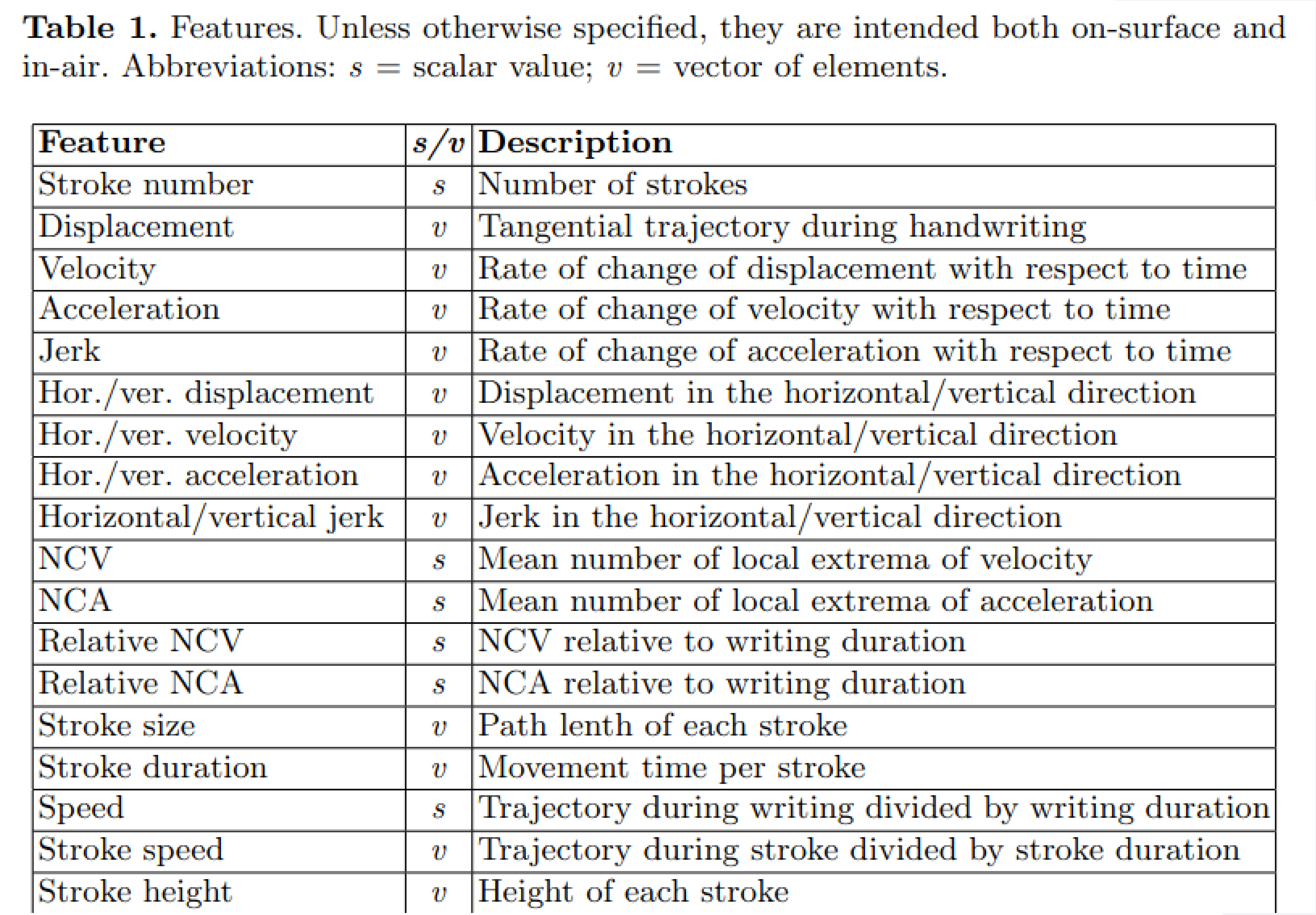
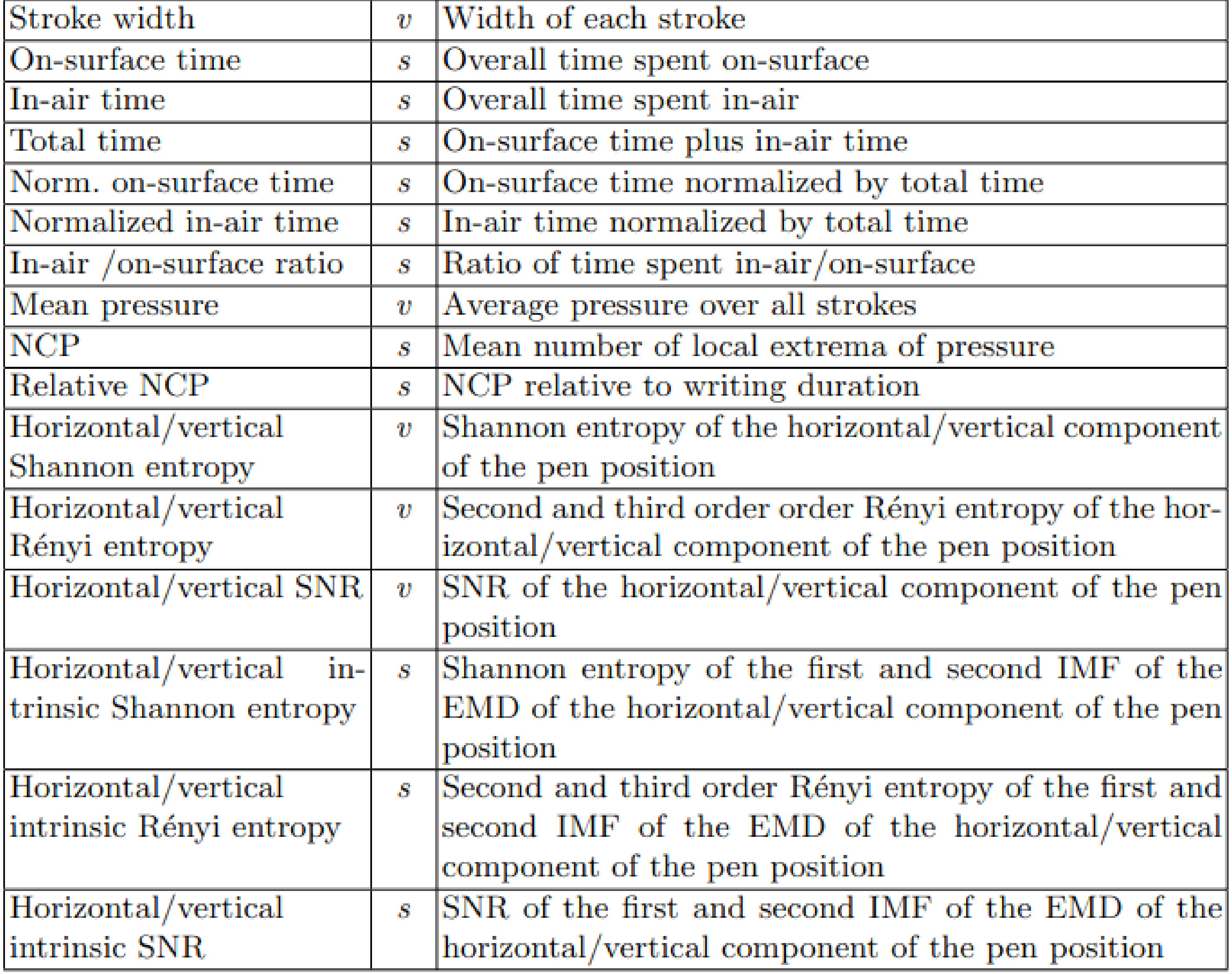


Table 1. Features. Unless otherwise specified, they are intended both on-surface and in-air. Abbreviations: s = scalar value; v = vector of elements.

Feature	s/v	**Description**
Stroke number	s	Number of strokes
Displacement	v	Tangential trajectory during handwriting
Velocity	v	Rate of change of displacement with respect to time
Acceleration	v	Rate of change of velocity with respect to time
Jerk	v	Rate of change of acceleration with respect to time
Hor./ver. displacement	v	Displacement in the horizontal/vertical direction
Hor./ver. velocity	v	Velocity in the horizontal/vertical direction
Hor./ver. acceleration	v	Acceleration in the horizontal/vertical direction
Horizontal/vertical jerk	v	Jerk in the horizontal/vertical direction
NCV	s	Mean number of local extrema of velocity
NCA	s	Mean number of local extrema of acceleration
Relative NCV	s	NCV relative to writing duration
Relative NCA	s	NCA relative to writing duration
Stroke size	v	Path lenth of each stroke
Stroke duration	v	Movement time per stroke
Speed	s	Trajectory during writing divided by writing duration
Stroke speed	v	Trajectory during stroke divided by stroke duration
Stroke height	v	Height of each stroke



Stroke width	v	Width of each stroke
On-surface time	s	Overall time spent on-surface
In-air time	s	Overall time spent in-air
Total time	s	On-surface time plus in-air time
Norm. on-surface time	s	On-surface time normalized by total time
Normalized in-air time	s	In-air time normalized by total time
In-air /on-surface ratio	s	Ratio of time spent in-air/on-surface
Mean pressure	v	Average pressure over all strokes
NCP	s	Mean number of local extrema of pressure
Relative NCP	s	NCP relative to writing duration
Horizontal/vertical Shannon entropy	v	Shannon entropy of the horizontal/vertical component of the pen position
Horizontal/vertical Rényi entropy	v	Second and third order order Rényi entropy of the horizontal/vertical component of the pen position
Horizontal/vertical SNR	v	SNR of the horizontal/vertical component of the pen position
Horizontal/vertical intrinsic Shannon entropy	s	Shannon entropy of the first and second IMF of the EMD of the horizontal/vertical component of the pen position
Horizontal/vertical intrinsic Rényi entropy	s	Second and third order Rényi entropy of the first and second IMF of the EMD of the horizontal/vertical component of the pen position
Horizontal/vertical intrinsic SNR	s	SNR of the first and second IMF of the EMD of the horizontal/vertical component of the pen position

Model fitting

Support Vector Machines:

- In this work, a linear as well as a radial basis function (RBF) kernel were considered.
- $C = 1$ and $\gamma = 1n$ where n is the number of features.

Logistic regression:

- Dual formulation with L2 regularization to avoid overfitting.

Linear Discriminant Analysis LDA:

- Automatic shrinkage using the Ledoit-Wolf lemma was employed to counter the fact that the number of training examples was small compared to the number of features.

AdaBoost ADA:

- In the present paper, 500 decision trees were used as base learners.

Ensemble:

- The individual classifiers are combined by using a voting scheme to predict the class labels;
- In this way, the individual weaknesses of each single classifier are balanced and mitigated.
- A majority voting scheme was used.
- Each classification model was trained on each task individually and the performance obtained were evaluated.
- This served to explore the most discriminant tasks among the eight originally proposed.
- Then, the three best tasks, i.e. those with the highest prediction accuracy, were pooled together in an ensemble scheme whose predictions were finally obtained via majority voting.

Validation

- The classification performance was validated through a stratified 10-fold cross-validation.

Feature selection

- The discriminating power of each feature was evaluated by considering its accuracy in separating PD from HC when used as a single input feature to a linear SVM classifier.
- All features were then ranked in accordance with this score and only the N features providing the highest score were retained for the final model fitting.
- N was not fixed, a dynamic threshold for N is established depending on the ranking of features for each specific task.

The experiment

Combining tasks:

- The features coming from each task were combined into a unique high dimensional feature vector.
- Only the best mean accuracy, averaged over all the cross-validation iterations, is reported.

Individual tasks:

- In order to evaluate the predictive potential of each task individually, a classification model was considered for each of them.

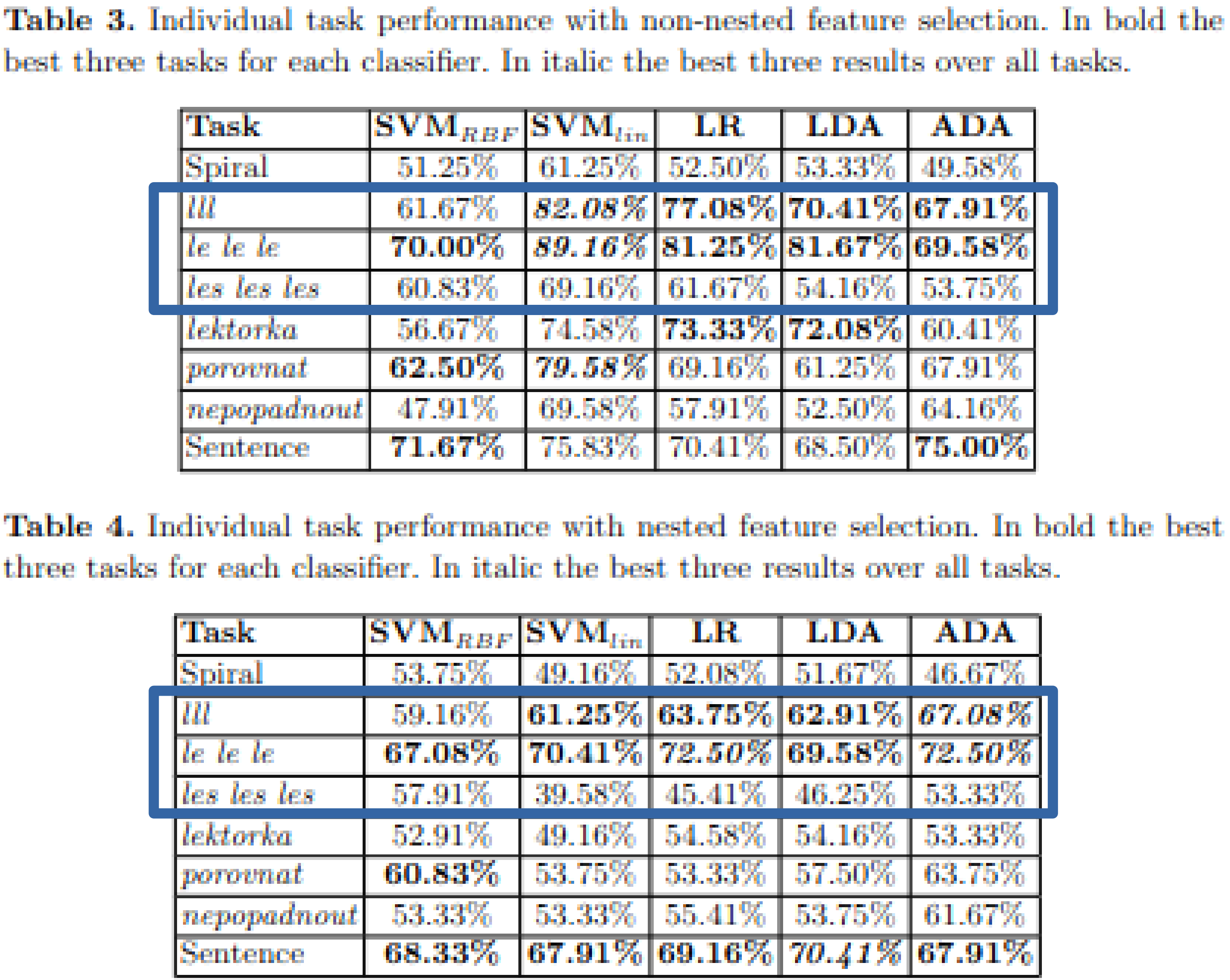


Table 3. Individual task performance with non-nested feature selection. In bold the best three tasks for each classifier. In italic the best three results over all tasks.

Task	**SVM$_{RBF}$**	**SVM$_{lin}$**	**LR**	**LDA**	**ADA**
Spiral	51.25%	61.25%	52.50%	53.33%	49.58%
lll	61.67%	***82.08%***	**77.08%**	**70.41%**	**67.91%**
le le le	**70.00%**	***89.16%***	**81.25%**	**81.67%**	**69.58%**
les les les	60.83%	69.16%	61.67%	54.16%	53.75%
lektorka	56.67%	74.58%	**73.33%**	**72.08%**	60.41%
porovnat	**62.50%**	***79.58%***	69.16%	61.25%	67.91%
nepopadnout	47.91%	69.58%	57.91%	52.50%	64.16%
Sentence	**71.67%**	75.83%	70.41%	68.50%	**75.00%**

Table 4. Individual task performance with nested feature selection. In bold the best three tasks for each classifier. In italic the best three results over all tasks.

Task	**SVM$_{RBF}$**	**SVM$_{lin}$**	**LR**	**LDA**	**ADA**
Spiral	53.75%	49.16%	52.08%	51.67%	46.67%
lll	59.16%	**61.25%**	**63.75%**	**62.91%**	***67.08%***
le le le	**67.08%**	**70.41%**	***72.50%***	**69.58%**	***72.50%***
les les les	57.91%	39.58%	45.41%	46.25%	53.33%
lektorka	52.91%	49.16%	54.58%	54.16%	53.33%
porovnat	**60.83%**	53.75%	53.33%	57.50%	63.75%
nepopadnout	53.33%	53.33%	55.41%	53.75%	61.67%
Sentence	**68.33%**	**67.91%**	**69.16%**	***70.41%***	**67.91%**

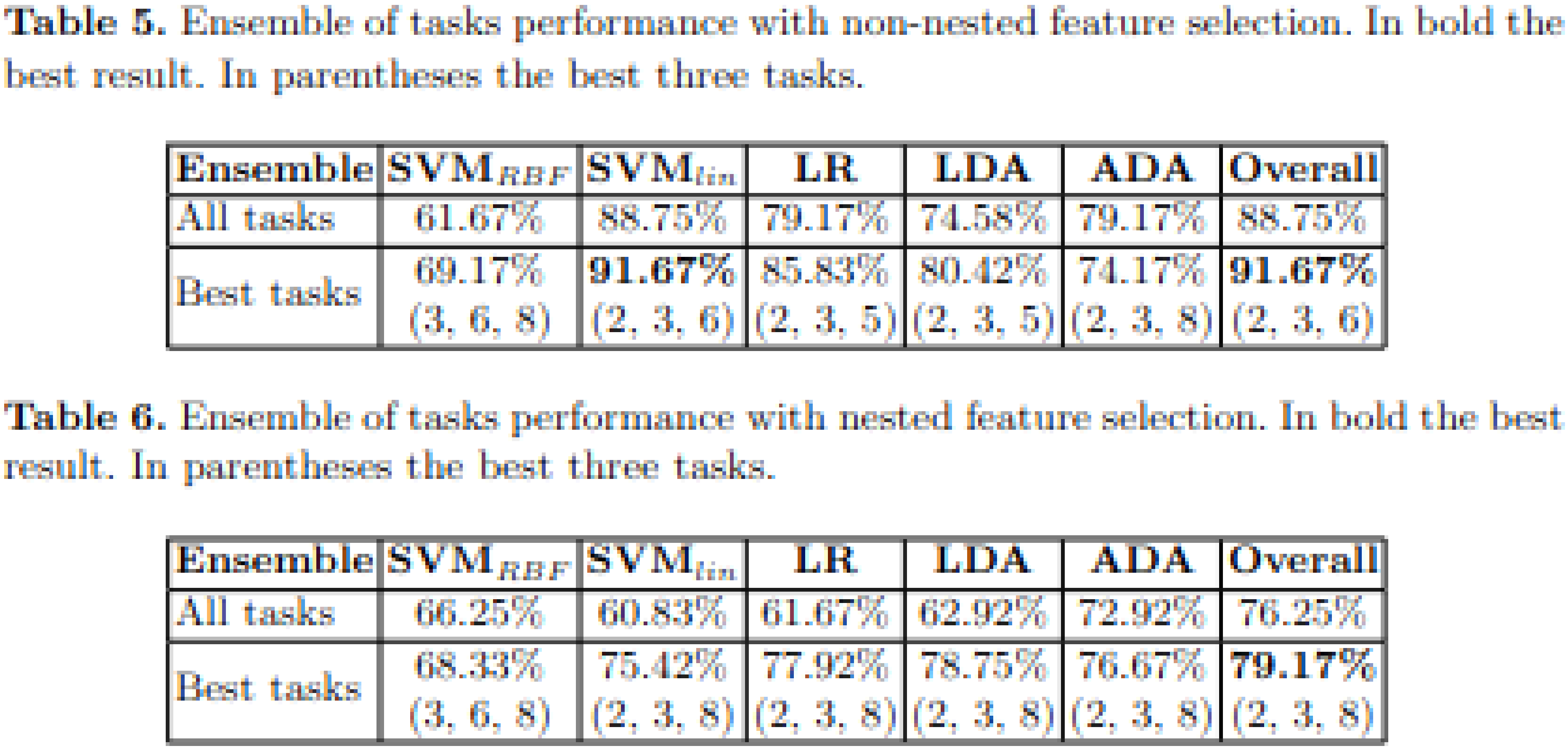


Table 5. Ensemble of tasks performance with non-nested feature selection. In bold the best result. In parentheses the best three tasks.

Ensemble	SVM_{RBF}	SVM_{lin}	LR	LDA	ADA	Overall
All tasks	61.67%	88.75%	79.17%	74.58%	79.17%	88.75%
Best tasks	69.17% (3, 6, 8)	**91.67%** (2, 3, 6)	85.83% (2, 3, 5)	80.42% (2, 3, 5)	74.17% (2, 3, 8)	**91.67%** (2, 3, 6)

Table 6. Ensemble of tasks performance with nested feature selection. In bold the best result. In parentheses the best three tasks.

Ensemble	SVM_{RBF}	SVM_{lin}	LR	LDA	ADA	Overall
All tasks	66.25%	60.83%	61.67%	62.92%	72.92%	76.25%
Best tasks	68.33% (3, 6, 8)	75.42% (2, 3, 8)	77.92% (2, 3, 8)	78.75% (2, 3, 8)	76.67% (2, 3, 8)	**79.17%** (2, 3, 8)

Conclusion

In this study, multiple machine learning models were used after feature extraction on the PaHaW dataset in order to find the best performing tasks when classifying PDs and HCs.

The most important discovery was the importance of loop like exercices in the discrimination between HCs and Pds.

Which is most likely due to the lack of fine motor control when it comes to circular motion in handwriting for PD patients.

Ensemble of convolutional neural networks for Parkinson’s disease diagnosis from offline handwriting

Gazda Matej et al, 2022

Introduction

- Goal: This paper proposes the ensemble of deep convolutional neural networks for diagnosing Parkinson’s disease from offline handwriting.
- Results: The experimental results on two handwriting datasets showed that the proposed approach currently provides the highest classification accuracy compared to other strategies for diagnosing Parkinson’s disease based on offline handwriting.

Proposed approach

- A majority voting ensemble method of five convolutional neural networks trained independently was applied.
- Since both PaHaW and NewHandPD are small datasets, transfer learning (TL) was imployed for better generalization.
- TL was divided into two categories. (a) without mediator dataset, and (b) with mediator dataset.

TL with and without mediation

- TL without mediator dataset:

  The network is trained by end-to-end approach on large scale source dataset S and then fine-tuned on the target task – diagnostics of Parkinson’s disease on PaHaW/NewHandPD datasets. These networks are denoted as CNN$_{S}$.

- TL with a mediator set:

  The network is trained on a large source dataset S that is far from the task at hand, then fine-tuned on a dataset A, closer to the final dataset, and finally fine-tuned to the target task. these networks are denoted as CNN$_{S,A}$.

The datasets

- **ImageNet** is a large-scale image database that serves as a benchmark for training and evaluating computer vision models. It contains millions of labeled images belonging to thousands of different categories or classes.
- **MNIST** (Modified National Institute of Standards and Technology) is a widely-used dataset in machine learning and computer vision. It consists of 28x28 grayscale images of handwritten digits from 0 to 9. The dataset contains 60,000 training images and 10,000 test images, making it a standard benchmark for developing and evaluating algorithms for digit recognition tasks.
- **UJIpenchars2** is a dataset consisting of handwritten character samples commonly used for research and evaluation in pattern recognition, machine learning, and handwriting analysis. The dataset includes multiple classes of handwritten characters, each represented as grayscale images.
- **NewHandPD** and **PaHaW**.

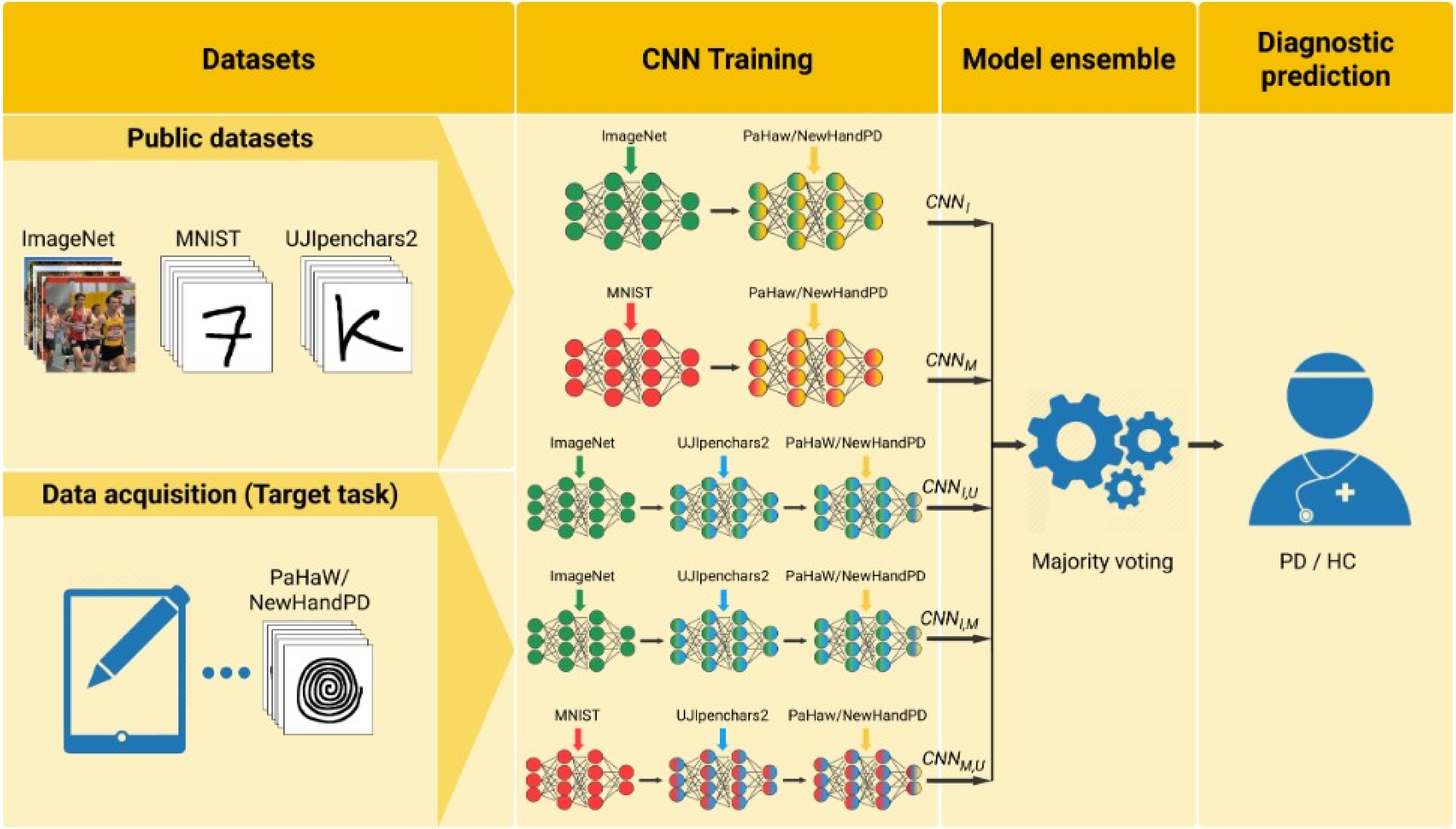


Fig. 1. Concept of the proposed decision support system incorporating MFT CNNs and ensemble voting.

Results

- The proposed ensemble approach was evaluated on two publicly available datasets: PaHaW and NewHandPD.
- In the PaHaW the sentence task is avoided since it is different from a single word and has more complex structure.
- If task contains multiple repetitions, every repetition is considered as single image and was used for training and testing as single sample.
- The VGG architecture was used for CNN, it is characterized by its simplicity and uniformity, primarily consisting of a series of convolutional layers with small 3x3 filters, followed by max-pooling layers to reduce spatial dimensions. VGG comes in different configurations, such as VGG16 and VGG19, which indicate the number of layers in the network.
- Stochastic gradient descent (SGD) was used for pre-training and training on mediator dataset and Adadelta for training on the target task.
- The learning rate was set up to value 0.01, and they used 300 epochs.
- All images were resized to 224 × 224 pixels to match ImageNet size.
- Stratified five-fold cross-validation was used while ensuring that handwriting samples from one subject were used only in the training dataset or testing dataset, and not in both.

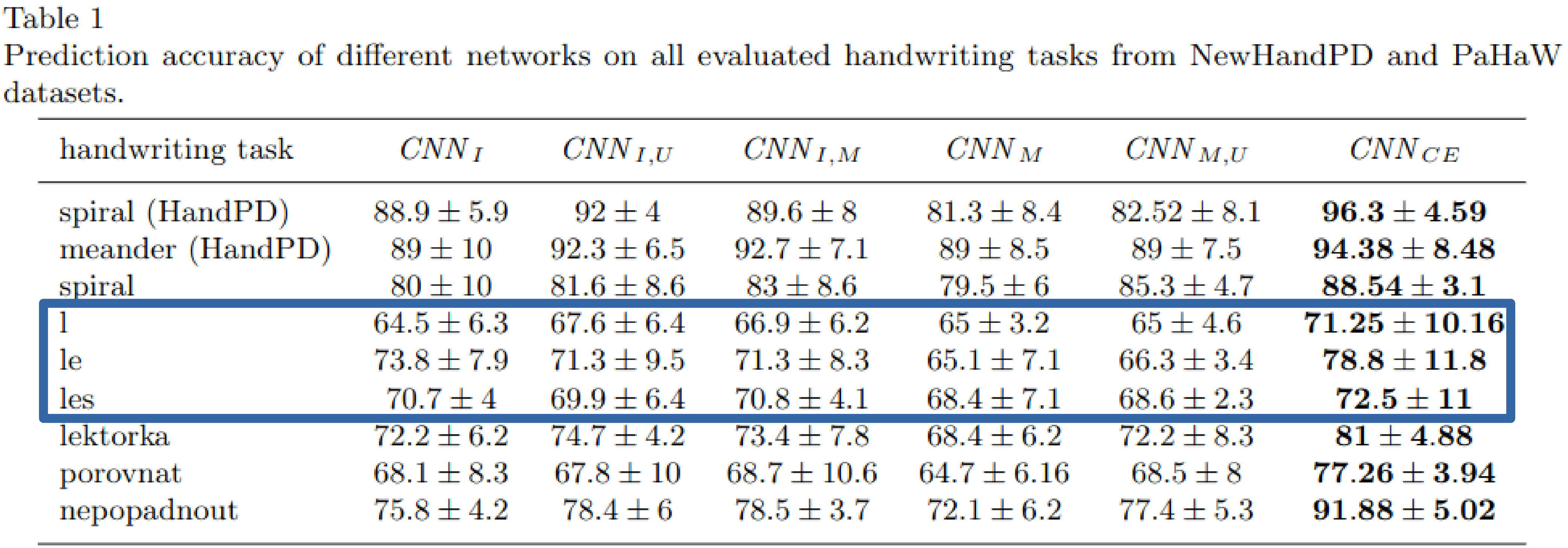


Table 1
Prediction accuracy of different networks on all evaluated handwriting tasks from NewHandPD and PaHaW datasets.

handwriting task	CNN_I	$CNN_{I,U}$	$CNN_{I,M}$	CNN_M	$CNN_{M,U}$	CNN_{CE}
spiral (HandPD)	88.9 ± 5.9	92 ± 4	89.6 ± 8	81.3 ± 8.4	82.52 ± 8.1	$\mathbf{96.3 \pm 4.59}$
meander (HandPD)	89 ± 10	92.3 ± 6.5	92.7 ± 7.1	89 ± 8.5	89 ± 7.5	$\mathbf{94.38 \pm 8.48}$
spiral	80 ± 10	81.6 ± 8.6	83 ± 8.6	79.5 ± 6	85.3 ± 4.7	$\mathbf{88.54 \pm 3.1}$
l	64.5 ± 6.3	67.6 ± 6.4	66.9 ± 6.2	65 ± 3.2	65 ± 4.6	$\mathbf{71.25 \pm 10.16}$
le	73.8 ± 7.9	71.3 ± 9.5	71.3 ± 8.3	65.1 ± 7.1	66.3 ± 3.4	$\mathbf{78.8 \pm 11.8}$
les	70.7 ± 4	69.9 ± 6.4	70.8 ± 4.1	68.4 ± 7.1	68.6 ± 2.3	$\mathbf{72.5 \pm 11}$
lektorka	72.2 ± 6.2	74.7 ± 4.2	73.4 ± 7.8	68.4 ± 6.2	72.2 ± 8.3	$\mathbf{81 \pm 4.88}$
porovnat	68.1 ± 8.3	67.8 ± 10	68.7 ± 10.6	64.7 ± 6.16	68.5 ± 8	$\mathbf{77.26 \pm 3.94}$
nepopadnout	75.8 ± 4.2	78.4 ± 6	78.5 ± 3.7	72.1 ± 6.2	77.4 ± 5.3	$\mathbf{91.88 \pm 5.02}$

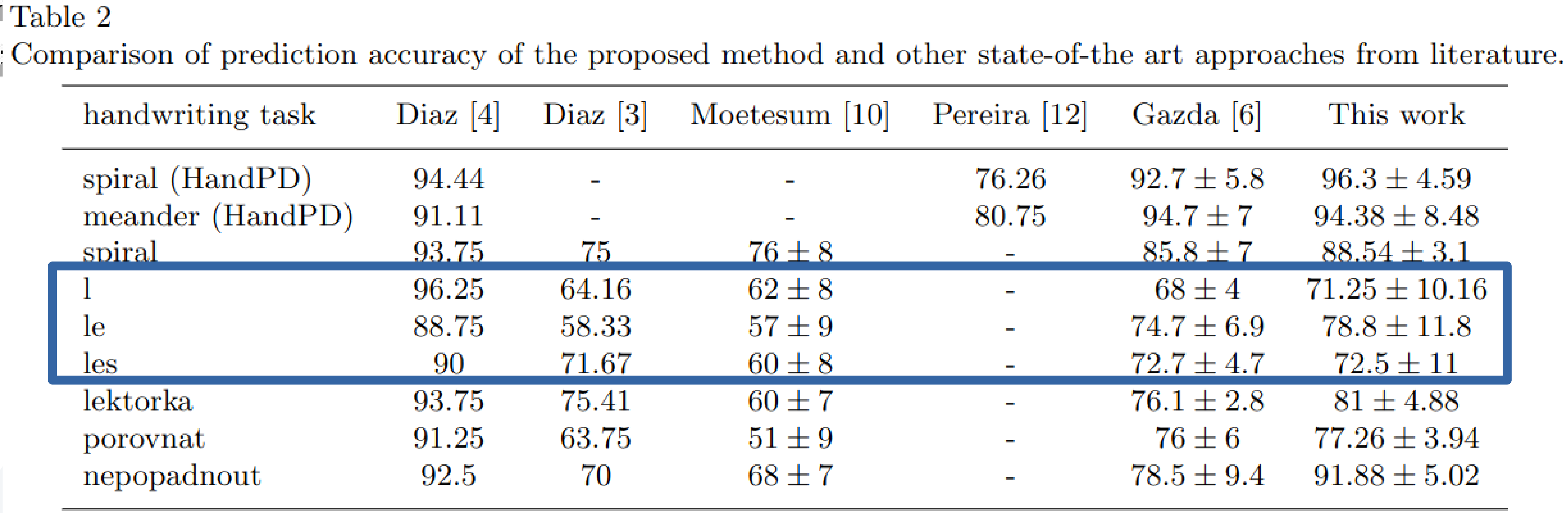


Table 2
Comparison of prediction accuracy of the proposed method and other state-of-the art approaches from literature.

handwriting task	Diaz [4]	Diaz [3]	Moetesum [10]	Pereira [12]	Gazda [6]	This work
spiral (HandPD)	94.44	-	-	76.26	92.7 ± 5.8	96.3 ± 4.59
meander (HandPD)	91.11	-	-	80.75	94.7 ± 7	94.38 ± 8.48
spiral	93.75	75	76 ± 8	-	85.8 ± 7	88.54 ± 3.1
l	96.25	64.16	62 ± 8	-	68 ± 4	71.25 ± 10.16
le	88.75	58.33	57 ± 9	-	74.7 ± 6.9	78.8 ± 11.8
les	90	71.67	60 ± 8	-	72.7 ± 4.7	72.5 ± 11
lektorka	93.75	75.41	60 ± 7	-	76.1 ± 2.8	81 ± 4.88
porovnat	91.25	63.75	51 ± 9	-	76 ± 6	77.26 ± 3.94
nepopadnout	92.5	70	68 ± 7	-	78.5 ± 9.4	91.88 ± 5.02

Diaz’s method takes advantage also of kinematic features, such as velocity and pressure, not only imagery data. Even then, we can see that the proposed CNN ensemble outperformed the Diaz’s approach on NewHandPD dataset and yielded very competitive results for word nepopadnout.

Conclusion

It is clear that this approach can not provide a complex view on handwriting as the online processing since it does not consider handwriting dynamics and kinematics but can capture significantly more data and screen a larger part of the population.

However it is noticeable that it performed much better on tasks that include much more data points than the short tasks.

Dynamic Handwriting Analysis for Supporting Earlier Parkinson’s Disease Diagnosis

Donato Impedovo et al, 2018

Introduction

- Goal: Investigate if and to which extent dynamic features of the handwriting process can support PD diagnosis at earlier stages.
- Techniques: A subset of the publicly available PaHaW dataset has been used, including those patients showing only early to mild degree of disease severity. They developed a classification framework based on different classifiers and an ensemble scheme.
- Results: Some encouraging results have been obtained; in particular, good specificity performances have been observed.

The experiment

- The PaHaW dataset was used.
- They focused only on those patients exhibiting earlier manifestations of the disease;
- The features coming from each task were combined into a single high dimensional feature vector that was fed into a single machine learning algorithm.
- The discriminating power of every single task and combinations of them was investigated, by using several machine learning algorithms and an ensemble approach.

Feature extraction

- The feature extraction stage resulted in either a single value feature or a vector feature.
- For vector features, the following basic statistical measures have been calculated:

Mean; median; standard deviation; 1st percentile; 99th percentile; 99th–1st percentile (outlier robust range).

Table 2. List of features. Unless otherwise specified, they are intended both on-surface and in-air.

Feature	*s*/*v*	Description
Stroke number	*s*	Number of strokes
Displacement	*v*	Tangential trajectory during handwriting
Velocity	*v*	Rate of change of position whit respect to time
Acceleration	*v*	Rate of change of velocity with respect to time
Jerk	*v*	Rate of change of acceleration with respect to time
Horizontal/vertical displacement	*v*	Displacement in the horizontal/vertical direction
Horizontal/vertical velocity	*v*	Velocity in the horizontal/vertical direction
Horizontal/vertical acceleration	*v*	Acceleration in the horizontal/vertical direction
Horizontal/vertical jerk	*v*	Jerk in the horizontal/vertical direction
NCV	*s*	Mean number of local extrema of velocity
NCA	*s*	Mean number of local extrema of acceleration
Relative NCV	*s*	NCV relative to writing duration
Relative NCA	*s*	NCA relative to writing duration
Stroke size	*v*	Path length of each stroke
Stroke duration	*v*	Movement time per stroke
Speed	*s*	Trajectory during handwriting divided by writing duration
Stroke speed	*v*	Trajectory during stroke divided by stroke duration
Stroke height	*v*	Height of each stroke
Stroke width	*v*	Width of each stroke
On-surface time	*s*	Overall time spent on-surface

In-air time	s	Overall time spent in-air
Total time	s	On-surface time plus in-air time
Normalized on-surface time	s	On-surface time normalized by total time
Normalized in-air time	s	In-air time normalized by total time
In-air/on-surface ratio	s	Ratio of time spent in-air/on-surface
Mean pressure	v	Average pressure over all on-surface strokes
NCP	s	Mean number of local extrema of pressure
Relative NCP	s	NCP relative to writing duration
Horizontal/vertical Shannon entropy	v	Shannon entropy of the horizontal/vertical component of the pen position
Horizontal/vertical Rényi entropy	v	Second and third order Rényi entropy of the horizontal/vertical component of the pen position
Horizontal/vertical signal-to-noise ratio	v	Signal-to-noise ratio of the horizontal/vertical component of the pen position
Horizontal/vertical intrinsic Shannon entropy	v	Shannon entropy of the first/second IMF obtained by the EMD of the horizontal/vertical component of the pen position
Horizontal/vertical intrinsic Rényi entropy	v	Second and third order Rényi entropy of the first/second IMF obtained by the EMD of the horizontal/vertical component of the pen position
Horizontal/vertical signal-to-noise ratio	v	Signal-to-noise ratio of the first/second IMF obtained by the EMD of the horizontal/vertical component of the pen position

Abbreviations: s = scalar value; v = vector of elements.

Model fitting

Some state-of-the-art supervised machine learning algorithms tailored to small datasets have been employed:

- K-Nearest Neighbours (KNN)
- Support Vector Machines (SVM)
- Gaussian Naïive Bayes (NB)
- Linear Discriminant Analysis (LDA)
- Random Forest (RF)
- AdaBoost (AB)

Model parameters

- KNN: In the present paper, the usual Euclidean distance has been used as distance metric, while K has been set to 5.
- SVM: Both the linear kernel and the RBF kernel are used. The penalty parameter C = 1 and the kernel coefficient $\gamma = 1/n$, where n is the number of features.
- AdaBoost: 500 decision trees have been used as base learners.
- Ensemble: A majority voting scheme has been adopted.

  First, every classification model has been trained on the features coming from each handwriting task and their performances have been evaluated.

  The best models, i.e., those showing the best result per task, have been pooled in the ensemble scheme to achieve the final classification.

  The ensemble obtained by combining only the best three tasks was also investigated.

Model validation

The classification performances have been validated with a stratified 10-fold cross-validation per group of participants

Feature Selection

- Nested single-feature SVMs were used for feature selection.
- All features have been ranked in accordance with their performance and only the n features providing the highest score have been selected for the final model fitting.

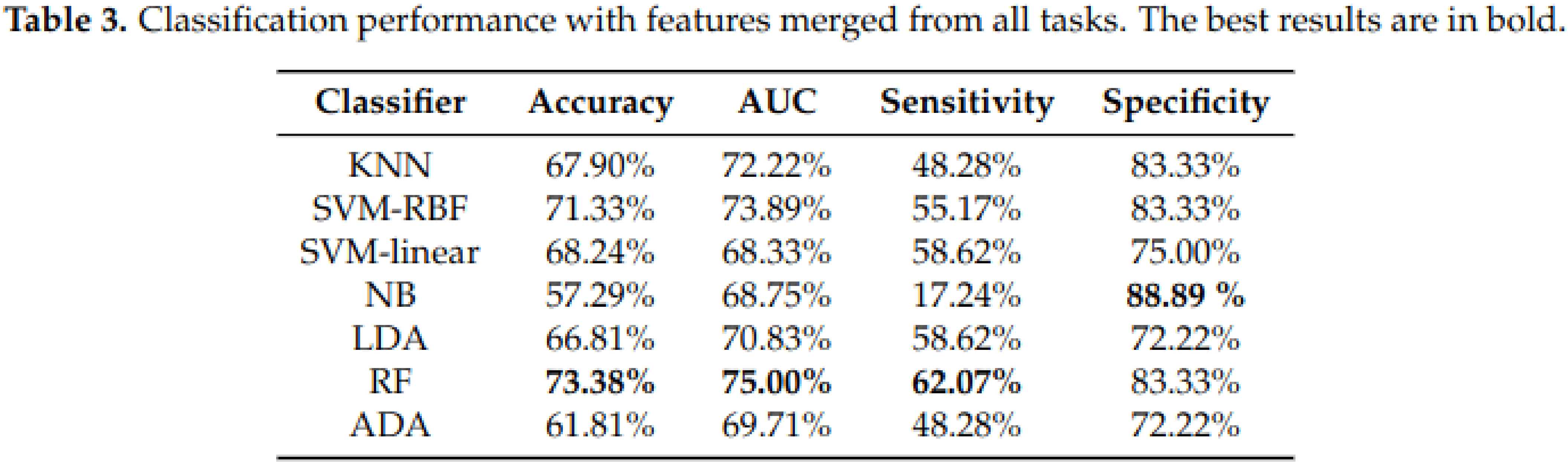


Table 3. Classification performance with features merged from all tasks. The best results are in bold.

Classifier	Accuracy	AUC	Sensitivity	Specificity
KNN	67.90%	72.22%	48.28%	83.33%
SVM-RBF	71.33%	73.89%	55.17%	83.33%
SVM-linear	68.24%	68.33%	58.62%	75.00%
NB	57.29%	68.75%	17.24%	**88.89 %**
LDA	66.81%	70.83%	58.62%	72.22%
RF	**73.38%**	**75.00%**	**62.07%**	83.33%
ADA	61.81%	69.71%	48.28%	72.22%

Generally speaking, all classifiers show low sensitivity and good specificity, indicating that such an approach may be better in identifying the absence of illness in the healthy population rather than the presence of illness in the pathological group.

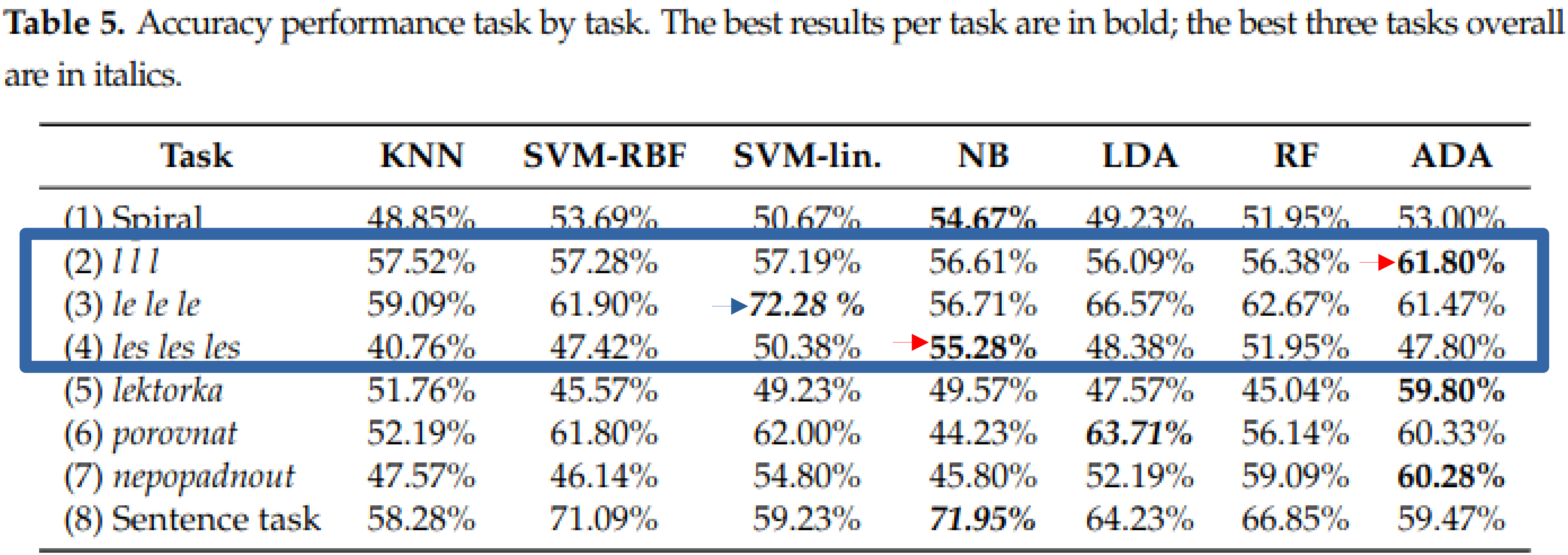


Table 5. Accuracy performance task by task. The best results per task are in bold; the best three tasks overall are in italics.

Task	KNN	SVM-RBF	SVM-lin.	NB	LDA	RF	ADA
(1) Spiral	48.85%	53.69%	50.67%	**54.67%**	49.23%	51.95%	53.00%
(2) *l l l*	57.52%	57.28%	57.19%	56.61%	56.09%	56.38%	**61.80%**
(3) *le le le*	59.09%	61.90%	***72.28 %***	56.71%	66.57%	62.67%	61.47%
(4) *les les les*	40.76%	47.42%	50.38%	**55.28%**	48.38%	51.95%	47.80%
(5) *lektorka*	51.76%	45.57%	49.23%	49.57%	47.57%	45.04%	**59.80%**
(6) *porovnat*	52.19%	61.80%	62.00%	44.23%	**63.71%**	56.14%	60.33%
(7) *nepopadnout*	47.57%	46.14%	54.80%	45.80%	52.19%	59.09%	**60.28%**
(8) Sentence task	58.28%	71.09%	59.23%	***71.95%***	64.23%	66.85%	59.47%

Conclusion

- In a nutshel, it was noticeable that ‘le’ task had the highest performance in the early detection of PD, on PDs that show mild symptoms using an SVM with a linear kernel, which may be due to the lack of fine motor control, especially in loop-like movements in the case of PDs.
- It’s also worth noting that, generally, all the models were better at identifying HCs than they were at identidying PDs due to their high specificity and low sensitivity.

Assessing visual attributes of handwriting for prediction of neurological disorders—A case study on Parkinson’s disease

Momina Moetesum et al, 2019

Introduction

- Goal: quantitatively evaluate the visual attributes in characterization of graphomotor samples of PD patients.
- Methods: Convolutional Neural Networks are employed to extract discriminating visual features from multiple representations of various graphomotor samples produced by both control and PD subjects.

  The extracted features are then fed to a Support Vector Machine (SVM) classifier.

  Evaluations are carried out on the PaHaW dataset using early and late fusion techniques.
- Results: an overall accuracy of 83% is realized with solely visual information.

Extracting images from the PaHaW dataset

Images are used in this study instead of dynamic data from the dataset like the pressure, inclination, etc.

To Extract images from the PaHaW dataset, the x and y coordinates were used to plot the image of pen-downs (when pressure > 0).

The images are then resized to a 227x227 pixels, the scaling technique wasn’t mentioned.

CNN for feature extraction

- Transfer learning was used by employing a pre-trained CNN (AlexNet).
- AlexNet is a convolutional neural network (CNN) architecture that was developed by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, and it played a significant role in advancing the field of deep learning. It won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012.
- AlexNet architecture comprises of 5 convolution layers, max-pooling layers, dropout layers, and 3 fully connected layers.
- It is trained on 1.2 Million images (with 1000 different classes) of the ImageNet dataset.
- The network constructs a hierarchical representation of input images. Deeper layers contain higher-level features, constructed using the lower-level features of earlier layers.
- Together, the convolutional and down sampling layers serve as feature extractors while the fully connected layers represent a trainable classifier similar to a standard multi-layer neural network.
- In transfer learning that was chosen, the fully connected layers (performing classification) are removed and the output of the feature extractor layers is fed to another classifier.
- In this study, they employed transfer learning by extracting features at fc7.

Early fusion technique

- One limitation of a CNN is the computation of only linear characteristics.
- Hence to enhance feature learning, it was proposed to present to the CNN the initial data and the result of the transformation of this initial data through different non linear transforms.
- Three representations of the input data are used to train three independent networks for each of the 8 tasks performed by a subject.

- Raw data (Dr): Conventionally raw images are used as input data for CNNs as they contain different frequency components that can be extracted and used for image classification. Raw images of the 8 tasks, completed by subjects, are used to train the first network.
- Median filter residual data (Dm): The second network is trained using median filter residuals of the same raw images. To compute the median filter residual, they applied a 3 × 3 median filter on the raw image and then subtracted the raw image from the resultant filtered image. The idea is to preserve high frequency imperfections.
- Edge data (De): The third network is trained using images containing only the edge information from the raw images. Edges are known to contain useful information in most cases. By applying linear convolution filters in vertical and horizontal directions, the magnitude of the gradient is computed in a non linear way. As a result, they obtained emphasized edge information of the shape and used it to train the network.

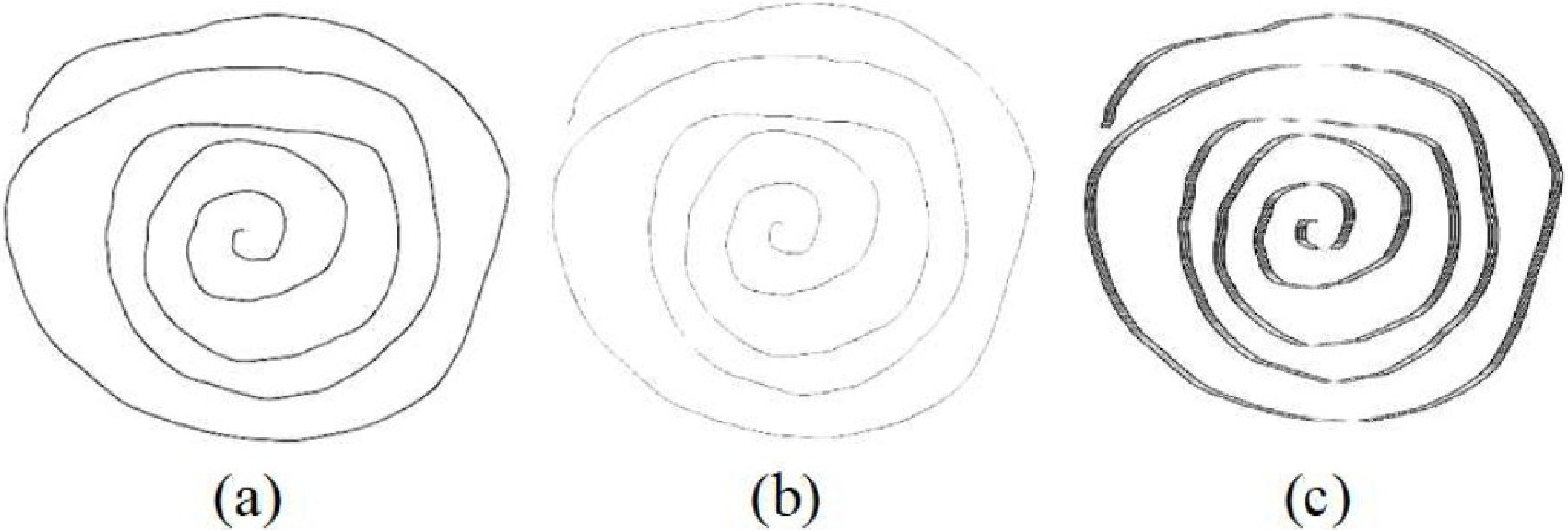


Fig. 7. (a) Raw image, (b) Median filter residual image (Pixel values inverted for better visualization), (c) Edge image (Inverted for better visualization)

Late fusion technique

- The objective at hand is to take task level decisions from multiple samples of the same subject.
- The outputs of the 8 classifiers in our system form the decision vector d defined as
- $d = [d_1, d_2, d_3, ..., d8]^T$;
- where $d_i \in \{c_1, c_2\}$ and c_i denotes label of either of the class (i.e. Healthy/PD).
- In the next step, voting based late fusion was applied.
- Majority Voting was used.

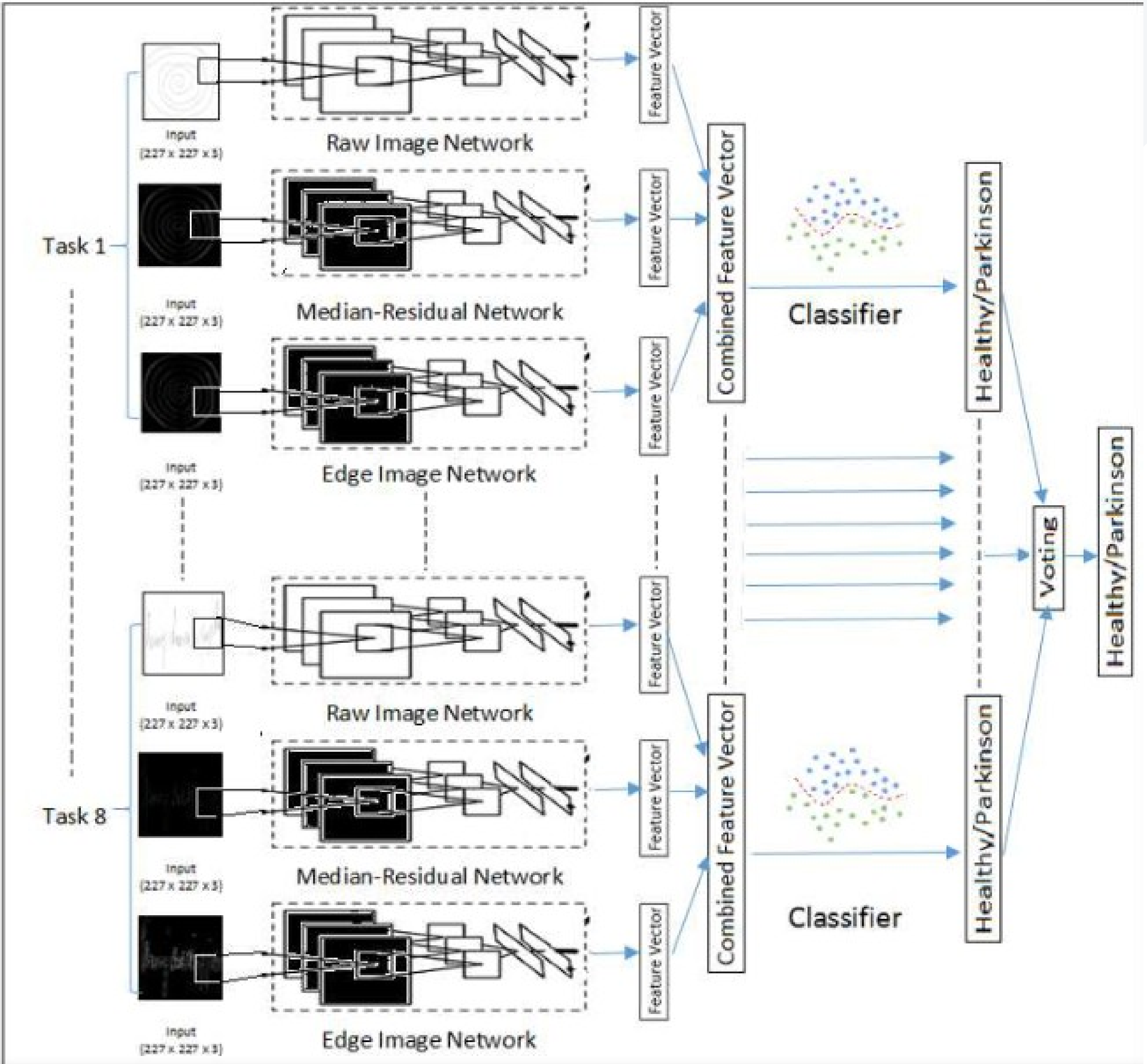


Fig. 4. System Overview

The experiment

- The effectiveness of the proposed image representations is evaluated by computing the system accuracy for each of the tasks separately and against each representation.
- Accuracy is also reported by combining the feature vectors of the three representations (early fusion) as well as by combining the predictions of the eight modalities through majority vote (late fusion).
- Furthermore, class-wise precision, specificity and sensitivity values are also reported.
- Validation is done through the average of the 10 runs of a 10 fold cross-validation.

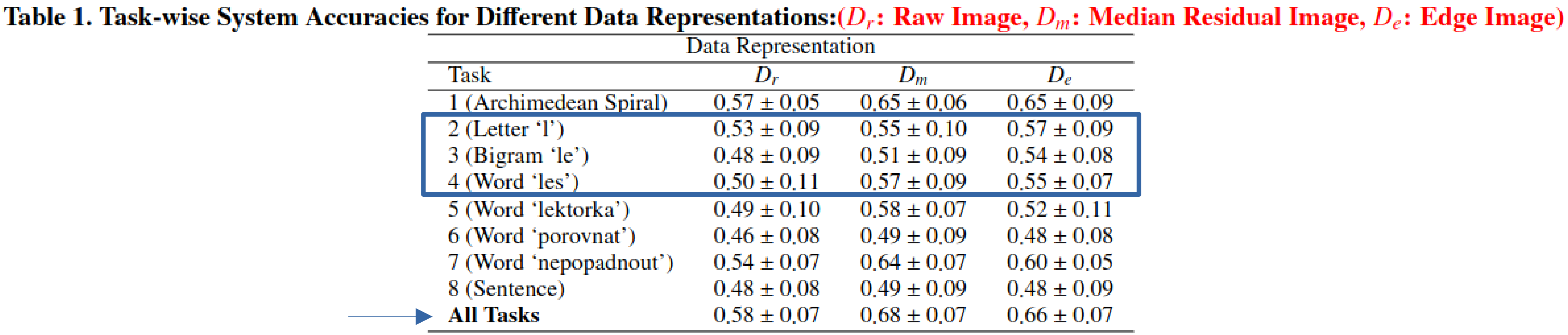


Table 1. Task-wise System Accuracies for Different Data Representations:(*D_r*: **Raw Image,** *D_m*: **Median Residual Image,** *D_e*: **Edge Image)**

	Data Representation		
Task	D_r	D_m	D_e
1 (Archimedean Spiral)	0.57 ± 0.05	0.65 ± 0.06	0.65 ± 0.09
2 (Letter ‘l’)	0.53 ± 0.09	0.55 ± 0.10	0.57 ± 0.09
3 (Bigram ‘le’)	0.48 ± 0.09	0.51 ± 0.09	0.54 ± 0.08
4 (Word ‘les’)	0.50 ± 0.11	0.57 ± 0.09	0.55 ± 0.07
5 (Word ‘lektorka’)	0.49 ± 0.10	0.58 ± 0.07	0.52 ± 0.11
6 (Word ‘porovnat’)	0.46 ± 0.08	0.49 ± 0.09	0.48 ± 0.08
7 (Word ‘nepopadnout’)	0.54 ± 0.07	0.64 ± 0.07	0.60 ± 0.05
8 (Sentence)	0.48 ± 0.08	0.49 ± 0.09	0.48 ± 0.09
All Tasks	0.58 ± 0.07	0.68 ± 0.07	0.66 ± 0.07

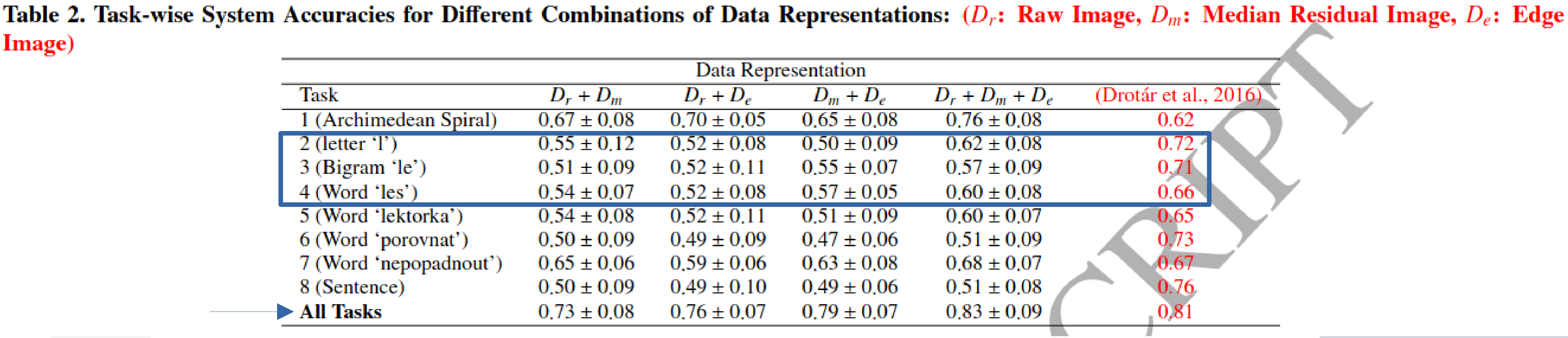


Table 2. Task-wise System Accuracies for Different Combinations of Data Representations: (*D_r*: **Raw Image,** *D_m*: **Median Residual Image,** *D_e*: **Edge Image)**

	Data Representation				
Task	$D_r + D_m$	$D_r + D_e$	$D_m + D_e$	$D_r + D_m + D_e$	(Drotár et al., 2016)
1 (Archimedean Spiral)	0.67 ± 0.08	0.70 ± 0.05	0.65 ± 0.08	0.76 ± 0.08	0.62
2 (letter ‘l’)	0.55 ± 0.12	0.52 ± 0.08	0.50 ± 0.09	0.62 ± 0.08	0.72
3 (Bigram ‘le’)	0.51 ± 0.09	0.52 ± 0.11	0.55 ± 0.07	0.57 ± 0.09	0.71
4 (Word ‘les’)	0.54 ± 0.07	0.52 ± 0.08	0.57 ± 0.05	0.60 ± 0.08	0.66
5 (Word ‘lektorka’)	0.54 ± 0.08	0.52 ± 0.11	0.51 ± 0.09	0.60 ± 0.07	0.65
6 (Word ‘porovnat’)	0.50 ± 0.09	0.49 ± 0.09	0.47 ± 0.06	0.51 ± 0.09	0.73
7 (Word ‘nepopadnout’)	0.65 ± 0.06	0.59 ± 0.06	0.63 ± 0.08	0.68 ± 0.07	0.67
8 (Sentence)	0.50 ± 0.09	0.49 ± 0.10	0.49 ± 0.06	0.51 ± 0.08	0.76
All Tasks	0.73 ± 0.08	0.76 ± 0.07	0.79 ± 0.07	0.83 ± 0.09	0.81

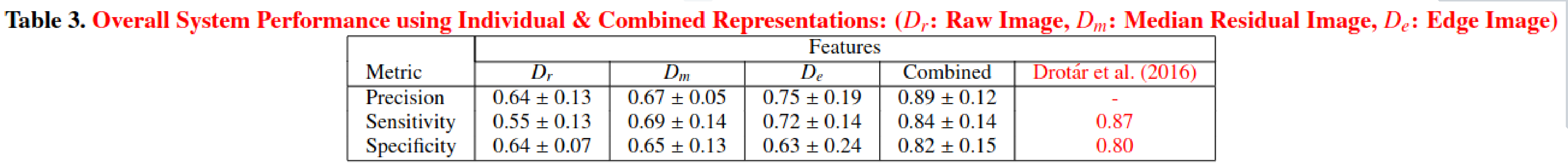


Table 3. Overall System Performance using Individual & Combined Representations: (D_r: Raw Image, D_m: Median Residual Image, D_e: Edge Image)

	Features				
Metric	D_r	D_m	D_e	Combined	Drotár et al. (2016)
Precision	0.64 ± 0.13	0.67 ± 0.05	0.75 ± 0.19	0.89 ± 0.12	-
Sensitivity	0.55 ± 0.13	0.69 ± 0.14	0.72 ± 0.14	0.84 ± 0.14	0.87
Specificity	0.64 ± 0.07	0.65 ± 0.13	0.63 ± 0.24	0.82 ± 0.15	0.80

Conclusion

Once again, the CNN model didn’t perform that well on individual tasks, however it did perform quite well when using combined tasks and fusion techniques.

Characterizing Early Stage Alzheimer through Spatiotemporal Dynamics of Handwriting

Christian Kahindo et al, 2018

Introduction

- Goal: To characterize early Alzheimer, based on the analysis of online handwritten cursive loops.
- Results: The classification performance significantly outperforms the state of the art, based on global kinematic features.

The dataset

The dataset consists of a task consisting of four cursive llll series, written by each participant of two cognitive profiles, Early-stage Alzheimer Disease and Healthy persons, each comprising 27 persons collected at Broca Hospital in Paris.

Segmentation and feature extraction

- Segment each llll series into isolated l letters, from which they keep only the loop part for subsequent feature extraction.
- Segmentation is done through the Fast Fourier Transform.
- The velocity in x and y directions.

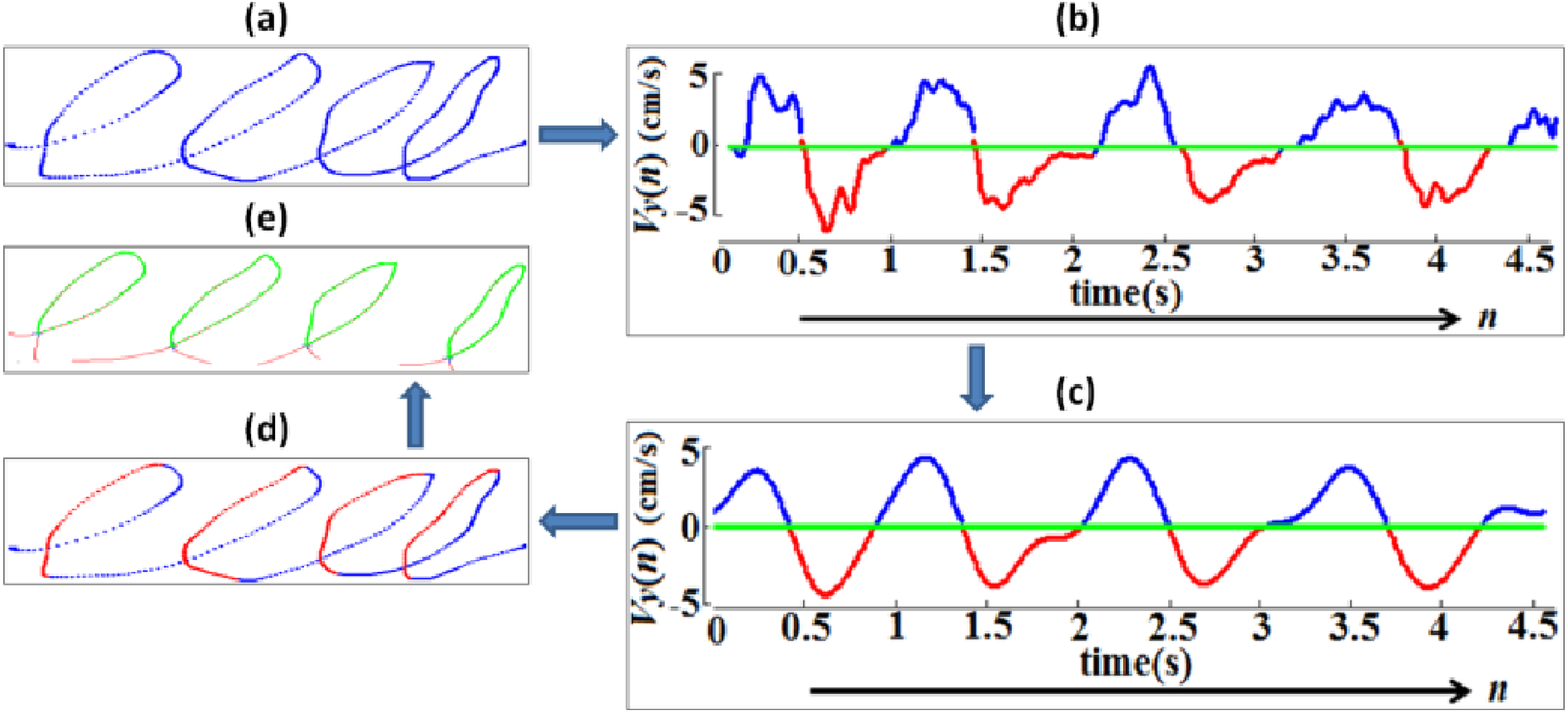


Fig. 1. Loop segmentation: (a) input loops series; (b) the *Vy*(*n*) signal, (c) low-pass filtering by the fundamental frequency; (d) segmentation into ascending and descending strokes; (e) extraction of the loops.

Modeling the data

- K-medoids clustering is a partitioning algorithm used for grouping data into K clusters, where each cluster is represented by a data point (medoid) that minimizes the average dissimilarity to all other points within the cluster.
- K-medoids clustering was performed based on the sequential velocity representation, Vx(n),Vy(n), of all the loops from all subjects.
- K was varied between 10 and 50, and obtained similar optimal performance for K between 30 and 50. Here, they report the results for K = 30.
- DTA distance was used as a distance metric.

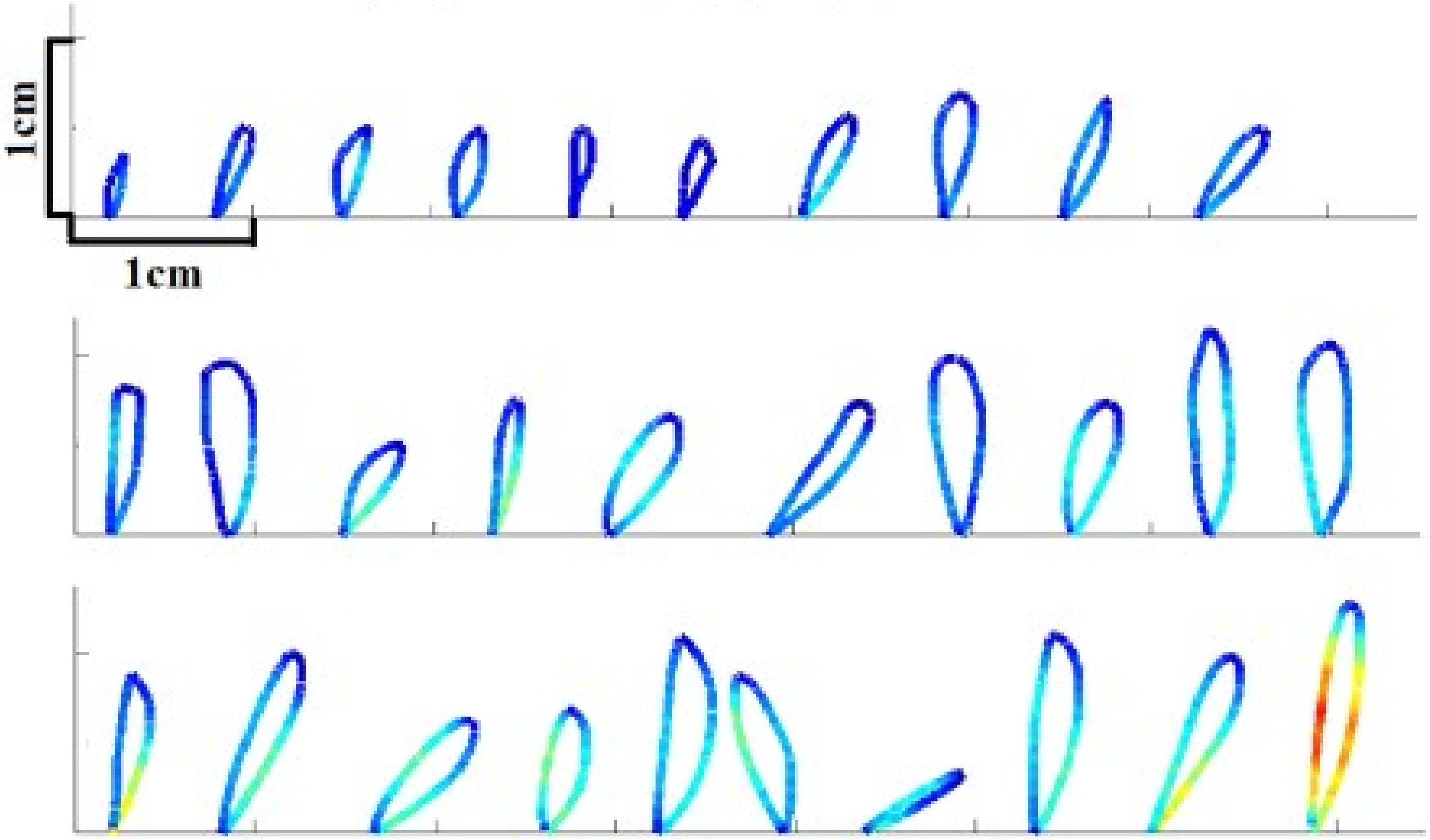
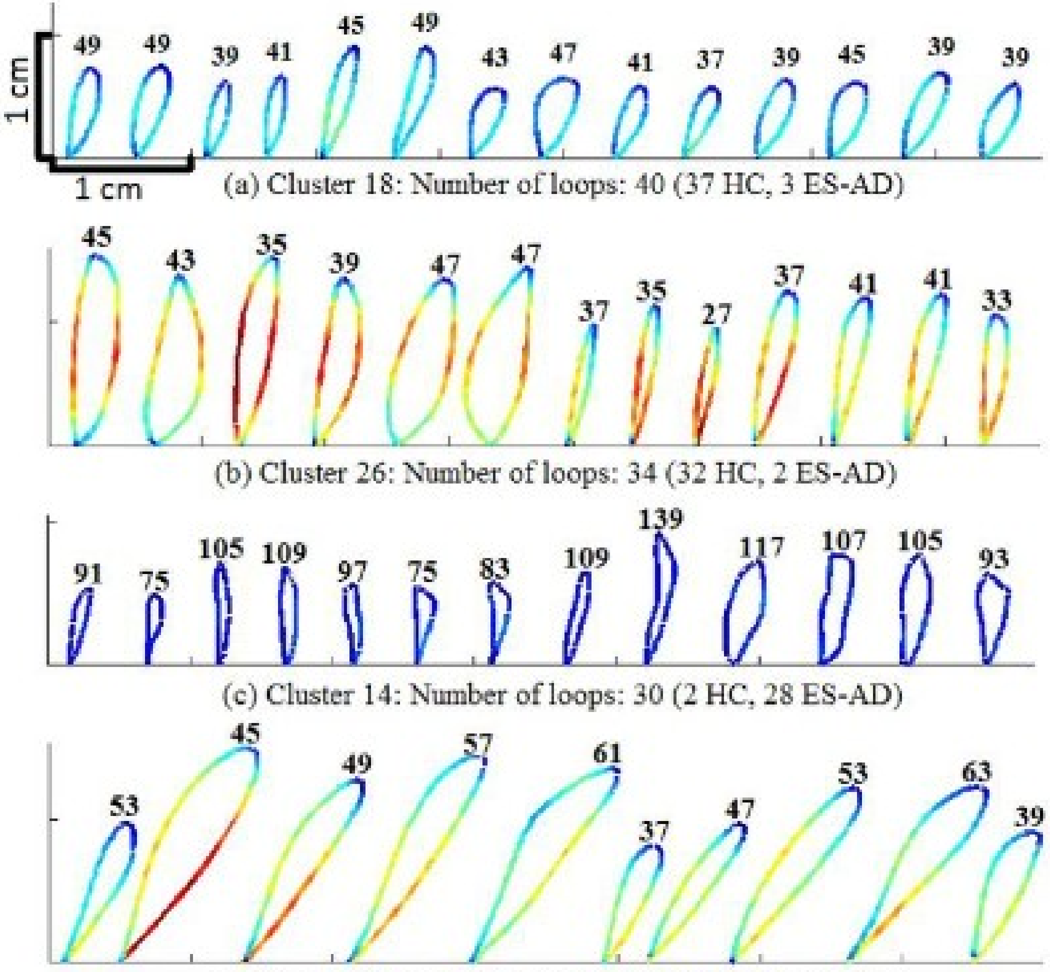


Fig. 2. Medoids characterized by their velocity trajectory, ordered by DTW, with the smallest medoid as reference. The color scale characterizes velocity magnitude: blue stands for low speed and red for high speed.



(a) Cluster 18: Number of loops: 40 (37 HC, 3 ES-AD)

(b) Cluster 26: Number of loops: 34 (32 HC, 2 ES-AD)

(c) Cluster 14: Number of loops: 30 (2 HC, 28 ES-AD)

(d) Cluster 19: Number of loops: 29 (6 HC, 23 ES-AD)

Fig. 3. Samples from four clusters. For each, we report its number of loops, the number of loops for each class, and the number of points for each loop.

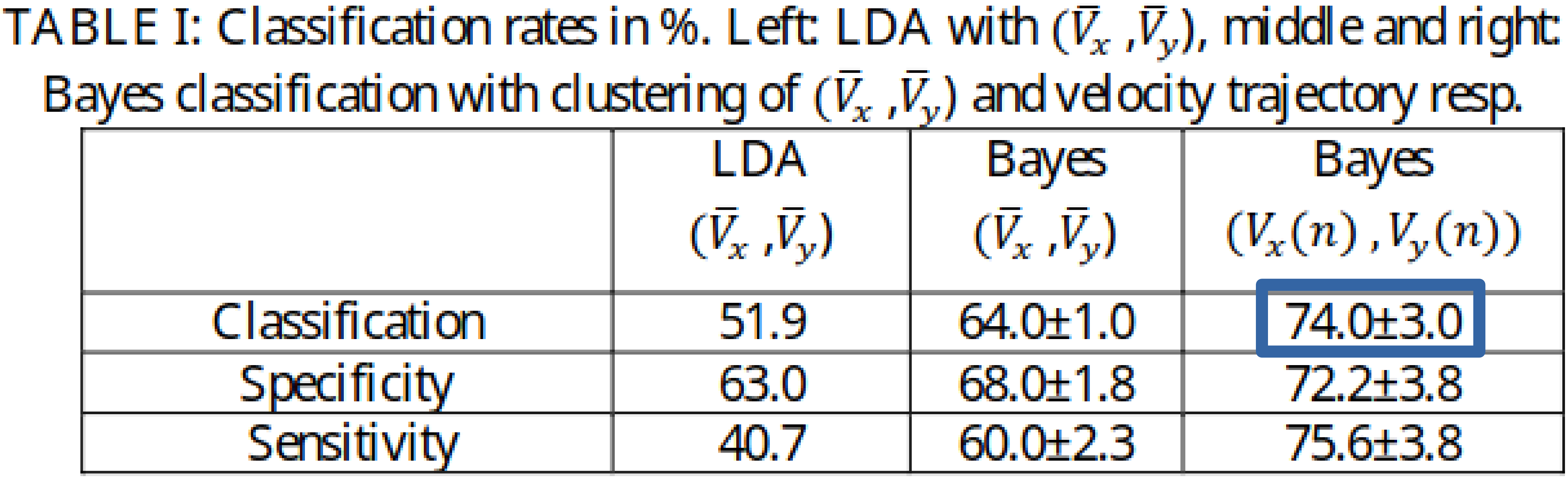
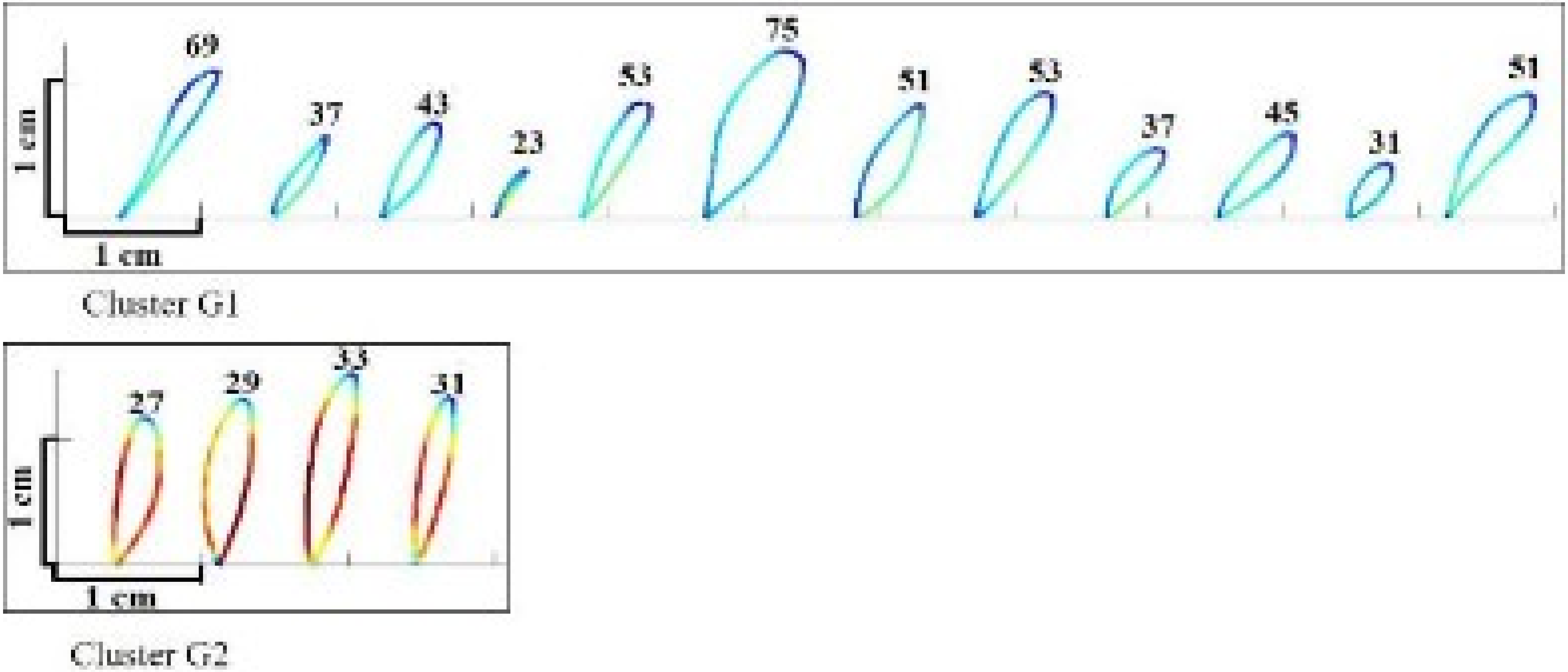


TABLE I: Classification rates in %. Left: LDA with $(\bar{V}_x ,\bar{V}_y)$, middle and right: Bayes classification with clustering of $(\bar{V}_x ,\bar{V}_y)$ and velocity trajectory resp.

	LDA $(\bar{V}_x ,\bar{V}_y)$	Bayes $(\bar{V}_x ,\bar{V}_y)$	Bayes $(V_x(n) ,V_y(n))$
Classification	51.9	64.0±1.0	74.0±3.0
Specificity	63.0	68.0±1.8	72.2±3.8
Sensitivity	40.7	60.0±2.3	75.6±3.8

They ran an additional experiment in which they performed a clustering of the loops based on their $(V\bar{x}, V\bar{y})$ values. The clustering is based on K-medoids as before, but now it takes as input $(V\bar{x}, V\bar{y})$, with an Euclidian distance as a dissimilarity measure, instead of DTW.



Cluster G1

Cluster G2

Fig. 4. Some samples from two clusters obtained with average velocity.

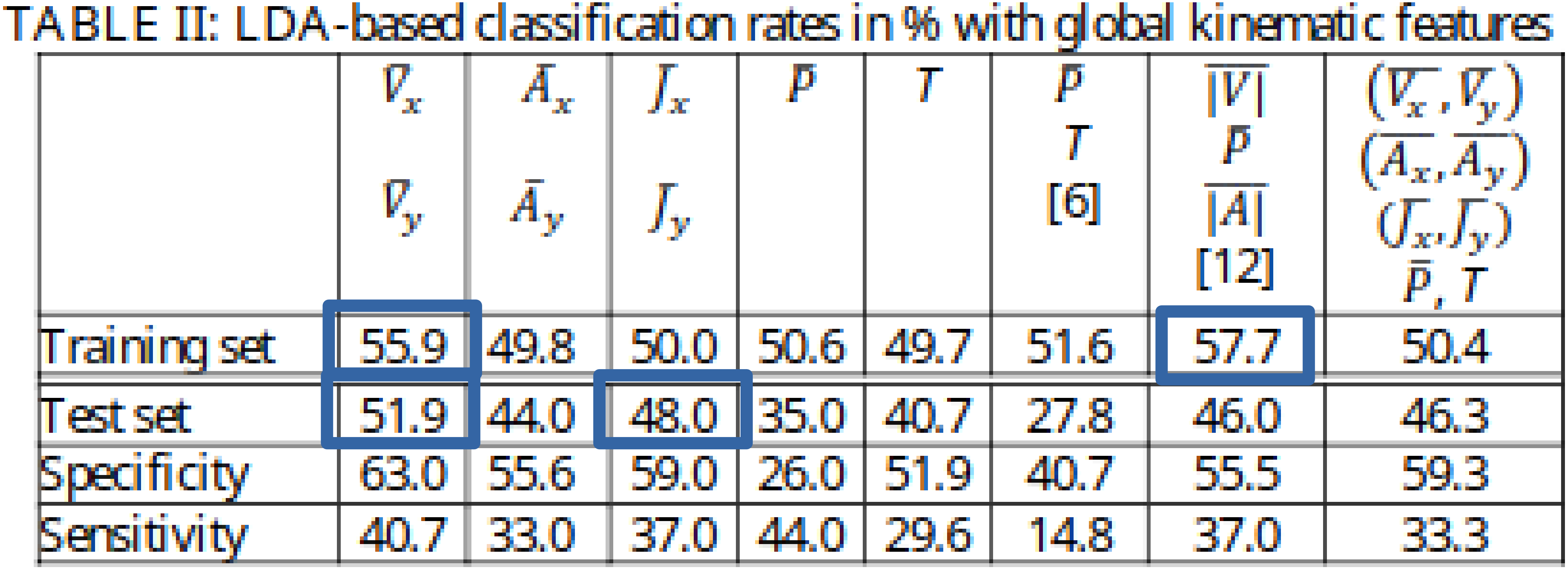


TABLE II: LDA-based classification rates in % with global kinematic features

	$\bar{V}_x$ $\bar{V}_y$	$\bar{A}_x$ $\bar{A}_y$	$\bar{J}_x$ $\bar{J}_y$	$\bar{P}$	T	$\bar{P}$ T [6]	$\overline{\lvert V\rvert}$ $\bar{P}$ $\overline{\lvert A\rvert}$ [12]	$(\bar{V}_x, \bar{V}_y)$ $(\bar{A}_x, \bar{A}_y)$ $(\bar{J}_x, \bar{J}_y)$ $\bar{P}$, T
Training set	55.9	49.8	50.0	50.6	49.7	51.6	57.7	50.4
Test set	51.9	44.0	48.0	35.0	40.7	27.8	46.0	46.3
Specificity	63.0	55.6	59.0	26.0	51.9	40.7	55.5	59.3
Sensitivity	40.7	33.0	37.0	44.0	29.6	14.8	37.0	33.3

Conclusion

Current methods assume homogeneous behavior in Alzheimer's patients based on kinematic parameters. To overcome this limitation, the proposed approach uses unsupervised learning and K-medoids clustering with Dynamic Time Warping to uncover subgroups within the population. The method achieves a 74% classification rate on the validation set, outperforming existing approaches by over 50% by analyzing the full velocity dynamics instead of global average parameters.

Key Takeaways

- The best performance was achieved using CNN-BiGRU model using derived features, a particularly high performance was observed in loop-like tasks.
- Deep learning techniques proved to be much better than traditional machine learning techniques even in small datasets.
- The challenge of overfitting that comes from the dataset being too small, is beat by techniques such as transfer learning, data augmentation, fusion, etc.
- In all the models, an ensemble scheme on all tasks provided much better results than single-task individual models.
- Deep learning models that used online data generally performed better than models that used offline data, in per-task evaluation and in ensemble learning.

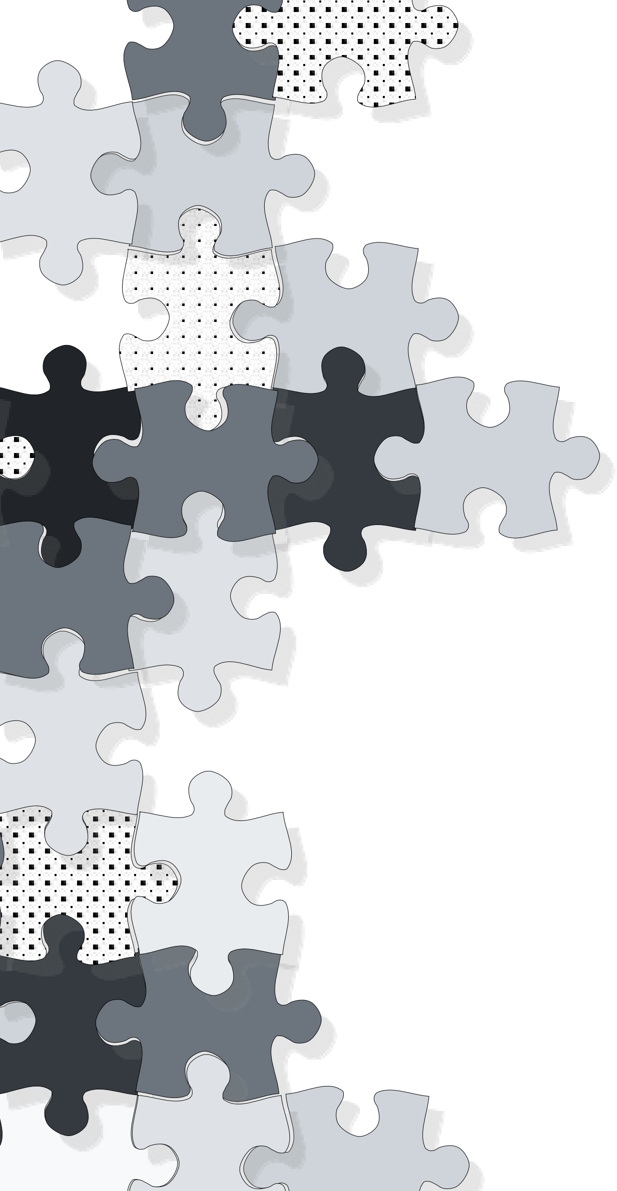
The Proposed Approach

- Statistical evaluation (e.g. clustering, fatigue over time, loop segmentation, etc)
- Deep learning approach using a combination of CNNs and RNNs.
- Online data and feature extraction.
- Some techniques such as transfer learning and data augmentation, since the dataset available is relatively small.
- Cross-validation also due to the size of the dataset.
- Evaluation is done:

  Model wise.

  Technique wise.

  Using ensemble schemes.



“

Thank you for
your attention !

”