# A Vision-Based Framework for Predicting Multiple Sclerosis and Parkinson's Disease Gait Dysfunctions—A Deep Learning Approach

Rachneet Kaur, Robert W. Motl, Richard Sowers, *Member, IEEE*,
and Manuel E. Hernandez, *Member, IEEE*

***Abstract*—This study examined the effectiveness of a vision-based framework for multiple sclerosis (MS) and Parkinson's disease (PD) gait dysfunction prediction. We collected gait video data from multi-view digital cameras during self-paced walking from MS, PD patients and age, weight, height and gender-matched healthy older adults (HOA). We then extracted characteristic 3D joint keypoints from the collected videos. In this work, we proposed a data-driven methodology to classify strides in persons with MS (PwMS), persons with PD (PwPD) and HOA that may generalize across different walking tasks and subjects. We presented a comprehensive quantitative comparison of 16 diverse traditional machine and deep learning (DL) algorithms. When generalizing from comfortable walking (W) to walking-while-talking (WT), multi-scale residual neural network achieved perfect accuracy and AUC for classifying individuals with a given gait disorder; for subject generalization in W trials, residual neural network resulted in the highest accuracy and AUC of $78.1\%$ and 0.87 (resp.), and 1D convolutional neural network (CNN) had highest accuracy of $75\%$ in WT trials. Finally, when generalizing over new subjects in different tasks, again 1D CNN had the top classification accuracy and AUC of $79.3\%$ and 0.93 (resp.). This work is the first attempt to apply and demonstrate the potential of DL with a multi-view digital camera-based gait analysis framework for neurological gait dysfunction prediction. This study suggests the viability of inexpensive vision-based systems for diagnosing certain neurological disorders.**

***Index Terms*—Multiple sclerosis, Parkinson's disease, gait videos, pose estimation, deep learning.**

Manuscript received 10 December 2021; revised 30 June 2022 and 28 August 2022; accepted 15 September 2022. Date of publication 20 September 2022; date of current version 5 January 2023. *(Corresponding author: Rachneet Kaur.)*

Rachneet Kaur is with the Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana Champaign, Urbana, IL 61801 USA (e-mail: rk4@illinois.edu).

Robert W. Motl is with the Department of Kinesiology and Nutrition, University of Illinois Chicago, Chicago, IL 60612 USA (e-mail: robmotl@uic.edu).

Richard Sowers is with the Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana Champaign, Urbana, IL 61801 USA, and also with the Department of Mathematics, University of Illinois at Urbana Champaign, Urbana, IL 61801 USA (e-mail: r-sowers@illinois.edu).

Manuel E. Hernandez is with the Department of Kinesiology and Community Health, University of Illinois at Urbana Champaign, Urbana, IL 61801 USA (e-mail: mhernand@illinois.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/JBHI.2022.3208077, provided by the authors.

Digital Object Identifier 10.1109/JBHI.2022.3208077

## I. Introduction

NEUROLOGICAL gait disorders are associated with an increased risk of falls in older adults [1]. Abnormal gait has been observed in 35% of older adults, and associated with a greater risk of institutionalization and mortality [2]. While gait evaluation is common [3], few studies have focused on the differentiation of neurological disorders, such as Parkinson's disease (PD) or multiple sclerosis (MS), using gait analysis [4], [5]. Various gait evaluations, such as motion capture during the timed 25 ft walk and timed up and go test have been explored in clinical settings to assess neurological conditions, such as MS [6], [7] and PD [8]. Typically, specialized equipment like a lab-based motion capture system, force plate or electromyography sensors often is needed for these clinical quantitative gait measures, which can be expensive and require skilled personnel to analyze. Recent work on movement analysis with wearable inertial measurement unit sensors [9], smartwatches and smartphones [10] has overcome some of these constraints, yet these approaches are not contact-free and may require installation of multiple sensors. Past studies have explored depth cameras for gait monitoring [11], but these are relatively costly and not as easy to use. Herein, we used a standard RGB digital camera to examine pathological gait. This proposed system allows for passive and remote gait monitoring at reduced cost and effort, which should aid in making it a viable point-of-care technology for early detection of gait alterations in real-world settings. Moreover, we apply computer vision and deep learning (DL) algorithms to process our gait videos and extract significant information for an automated and objective quantification of neurological conditions. Given the inherently complex gait dynamics with little-known direct descriptors for the disorders, hand engineering of features in this situation is complicated. DL automates this process of feature extraction and eliminates the need for domain expertise to allow for a remote real-time application, possibly at homes, of our entire workflow.

This study introduced and examined a vision-based gait analysis framework using DL for MS and PD gait dysfunction prediction. We extend prior work examining MS-related variations in spatiotemporal and kinetic gait characteristics [12], [13]. We classify strides of persons with MS (PwMS), healthy older adults (HOA), and persons with PD (PwPD) across three classification designs:

1) *Task Generalization:* We train ternary (HOA, PwMS or PwPD) classifiers on walking (W) trials (tasks) and use them to test on walking-while-talking (WT) trials. This experimental paradigm might be useful in quantifying how algorithms trained on normative data collected in a supervised lab or clinic could be used as a basis to assess gait data collected in a real-world home-based setting with challenges of divided attention.
2) *Subject Generalization:* We train ternary classifiers on a balanced subset of subjects and use them to test on the remaining subjects. These algorithms may be useful in detection of disease in new patients.
3) *Task-Subject Generalization:* We train our classifiers on a balanced subset of subjects in W trials and use them to test on the remaining subjects in WT trials. This generalization framework is designed to simulate how algorithms could be used to predict disease in new subjects in more real-world settings.

## II. Related Results and Context

Past studies have quantified the decline of gait performance in PwMS [14]. Most gait-based approaches for MS detection have been based on statistical analyses of kinematic and kinetic data [15], [16]. Several recent works have applied traditional supervised machine learning (ML) to classify MS using gait data collected via treadmill [13], smartwatches and smartphones [10]. Vision methodologies based upon digital cameras have also been used to estimate clinical gait parameters in human gait analysis [17], [18] and categorize other neurological populations [5], [19]. Depth cameras capturing 3D movement patterns have been explored for gait assessment in subjects with motion difficulties [20], [21], but those systems require a relatively costlier hardware, have some limitations when used outdoors and are constrained by the camera to object distance. Our contribution is using DL with a multi-view digital camera-based gait analysis framework for prediction of gait-related neurological disorders. Of particular novelty is our focus on MS. We further considered a dual-task walking paradigm and consequently, a task-subject generalization classification framework. Most prior work has been focused on binary healthy-vs.-pathological gait [5], [10], [13]; we investigated a more challenging multi-class setup which further involves distinguishing between different causes of the neurological gait. Unlike past studies [5], [19], we have added feet features along with other body coordinates in our analysis.

The proposed application of vision and DL to learn gait dynamics in PwMS and PwPD across tasks, and over new subjects is a step towards the identification of worsening of symptoms in the near term. Our system requires only an inexpensive digital camera, and thus can be easily and economically deployed in homes of older adults for a real-time gait analysis with negligible user interaction. We provided a comprehensive quantitative comparison of 16 diverse ML and DL algorithms for all classification designs which may assist researchers in the selection of suitable model architectures and hyperparameters. Moreover, we discussed the global and local importance of our extracted features in the classification performance; and explored a potential association between our model predictions and the lower extremity function of subjects.

## III. Experimental Design: Subjects and Setup

The study consisted of 33 participants; 10 PwMS (age: $66 \pm 5$ years, 3 male), 9 PwPD (age: $68 \pm 9$ years, 6 male), and 14 HOA (age: $63 \pm 9$ years, 3 male). All participants were medically stable, had a cognitive status score [22] of above 18, were right-hand dominant, had no lower limb injury in the past six months, and had normal or corrected-to-normal vision. PwMS had mild to moderate disability as evaluated by the Kurtzke Expanded Disability Status Scale (EDSS) [23] $[2.0 - 6.0]$, were relapse-free for 30 days prior to experimental trials and had no other cognitive dysfunction or neurological disorders. PwPD had mild to moderate severity on the Hoehn & Yahr Scale [24] $[1.0 - 4.0]$, were "ON" anti-Parkinsonian medication state and had no other cognitive dysfunction. Prior to testing, all participants provided informed consent approved by the local Institutional Review Board (Protocol No. 15674).

All subjects performed two self-paced walking tasks on an instrumented treadmill (C-Mill, Motekforce Link): 1) a trial in single-task walking (W) and 2) a trial in dual-task walking-while-talking (WT) condition. For the WT task, subjects were asked to walk and recite alternate letters of the alphabet while providing an equal priority to both walking and talking. Two $800 \times 448$ pixels resolution digital cameras were located facing subject's front and right side to record their lower half and feet movements (resp.) at 30 frames per second. Given prior evidence of increased variability in footfall placement in PwMS [25], we focused our cameras on subject's feet and lower extremity in this study (Fig. 1). All extracted gait videos were truncated to 60 seconds, to account for alterations in gait speed during gait initialization and gait arrest. Additionally for validation, CueFors 2 software was used to collect gait events and raw center of pressure (CoP) position coordinates at 500 Hz during each walking trial. A total of 116 gait videos, combining subject's front- and side-views, were gathered for 33 (W: 32, WT: 26) subjects.

## IV. Data Analysis: Gait Video Processing

### A. 2D Body Pose Estimation

OpenPose [26] was used to locate the 2D pixel coordinates, estimating the skeletal joint positions of a detected subject in each frame of the collected gait videos. OpenPose provides an open-source real-time architecture for robust body pose estimation [27] using a fine-tuned VGG-19 convolutional neural network (CNN) [28]. OpenPose generated the 2D location coordinates and corresponding prediction confidence of 12 front-view lower extremity landmarks (i.e., hips, knees, ankles and foot keypoints) and 8 side-view ankle and foot landmarks for both sides of the body (see Fig. 1). OpenPose may occasionally generate erroneous poses with left and right sides swapped, missing keypoints, or falsely perceived human body due to a range of possible reasons, including self-occlusion, varying lighting or color information. Thus post-processing involved correcting for

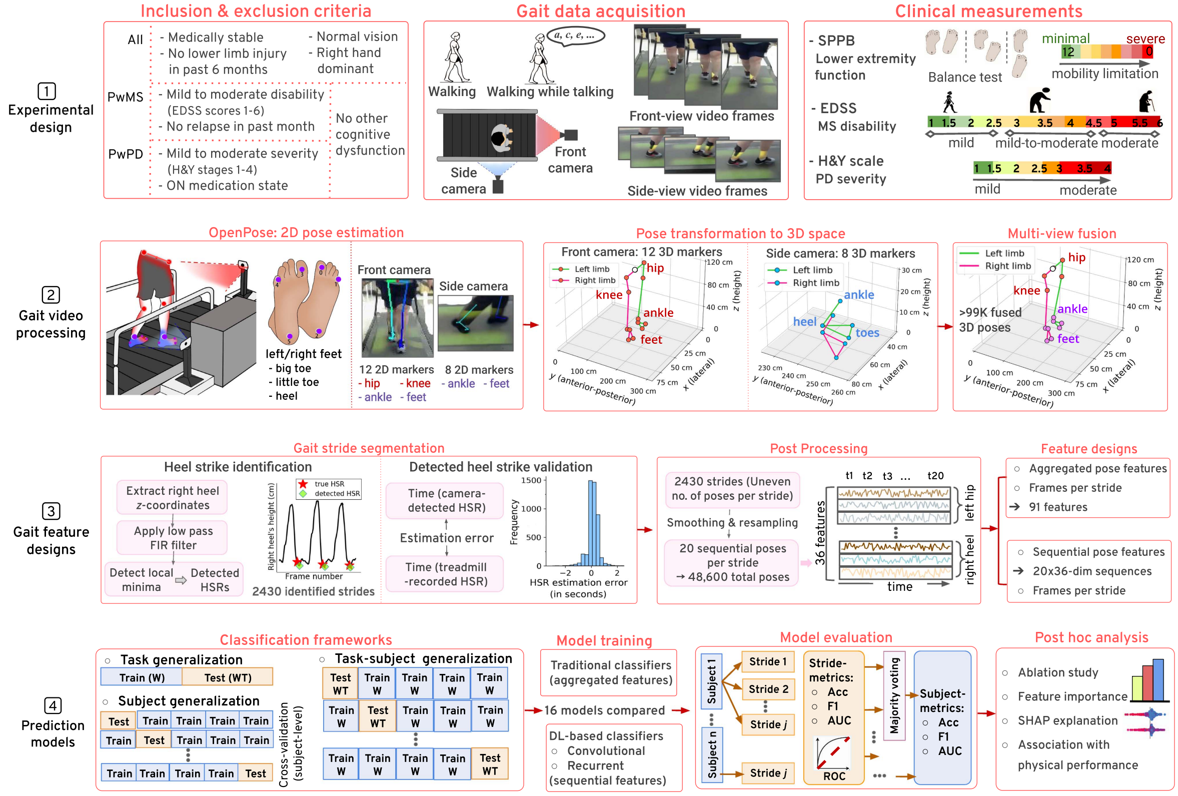
1 Experimental design
Inclusion & exclusion criteria
All - Medically stable - No lower limb injury in past 6 months - Normal vision - Right hand dominant
PwMS - Mild to moderate disability (EDSS scores 1-6) - No relapse in past month
PwPD - Mild to moderate severity (H&Y stages 1-4) - ON medication state
No other cognitive dysfunction
Gait data acquisition
Walking
Walking while talking
a, c, e, ...
Front-view video frames
Side camera
Front camera
Side-view video frames
Clinical measurements
- SPPB Lower extremity function
Balance test
minimal severe
12 0
mobility limitation
- EDSS MS disability
1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6
mild mild-to-moderate moderate
- H&Y scale PD severity
1 1.5 2 2.5 3 3.5 4
mild moderate
2 Gait video processing
OpenPose: 2D pose estimation
left/right feet - big toe - little toe - heel
Front camera
Side camera
12 2D markers - hip - knee - ankle - feet
8 2D markers - ankle - feet
Pose transformation to 3D space
Front camera: 12 3D markers
Left limb
Right limb
hip
knee
ankle
feet
Side camera: 8 3D markers
heel
toes
y (anterior-posterior)
x (lateral)
z (height)
Multi-view fusion
>99K fused 3D poses
3 Gait feature designs
Gait stride segmentation
Heel strike identification
Extract right heel z-coordinates
Apply low pass FIR filter
Detect local minima
Detected HSRs
true HSR
detected HSR
Right heel's height (cm)
Frame number
2430 identified strides
Detected heel strike validation
Time (camera-detected HSR)
Estimation error
Time (treadmill-recorded HSR)
Frequency
HSR estimation error (in seconds)
Post Processing
2430 strides (Uneven no. of poses per stride)
Smoothing & resampling
20 sequential poses per stride → 48,600 total poses
t1 t2 t3 ... t20
36 features
left hip
right heel
time
Feature designs
Aggregated pose features
Frames per stride
→ 91 features
Sequential pose features
→ 20x36-dim sequences
Frames per stride
4 Prediction models
Classification frameworks
Task generalization
Train (W) Test (WT)
Subject generalization
Task-subject generalization
Test Train Train Train Train
Train Test Train Train Train
Train Train Train Train Test
Test WT Train W Train W Train W Train W
Cross-validation (subject-level)
Model training
Traditional classifiers (aggregated features)
16 models compared
DL-based classifiers
Convolutional
Recurrent
(sequential features)
Model evaluation
Subject 1
Subject n
Stride 1
Stride 2
Stride j
Stride-metrics: Acc F1 AUC
ROC
Majority voting
Subject-metrics: Acc F1 AUC
Post hoc analysis
Ablation study
Feature importance
SHAP explanation
Association with physical performance

Fig. 1. Workflow pipeline. The proposed vision-based gait analysis framework for MS and PD gait dysfunction prediction.

switches between the left and right limbs, quadratic interpolation of missing markers, and identification of erroneous landmarks. Following processing, 2D skeletal landmark coordinates (in pixels) were retained from 102,598 front- and side-view gait postures.

## B. Pose Transformation to 3D Global Coordinate Space

Camera calibration was carried out, using intrinsic and extrinsic camera matrices, to transform the estimated front- and side-view 2D joint locations in local image pixel coordinates to 3D positions in a global coordinate system. To computationally approximate intrinsic and extrinsic camera parameters, we calibrated both of our cameras using sample patterns from 3D real world position and corresponding 2D image coordinates of square corners in a chess board. Post-processing consisted of bounding all computed 3D position coordinates using real-world constraints (i.e., treadmill width and length, and height of person). An example of computed front- (red markers) and side-view (blue markers) 3D poses is shown in Fig. 1.

## C. Multi View Fusion of 3D Body Poses

We conducted a weighted mean-based multi-view fusion, as proposed in [19], of 3D joint positions across views in the two planes; this helps to account for deviations in 2D pose approximation (IV-A) and 3D transformation (IV-B). Only frames with both front- and side-view pose available were merged. Subsequently, 36 (3 $(x, y, z) \times 12$ joints) body keypoint features were derived from a total of 99,942 multi-view fused poses, split into 57,708 (HOA: 28,174, PwMS: 16,210, PwPD: 13,324) and 42,234 (HOA: 13,763, PwMS: 13,572, PwPD: 14,899) poses across 32 and 26 subjects in trials W and WT (resp.). See Fig. 1 for an illustration of a fused 3D pose. Lastly, fused 3D poses were normalized based on the American median hip height [29].

## D. Validating Estimated 3D Poses Through Treadmill's CoP

*1) Validation Procedure:* The average CoP during the single support and dual support stance was calculated using the treadmill's CoP data and detected gait events. Overall, 2483 strides with 9768 (single support: 4802, dual support: 4966) valid support phases were retrieved across all subject videos in both W and WT trials and used to validate the centroid of our estimated base of support (BoS) against the treadmill's CoP.

*2) Validation Results:* Quantitatively, for single support samples, the aggregated (over all videos) mean and standard deviation of Euclidean, lateral and anterior-posterior distances (in cm) were $9.95 \pm 5.68$, $0.04 \pm 4.85$ and $0.61 \pm 8.96$ (resp.); and similarly, $8.82 \pm 5.49$, $-0.04 \pm 3.10$ and $-0.57 \pm 8.53$ (resp.), for the dual support samples. While the CoP of the participant and centroid of the BoS are not expected to be perfectly aligned, we found congruence between these measures, which helped reaffirm the validity of our estimated 3D poses.

# V. Data Analysis: Gait Feature Designs

For our ML and DL classifiers, we derived features across individual strides. This stride-wise feature extraction approach extracts multiple samples from a single subject; thus augmenting and introducing significant variations to our dataset to improve the generality of ML and DL learners. Moreover, stride-wise predictions allow for frequent and even near real-time inferences for potential clinical applications.

### A. Gait Stride Segmentation

After fusing 3D poses (IV-C), we performed automatic gait stride segmentation. In order to do so, we detected heel strikes on the right side of body (HSRs) that conventionally mark the start of every new stride. Let $[x_w^{(k)},\ y_w^{(k)},\ z_w^{(k)}]$ denote the fused 3D joint position coordinates for keypoint $k$. Then, we defined HSRs as the local minimas (at least one second apart) in the filtered right heel height series, $z_w^{(right\ heel)}$.

*1) Heel Strike Detection:* Overall, 2430 strides were retrieved from 33 (HOA: 14, PwMS: 10, PwPD: 9) subjects across three cohorts and two trials. More specifically, 1380 (HOA: 658, PwMS: 389, PwPD: 333) and 1050 (HOA: 351, PwMS: 332, PwPD: 367) strides were retrieved from 32 and 26 subjects in trials W and WT (resp.). HOA, PwMS and PwPD had on average $47.0 \pm 7.9, 38.9 \pm 8.3, 41.6 \pm 2.1$ strides and $43.9 \pm 2.8$, $36.9 \pm 9.6$, $40.8 \pm 3.9$ strides in trials W and WT (resp.). Subjects from the same cohort on average walked fewer strides in the more challenging WT task than in the W task. Healthy subjects had more strides than impaired in both the trials. Next, we discarded all the poses before the start of the first stride and after the end of the last stride. Thus out of 99,942 (W: 57,708, WT: 42,234) multi-view fused poses (in section IV-C), now, 56,226 (HOA: 26,541, PwMS: 16,187, PwPD: 13,498) and 41,747 (HOA: 13,638, PwMS: 13,448, PwPD: 14,661) poses remained across 1380 and 1050 detected strides in trials W and WT (resp.). HOA, PwMS and PwPD averaged $40.3 \pm 10.0, 41.6 \pm 9.4, 40.5 \pm 8.8$ frames and $38.9 \pm 9.2$, $40.5 \pm 8.5$, $39.9 \pm 8.8$ frames per stride in trials W and WT (resp.). A higher frame count per stride indicates a slower stride speed, i.e., HOA on average walked with an increased gait speed in both trials.

*2) Heel Strike Validation:* To quantify the performance of our HSR detection procedure, we begin with using the treadmill-recorded gait event data to mark frames with true HSRs. This required syncing video and treadmill records for each subject-trial pair. Heel strike identification segment in Fig. 1 plots a snippet of the filtered right heel height series for a PwMS with true and algorithmically detected HSRs shown in red stars and green diamonds (resp.). We define the HSR estimation error as the time gap (in seconds) between the pose-estimated HSR and the corresponding closest true HSR. The error is positive for a late and negative for an early estimate of the HSR. Heel strike validation segment in Fig. 1 depicts the frequency distribution of estimation errors across all our subjects. Overall, detected HSRs were on average $0.125 \pm 0.35$ seconds late relative to ones recorded via treadmill. In general, a good correspondence was attained between true and identified HSR markers across HOA as well as PwMS and PwPD with likely gait irregularities.

*3) Heel Strike Normalization:* Following stride segmentation, we had a varying number of poses grouped by stride. Thus we carried out temporal down sampling and smoothing (using a disjoint window-based moving average approach) in order to normalize poses to 20 per stride. Ultimately, we had 1380 (HOA: 658, PwMS: 389, PwPD: 333) and 1050 (HOA: 351, PwMS: 332, PwPD: 367) strides *(data samples)* in trials W and WT (resp.), where each stride was a $20 \times 36$-dimensional sequence with 36 body keypoint coordinates *(features/time series)* over 20 consecutive time-normalized frames *(time steps)*.

### B. Feature Designs

*1) Aggregated Pose Features:* We utilized descriptive statistical measures to aggregate our 2D ($20 \times 36$) strides along the time dimension into a vector with 91 features. These aggregated features were used with traditional ML classifiers (logistic regression, random forest, etc.) that expect a flattened feature vector as input data. In particular, to assess deviation in a stride, we compute the coefficient of variation and range for 36 joint coordinate series, each with 20 time steps; hence, obtaining 72 aggregate features. Further, to estimate mismatch in gait between left and right sides of the body, we measured asymmetry as absolute difference in the range of left and corresponding right keypoint coordinate series; thus securing 18 ($3\ (x, y, z) \times 6$ joints) more features. Finally, we included the original number of frames per stride as a feature indicative of subject's gait speed; thereby, totaling to 91 variation-, asymmetry- and speed-based characteristics in each stride to distinguish gait variations in controls from neurological population. As a result, we gathered a dataset with 2430 strides (*data samples*/*rows*) across W and WT and 91 features (*columns*) to feed into our traditional ML classifiers.

*2) Sequential Pose Features:* In contrast to traditional ML models, DL-based classifiers do take 2D sequential keypoints data directly as input; therefore, we did not carry out any additional feature engineering for our strides. This configuration did not risk losing information during the aggregation of features. Given the temporal fluctuations and irregularities in gait features within a stride, DL classifiers should be able to leverage this sequential information to generate improved predictions. Similar to the aggregated pose features, we included the original number of frames per stride as an additional feature, demonstrative of gait speed, to the model's input. This resulted in 2430 strides (*data samples*) across W and WT, each consisting of a 36-dimensional sequence over 20 consecutive time steps and scalar speed, as input for the DL algorithms.

## VI. Data Analysis: Classification and Evaluation

We utilized the designed features to classify unique gait dynamics in HOA, PwMS and PwPD on a stride-by-stride basis. We used nine traditional supervised ML algorithms to establish baseline performance: logistic regression (LR), support vector machine with linear (LSVM) and radial basis function (RBF SVM) kernels, decision tree (DT), random forest (RF), adaptive boosting (AdaBoost), eXtreme gradient boosting (XGBoost), gradient boosting machine (GBM) and multilayer perceptron (MLP). All these classifiers required 1D feature vector and thereby, the aggregated pose features (V-B1) are used as their input. Z-score normalization was applied to all aggregated features to eliminate the influence of variable feature ranges in the model's input.

## A. Deep Learning Classifiers: Convolutional Architectures

In this segment, we describe the 4 convolutional DL models used in our study. For these algorithms, temporal data with concatenated features over 20 consecutive frames (V-B2) was presented directly as input. We used Z-score normalization before feeding in data to the models.

*1) 1D Convolutional Neural Network (CNN):* Our 1D CNN model included $b$ convolutional blocks (ConvBlocks), where $b$ is a tuned hyperparameter; each ConvBlock consisted of a 1D convolutional layer (ConvLayer) followed by batch normalization, non-linear activation, dropout [30] and a pooling operation. ConvLayers take advantage of sparse connectivity and further impose local connectivity within proximate neural units, to lessen the parameters learnt as well as the chances of overfitting. We used batch normalization to standardize the input for the subsequent ConvLayer over each batch in the training process; it helps expedite training by offering some regularization. Following normalization, we applied an activation function to introduce non-linearity into ConvLayer's output neurons. A rectified linear unit (ReLU), $\mathrm{ReLU}(x) = \max(0,\, x)$, is amongst the most frequently used activation, for it does not saturate or cause vanishing gradients. Further, dropout disables neurons and their corresponding connections at random in the model with probability $p$ (hyperparameter) to help prevent overfitting during training. Additionally, a pooling (sub-sampling) layer is intermittently included in between ConvLayers to manage overfitting; max pooling preserves maximum value from a bunch of $r$ neurons, thus dividing the current dimensionality by $r$. Following these $b$ ConvBlocks, the 2D output ($l \times h$) is flattened to a vector of length either $lh$ or $h$ (via global average pooling, where we only retain the average of each feature map). The additional frames per stride feature is now concatenated with the 1D model output vector and passed through multiple DenseBlocks. Each DenseBlock consisted of a fully connected layer with a non-linear activation at the outcome, except for the last layer.

Since CNNs do not include any recurrence mechanism, we used positional encoding to explicitly add information with each pose about its corresponding order in the input stride. Specifically, let $x_t \in \mathbb{R}^{36}$ be the feature vector for the $t$-th pose ($0 \leq t < 20$) in the stride and $p_t \in \mathbb{R}^{36}$ be the corresponding positional encoding vector, then, $x'_t = x_t + p_t \;\forall\; 0 \leq t < 20$ is the upgraded embedding that is fed as input to the model. We used the sinusoidal encoding [31] that generated $p_t$ as follows:

$$p_t(j) \overset{\text{def}}{=} \begin{cases} \sin(t/10000^{2\,k/36}), & \text{if } j = 2\,k \\ \cos(t/10000^{2\,k/36}), & \text{if } j = 2k+1 \end{cases},$$

where $t \in [0, 20)$ and $j \in [0, 36)$ denote the corresponding time step and index of the feature dimension (resp.).

*2) Residual Neural Network (ResNet):* To extract more intricate features, deeper CNN networks are generally desired. However, deeper networks are increasingly challenging to train due to the *degradation issue* wherein as the model depth increases, its corresponding performance saturates and then degrades swiftly owing to a higher training error than its shallower counterpart. In theory, the deeper layers could simply be identity maps stacked to the corresponding shallow architecture to avoid any degradation in accuracy with added layers. ResNets precisely leverage this understanding and lets network layers explicitly learn residual functions relative to the layer inputs [32]. Let $g(x)$ be the expected function to be fit by a given stack of layers, where $x$ indicates the input to the first layer. The residual connection learns $f(x) \overset{\text{def}}{=} g(x) - x$ and later recasts the learnt mapping as $f(x) + x$ via element-wise addition to recover the original function $g(x)$. ResNets benefit from increased model depths by easing optimization and adding no extra computational cost.

We experimented with two kinds of residual blocks, namely, basic and bottleneck blocks. A basic block is a stack of 2 1D ConvLayers, each followed with a batch normalization and a ReLU activation. Note that the second non-linearity was applied after the element-wise addition of the input with the learnt residual mapping. A $1 \times 1$ convolution is used on the input when required to match dimensions for the element-wise addition. The deeper bottleneck block is similar in design but with 3 ConvLayers instead of 2. The $20 \times 36$ model input is first parsed using an initial ConvBlock, comprising a ConvLayer followed by batch normalization, ReLU activation and max pooling (in order), to embed features prior to residual blocks. Next, we used a stack of $b$ (hyperparameter) basic or bottleneck blocks to set up residual learning within every few layers for deeper network designs. Eventually, the 2D output is flattened via global average pooling, concatenated with frames per stride and transformed using multiple DenseBlocks to a length 3 vector. We also experimented with using positional encoding with ResNets.

*3) Multi-Scale Residual Neural Network (MSResNet):* We applied multi-scale kernel-based ResNet architecture, as proposed in [33], to derive deep-hierarchical features from multiple scales out of raw poses. MSResNet incorporates both the residual learning framework and multi-scaled convolutional kernels to address performance degradation issues and learn robust characteristics in multi-scale views from pose locations. Similar to ResNet, the input pose positions are firstly passed via a ConvLayer followed by batch normalization and ReLU activation. Next, we traversed the extracted features through three branches, each applying a different scale of convolutional kernels to acquire attributes from multiple receptive fields. Each branch is a stack of 3 basic blocks with {64, 128, 256} filters (resp.); filter lengths for ConvLayers in three different branches were set to be 3, 5, and 7 (resp.). The batch of residual blocks in all branches is followed by a global average pooling layer to reshape output features into a flattened vector. The vectors from the three branches are then concatenated into a single vector of length 768 ($= 256 \times 3$) and appended with the additional frames per stride feature; finally, this concatenated vector is fed to a fully connected network with 3 output units.

*4) Temporal Convolutional Network (TCN):* Recently introduced TCNs [34] have matched and even exceeded the performance of several recurrent models over numerous sequential modelling tasks. In general, the TCN architecture is relatively simpler, possesses longer memory to capture a more extended history and in practice, demands minimal tuning. TCN employs 1) dilated causal convolutions to process temporal data, where causality ensures no data from the future is leaked to the past and dilations assist the network to form long histories through large receptive fields, and 2) residual connections to train deeper models well. Our TCN model consisted of a stack of $n$ (hyperparameter) TCN residual blocks. Each block first learnt the residual via 2 1D dilated causal fully convolutional layers, each followed by weight normalization [35], ReLU activation and a dropout layer (in order), and then, is further succeeded by another ReLU after the element-wise addition of the input with the estimated residual mapping. A fully convolutional layer is simply a ConvLayer with an output of the same size as the input; causal convolutions ensured that output at time $t$ is convolved solely with features prior to and at time $t$ in the earlier layer. Further, a convolution with dilation factor $d$ on an element $x$ of a 1D input $g$ with filter $f$ of length $k$ is computed as $(g *_d f)(x) = \sum_{j=0}^{k-1} f(j) \,.\, g(x - d.j)$; i.e., it inserts a fixed step $d$ between every two adjacent filter taps. In practice, we set $d = 2^i$ for the $i$-th level (TCN block) of our network; this allows the receptive field size to exponentially increase relative to the depth of the network. We extracted the output from the $n$-th TCN block at the last time step, concatenated it with frames per stride and then parsed via a fully connected network to acquire the prediction output.

## B. Deep Learning Classifiers: Recurrent Architectures

We applied 3 recurrent DL models for classification of gait strides. Similar to VI-A, Z-score normalized temporal features were provided straightaway as input for these classifiers.

*1) Vanilla Recurrent Neural Network (RNN):* RNN is one of the most widely used models for apprehending dynamic information in sequential data. RNN layers utilize a chain-like arrangement of repeated units that possess hidden activation from the past to be propagated over future time steps. The current hidden state $h_t$ at the $t$-th time step is recurrently computed from the prior hidden state $h_{t-1}$ and the current input $x_t$ as $z_t = b + Wh_{t-1} + Vx_t, \quad h_t = \tanh(z_t)$, where $V$, $W$, and $b$ are trainable parameters. Our RNN model consisted of a stack of $n$ (hyperparameter) RNN layers, where each layer $i$ outputs a sequence of hidden size $s_i$ (hyperparameter) features. Along with the usual unidirectional RNN layers, where the inputs are run only from past to future, we also tried bidirectional layers in which our inputs are fed in both past to future and future to past directions. We extracted the output features from the $n$-th RNN layer at the last time step, concatenated it with frames per stride and followed with a few DenseBlocks to output the prediction probabilities. We also experimented with using a dropout layer before the DenseBlocks.

*2) Long Short-Term Memory (LSTM):* Although powerful temporal models, vanilla RNNs suffer from the *vanishing gradient problem* in longer sequences. That is, as we propagate forward in the network, small weight values for the hidden layers are multiplied together several times declining the gradients rapidly. Thus the weights for the initial layers are harder to train which in turn creates a domino effect for all further weights as well, making RNNs notably harder to train. LSTM [36], an extension of RNNs, mitigates these challenges via a memory cell with different gates to regulate the information flow into and out of the cell. Thus they are capable of handling long-short term dependencies in our gait stride inputs. Formally, an LSTM unit uses a cell state and 3 regulated gates, namely, input, forget and output gates, to add or remove information to the cell state. Each gate consists of a sigmoid layer $\sigma$, with values between 0 and 1 describing the fraction of constituents to allow in via the gate, and a point-wise multiplication operation. Our LSTM model had the same architecture as the RNN model described in VI-B1, but with RNN layers replaced with LSTM layers.

*3) Gated Recurrent Unit (GRU):* Similar to LSTMs, GRUs [37] also use a gating mechanism to address the vanishing gradient issue. However, it eliminated the cell state and has only 2 gates, namely, reset and update gates; therefore, they have fewer parameters and are a bit faster to train than LSTMs. Update gate $z_t = \sigma(W_z.[h_{t-1}, x_t])$ selects the information to add and discard in the hidden state, and reset gate $r_t = \sigma(W_r.[h_{t-1}, x_t])$ determines on how much prior information to forget based on the current input $x_t$ and past hidden state $h_{t-1}$. Again, our GRU model had the same architecture as RNN and LSTM models, but with GRU layers.

## C. Model Training and Evaluation

Our ternary (HOA, PwMS or PwPD) classification was studied across four different designs, namely, task-, subject- (W and WT) and task-subject generalization. All classifiers for task generalization were trained on 1008 (HOA: 334, PwMS: 341, PwPD: 333) gait strides in W and tested to categorize 1016 (HOA: 351, PwMS: 332, PwPD: 333; corresponding imbalance ratio being 1.0: 0.95: 0.95) strides in WT across 25 common subjects that undertook both W and WT trials. Since our data set was limited to 1380 (HOA: 658, PwMS: 389, PwPD: 333; corresponding imbalance ratio being 1.0: 0.59: 0.51) strides across 32 subjects in W trials and 1050 (HOA: 351, PwMS: 332, PwPD: 367; corresponding imbalance ratio being 1.0: 0.95: 1.05) strides across 26 subjects in WT trials, we used a 5-fold cross-validation mechanism in all classifiers for both subject generalization frameworks. Task-subject generalization also utilized 5-fold cross-validation where training splits consisted of samples from 1380 strides in W and, correspondingly, we validated on separate subjects (than in training) with samples from 1050 strides in WT. In order to prevent information leakage, we ensured that no same subject had strides split between training and validation folds. Further, given the imbalance ratio in W strides, we applied stratification in all our cross-validation setups to preserve the class distribution of the whole dataset in each generated fold.

The computations for this work were implemented on a 12 GB NVIDIA Tesla P100 GPU using PyTorch v1.7.0 DL platform in Python 3.6. In all DL algorithms, the last layer outputs $z_i$ for class $i$ were converted to normalized prediction probabilities

TABLE I
TASK GENERALIZATION: COMPARING STRIDE- AND SUBJECT-WISE TEST SET PERFORMANCE ACROSS TOP-3 ML AND DL ALGORITHMS

| | | Stride-wise evaluation metrics | | | | | Subject-wise evaluation metrics | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Algorithm | Accuracy | Precision | Recall | $F_1$ score | AUC | Accuracy | Precision | Recall | $F_1$ score | AUC |
| ML | LSVM | 0.781 | 0.784 | 0.780 | 0.780 | 0.905 | 0.960 | 0.963 | 0.963 | 0.961 | **1.0** |
| | XGBoost | 0.831 | 0.835 | 0.830 | 0.830 | 0.944 | 0.920 | 0.926 | 0.926 | 0.920 | **1.0** |
| | GBM | 0.825 | 0.834 | 0.823 | 0.824 | 0.945 | 0.960 | 0.963 | 0.963 | 0.961 | **1.0** |
| DL | ResNet | 0.876 | 0.877 | 0.876 | 0.876 | 0.972 | **1.0** | **1.0** | **1.0** | **1.0** | **1.0** |
| | MSResNet | **0.899** | **0.903** | **0.898** | **0.899** | **0.975** | **1.0** | **1.0** | **1.0** | **1.0** | **1.0** |
| | GRU | 0.862 | 0.863 | 0.861 | 0.862 | 0.961 | **1.0** | **1.0** | **1.0** | **1.0** | **1.0** |

*Note: ML is machine learning, DL is deep learning, AUC is area under the receiver operating curve; the numbers in* ***bold*** *represent the highest model performance. See section VI for details on algorithms.*

$p_i = e^{z_i} / \sum_{j=1}^{n} e^{z_j}$. These $p_i$ were then used to compute the cross entropy loss $l_{ce} = -\sum_{i=1}^{n} y_i \log p_i$, where $y_i$ is the binary truth label for class $i$. Next, the backpropagation algorithm computed the gradients of the loss relative to the weight parameters for all the layers. These gradients were used to iteratively update the weights via stochastic gradient descent optimization algorithm in order to minimize the loss function. We experimented with various optimizers including, stochastic gradient descent with and without momentum, adaptive moment estimation (Adam), Adam with decoupled weight decay (AdamW) and root mean square propagation (RMSprop), all with varying learning rate schedules and weight decay regularization. We used *early stopping* to decide optimal number of training epochs, i.e., our training stops if the validation set accuracy did not improve after $patience$ (hyperparameter) epochs. Exploratory hyperparameter optimization was performed.

In order to evaluate the prediction efficiency for the task generalization classifiers, we used the test set classification metrics, namely, precision, recall, accuracy, $F_1$ score and area under receiver operating characteristic curve (AUC), whereas for the subject- and task-subject generalization frameworks, we computed mean and standard deviation in cross-validation metrics. All models were evaluated at stride- and subject-level categorizations, where majority voting was used to classify subjects as HOA, PwMS or PwPD. Thus a correctly classified subject's video had majority of strides accurately detected as of the appropriate cohort. Precisely, we annotate the stride and subject-level performance metrics with $str$ (i.e. $\text{P}_{str}$, $\text{R}_{str}$, $\text{A}_{str}$, $\text{F1}_{str}$, $\text{AUC}_{str}$) and $sub$ (i.e. $\text{P}_{sub}$, $\text{R}_{sub}$, $\text{A}_{sub}$, $\text{F1}_{sub}$, $\text{AUC}_{sub}$) in the subscript, respectively.

# VII. EXPERIMENTAL RESULTS

## A. Prediction Models

Overall, 16 classifiers were compared to categorize strides and subjects between HOA, MS and PD cohorts for task (VII-A1), subject (VII-A2) and cross (VII-A3) generalization.

*1) Task Generalization:* Table I summarizes the stride- and subject-wise evaluation metrics for top-3 ML and DL task generalization classifiers on categorizing the test set strides of trial WT. The aggregated performance of all the subject's strides via majority voting improved upon the accuracy of individual stride-wise predictions, for instance from 83.1% to 92% on XGBoost. The top-3 DL models, viz. ResNet, MSResNet and GRU, all had perfect accuracy for classifying individuals with a given gait disorder ($\text{A}_{sub}$) and the corresponding stride-level accuracy ($\text{A}_{str}$) of 87.6%, 89.9% and 86.2% (resp.). In contrast, the top-3 ML models i.e. LSVM, XGBoost and GBM, all resulted in an $\text{A}_{str}$ of less than 85%. Analogously, the highest stride-level $\text{F}_1$ ($\text{F1}_{str}$) was 0.90 using MSResNet followed by 0.88 and 0.86 by ResNet and GRU (resp.), whereas $\text{F1}_{str}$ was lower than 0.85 applying any traditional ML approach. In Table I, MSResNet had the highest accuracy, $\text{F}_1$ and AUC of 89.9%, 0.90 and 0.98 (resp.) at stride-level, followed by ResNet and GRU with a matching perfect subject-level classification. The top task generalization algorithm was MSResNet trained for 40 epochs (as determined by the early stopping paradigm with *patience* 10) with a batch size of 128, AdamW optimizer along with a learning rate of 0.002 and a weight decay of 0.01; with nearly 2.1 million model parameters, this model took 45 minutes (min) to train and 10 seconds to evaluate on a GPU.

*2) Subject Generalization:* Table II summarizes the mean and standard deviation of 5-fold cross-validation performance metrics for the top-3 ML and DL subject generalization classifiers across W and WT trials independently. Similar to Table I, the subject-wise diagnostic performance is higher than the stride-wise measures. The top DL model, viz, ResNet for W trials and CNN for WT trials, outperformed all classical ML classifiers across all subject evaluation metrics in Table II, except AUC for WT trials. Interestingly, none of the recurrent models made it to top-3 DL algorithms for subject generalization in W. The highest-performing subject generalization algorithm for W trials was ResNet with mean accuracy, $\text{F}_1$ and AUC of 78.1%, 0.76 (class-wise $\text{F}_1$: (HOA: 0.87, PwMS: 0.8, PwPD: 0.7)) and 0.87 (resp.), at subject-level. However, the top-3 ML models, namely, LR, DT and MLP, all ended up with a mean $\text{A}_{sub}$, $\text{F1}_{sub}$ and $\text{AUC}_{sub}$ of less than 70%, 0.70 and 0.85 (resp.). Our highest-performing ResNet architecture employed positional encoding layer followed by an initial ConvBlock and 3 basic residual blocks, first with 64 filters and subsequent two with 128 filters, each with 2 ConvLayers with stride 1 and kernel sizes 8 and 5 (resp.). It was trained for 13 epochs with a batch size of

TABLE II
SUBJECT GENERALIZATION: COMPARING STRIDE- AND SUBJECT-WISE MEAN CROSS-VALIDATION PERFORMANCE ACROSS TOP-3 ML AND DL ALGORITHMS

| | | | Stride-wise evaluation metrics | | | | | Subject-wise evaluation metrics | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Algorithm | Accuracy | Precision | Recall | $F_1$ score | AUC | Accuracy | Precision | Recall | $F_1$ score | AUC |
| W | ML | LR | 0.576±0.07 | 0.565±0.06 | 0.558±0.06 | 0.542±0.05 | 0.732±0.05 | 0.690±0.17 | 0.711±0.18 | 0.70±0.21 | 0.671±0.19 | 0.806±0.13 |
| | | DT | 0.557±0.07 | 0.550±0.06 | 0.532±0.07 | 0.517±0.06 | 0.691±0.05 | 0.633±0.13 | 0.611±0.15 | 0.611±0.22 | 0.574±0.17 | 0.844±0.10 |
| | | MLP | 0.541±0.05 | 0.530±0.04 | 0.528±0.05 | 0.514±0.05 | 0.678±0.04 | 0.595±0.11 | 0.589±0.11 | 0.589±0.10 | 0.558±0.12 | 0.783±0.08 |
| | DL | CNN | 0.547±0.07 | 0.526±0.05 | 0.526±0.07 | 0.506±0.07 | 0.70±0.07 | 0.752±0.11 | 0.722±0.12 | 0.638±0.19 | 0.647±0.15 | 0.810±0.12 |
| | | ResNet | 0.523±0.05 | 0.503±0.05 | 0.504±0.05 | 0.492±0.05 | 0.680±0.05 | **0.781±0.21** | **0.789±0.22** | **0.767±0.25** | **0.758±0.24** | **0.869±0.11** |
| | | MSResNet | 0.544±0.08 | 0.501±0.07 | 0.498±0.08 | 0.492±0.08 | 0.708±0.06 | 0.686±0.14 | 0.656±0.15 | 0.644±0.20 | 0.622±0.16 | 0.724±0.12 |
| WT | ML | DT | 0.516±0.07 | 0.525±0.08 | 0.521±0.06 | 0.501±0.08 | 0.643±0.05 | 0.650±0.13 | 0.633±0.12 | 0.60±0.15 | 0.580±0.13 | **0.917±0.06** |
| | | RF | 0.514±0.12 | 0.532±0.11 | 0.508±0.13 | 0.489±0.13 | 0.707±0.13 | 0.567±0.17 | 0.567±0.17 | 0.538±0.25 | 0.514±0.22 | 0.80±0.20 |
| | | MLP | 0.546±0.17 | 0.557±0.15 | 0.547±0.16 | 0.523±0.18 | 0.734±0.16 | 0.683±0.27 | 0.70±0.24 | 0.667±0.34 | 0.640±0.31 | 0.842±0.23 |
| | DL | CNN | 0.486±0.12 | 0.479±0.12 | 0.488±0.11 | 0.470±0.13 | 0.663±0.12 | **0.750±0.17** | **0.767±0.13** | **0.711±0.21** | **0.707±0.18** | 0.771±0.17 |
| | | MSResNet | 0.513±0.07 | 0.523±0.07 | 0.503±0.06 | 0.482±0.06 | 0.709±0.05 | 0.720±0.08 | 0.70±0.12 | 0.644±0.18 | 0.631±0.14 | 0.825±0.09 |
| | | GRU | 0.522±0.10 | 0.503±0.08 | 0.519±0.08 | 0.489±0.09 | 0.687±0.08 | 0.633±0.20 | 0.633±0.16 | 0.589±0.27 | 0.571±0.22 | 0.725±0.16 |

*Note: W is walking trial, WT is walking-while-talking trial; the numbers in **bold** represent the highest* model performance in W and WT.

TABLE III
TASK-SUBJECT GENERALIZATION: COMPARING STRIDE- AND SUBJECT-WISE MEAN CROSS-VALIDATION PERFORMANCE ACROSS TOP-3 ML AND DL ALGORITHMS

| | | Stride-wise evaluation metrics | | | | | Subject-wise evaluation metrics | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Algorithm | Accuracy | Precision | Recall | $F_1$ score | AUC | Accuracy | Precision | Recall | $F_1$ score | AUC |
| ML | LR | 0.458±0.12 | 0.485±0.09 | 0.459±0.12 | 0.450±0.12 | 0.663±0.10 | 0.50±0.33 | 0.50±0.33 | 0.450±0.33 | 0.467±0.33 | 0.706±0.18 |
| | AdaBoost | 0.447±0.16 | 0.468±0.12 | 0.460±0.16 | 0.441±0.16 | 0.667±0.12 | 0.577±0.29 | 0.60±0.31 | 0.578±0.32 | 0.564±0.30 | 0.732±0.19 |
| | MLP | 0.505±0.09 | 0.506±0.10 | 0.505±0.09 | 0.486±0.10 | 0.679±0.08 | 0.507±0.16 | 0.522±0.17 | 0.444±0.23 | 0.438±0.20 | 0.710±0.09 |
| DL | CNN | 0.557±0.08 | 0.567±0.06 | 0.557±0.08 | 0.545±0.08 | 0.718±0.07 | **0.793±0.24** | **0.811±0.25** | 0.789±0.29 | **0.782±0.27** | 0.933±0.11 |
| | ResNet | 0.538±0.04 | 0.589±0.04 | 0.547±0.05 | 0.523±0.05 | 0.747±0.05 | 0.707±0.26 | 0.756±0.25 | 0.694±0.30 | 0.689±0.28 | **0.936±0.07** |
| | MSResNet | 0.561±0.09 | 0.612±0.06 | 0.566±0.07 | 0.552±0.08 | 0.748±0.06 | 0.753±0.21 | 0.789±0.19 | **0.822±0.16** | 0.760±0.20 | 0.922±0.09 |

*Note: The numbers in **bold** represent the highest model performance.*

128 and AdamW optimizer (learning rate: $0.22 \times 10^{-3}$, weight decay: 0.01); with nearly 360 K model parameters, training took around 15 min on GPU. Further, to tackle imbalance, we weighed our loss function by 0.18, 0.36 and 0.45 for strides in HOA, PwMS and PwPD (resp.). Correspondingly, the highest-performing algorithm for subject generalization in WT was CNN with mean $\mathrm{A}_{sub}$, $\mathrm{F1}_{sub}$ and $\mathrm{AUC}_{sub}$ of 75%, 0.71 (class-wise $\mathrm{F}_1$: (HOA: 0.8, PwMS: 0.9, PwPD: 0.6)) and 0.77 (resp.). The top-3 ML models, namely, DT, RF and MLP, all had mean $\mathrm{A}_{sub}$ and $\mathrm{F1}_{sub}$ less than 70%, 0.65 (resp.), however surprisingly, DT had the highest mean $\mathrm{AUC}_{sub}$ of 0.92. Our tuned CNN architecture had 2 ConvBlocks, first one having a ConvLayer with 64 filters of length 3 and stride 1 followed by batch normalization, ReLU and dropout layer with probability $p = 0.4$ and next one with a ConvLayer with 128 filters of length 2 and stride 1 followed by just ReLU activation layer. This CNN was trained for 25 epochs (35 min, 86 K parameters) with 128 samples per batch and Adam optimizer with learning rate 0.001; no weight balancing was done in this case.

*3) Task-Subject Generalization:* Table III summarizes the mean and standard deviation for stride- and subject-wise evaluation metrics of 5-fold cross-validation across top-3 ML and DL task-subject generalization classifiers. The top-3 DL models, i.e., CNN, ResNet and MSResNet, attained mean $\mathrm{A}_{sub}$ of 79.3%, 70.7% and 75.3% (resp.), and mean $\mathrm{F1}_{sub}$ of 0.78, 0.69 and 0.76 (resp.). The top-3 ML models, namely, LR, AdaBoost and MLP, all had a mean $\mathrm{A}_{sub}$ and $\mathrm{F1}_{sub}$ of less than 60% and 0.60 (resp.). A 1D CNN had the highest overall subject-wise performance for task-subject generalization with the mean $\mathrm{A}_{sub}$, $\mathrm{F1}_{sub}$ and $\mathrm{AUC}_{sub}$ of 79.3%, 0.78 (class-wise $\mathrm{F}_1$: (HOA: 0.9, PwMS: 0.9, PwPD: 0.63)) and 0.93 (resp.). This optimal CNN used positional encoding followed by 3 ConvBlocks, each having a ConvLayer with 64, 128 and 64 filters (resp.), of corresponding lengths 9, 5 and 3 and stride 1 each. Further, batch normalization and dropout with $p = 0.4$ were used in the first ConvBlock and max pooling with kernel size 2 was applied in the last ConvBlock to manage overfitting. It was trained for 20 epochs (10 min) with RMSprop optimizer (learning rate: 0.001) processing 128 samples per batch with loss function weighed by 0.1, 0.35 and 0.55 for samples belonging to HOA, PwMS and PwPD (resp.). The model had total 86 K parameters.

It is interesting to note that convolutional models were top-performers across all designs. Next, we 1) perform an ablation study to quantify the value of features from different body areas

TABLE IV
ABLATION STUDY IN SUBJECT- AND TASK-SUBJECT GENERALIZATION FRAMEWORKS

| | Subject generalization (W) | | | Subject generalization (WT) | | | Task-subject generalization | | |
|---|---|---|---|---|---|---|---|---|---|
| Data stream | Top-performing algorithm | $A_{sub}$ | $F_{1sub}$ | Top-performing algorithm | $A_{sub}$ | $F_{1sub}$ | Top-performing algorithm | $A_{sub}$ | $F_{1sub}$ |
| F | ResNet | 0.629±0.12 | 0.508±0.11 | CNN | 0.583±0.11 | 0.523±0.09 | CNN | 0.707±0.20 | 0.620±0.23 |
| F+A | CNN | 0.624±0.08 | 0.483±0.07 | ResNet | 0.650±0.24 | 0.616±0.27 | MSResNet | 0.713±0.25 | 0.674±0.29 |
| F+A+K | CNN | 0.662±0.10 | 0.620±0.11 | CNN | 0.683±0.11 | 0.627±0.18 | CNN | 0.723±0.15 | 0.673±0.18 |
| All | ResNet | **0.781±0.21** | **0.758±0.24** | CNN | **0.750±0.17** | **0.707±0.18** | CNN | **0.793±0.24** | **0.782±0.27** |

*Note: F is feet, A is ankle and K is knee features; the numbers in* ***bold*** *represent the highest* model performance.

(VII-B1), 2) analyze feature importance (VII-B.2) and 3) assess our DL predictions relative to physical performance of subjects (VII-C).

## B. Post Hoc Analysis

*1) Ablation Study:* We compared the task-, subject- and task-subject generalization performances on features from several body subsets, i.e, 18 $(= 2 \text{ (left, right)} \times 3 \text{ (2 toes, 1 heel)} \times 3\ (x,\ y,\ z))$ feet-extracted (F), 24 combined feet-ankle (F+A) and 30 feet-ankle-knee (F+A+K) coordinates, to that of using all 36 lower body features. Precisely, we studied the impact of eliminating body parts in turn as we descend from hips to feet. All ML and DL models were trained and tuned from scratch on these data streams for comparison. Table IV reports $A_{sub}$ and $\mathrm{F}_{1sub}$ for the highest-performing model on each subset with subject- and task-subject generalization schemes. DL models, specifically, CNN, ResNet and MSResNet, surpassed conventional ML performance across all data streams and model designs. Not surprisingly, the same three convolutional models were highest performers in VII-A as well. Task generalization revealed the top stride-wise performance when using all 36 features with MSResNet ($\mathrm{A}_{str}$: 90%), closely followed by F+A+K also with MSResNet ($\mathrm{A}_{str}$: 89%) and then, F+A with ResNet ($\mathrm{A}_{str}$: 83%); although, all data subsets, except using only feet features, had comparable subject-wise metrics. For subject generalization in both W and WT trials, using all features resulted in the highest mean cross-validation accuracy ($\mathrm{A}_{sub}$ in W: 78%, WT: 75%) followed by F+A+K (W: 66%, WT: 68%) and F+A (W: 62%, WT: 65%). A similar trend was noted in task-subject generalization where employing all 36 features achieved top $\mathrm{A}_{sub}$ of 79% via CNN, succeeded by F+A+K at 72% also with CNN and F+A at 71% with MSResNet. In all model frameworks, adopting entire lower body coordinates outperformed any other considered combination. Further, we saw a consistent improvement in performance as we augment additional coordinates, i.e, F $<$ F+A $<$ F+A+K $<$ All, where $<$ denotes an increase in our defined performance metrics. This indicated the importance of adding feet features to our study as their use in solidarity represented a major chunk of the overall model performance, for instance, $\mathrm{A}_{sub} = 71\%$ using only feet in comparison to 79% with all features in task-subject generalization. In conclusion, these ablation results indeed support our decision to use all lower body features for prediction.

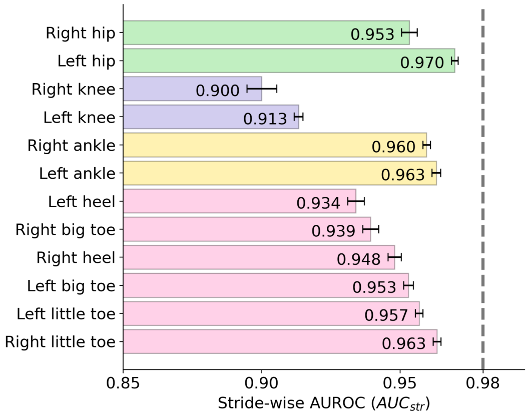

Fig. 2. Permutation feature importance in task generalization. A low score (relative to best $\mathrm{AUC}_{str}$ of 0.98) after permutation signifies more importance. Hip, knee, ankle and feet keypoints are grouped in green, violet, yellow and pink (resp.), where features are sorted in decreasing order of importance within each group.

*2) Analysis of Feature Importance:* In an attempt to *explain* and thereby, establish trust in our classifications from DL models, we examined global (via permutation feature importance) and local (via **Sh**apley **a**dditive ex**p**lanations (SHAP)) feature importance for our top models. Local feature importance focused on understanding the contribution of factors that led to a specific prediction, while global feature importance took all predictions into account.

**(i) Permutation feature importance:** Permutation feature importance measured the decrease in performance of our leading and optimally tuned DL algorithms, i.e., MSResNet for task generalization, ResNet and CNN for subject generalization in W and WT, respectively, and again, CNN for task-subject generalization, as we shuffle $(x,\ y,\ z)$ position values of an individual body part, such as right knee and left heel. This permutation cuts the association between actual feature values and the corresponding class labels. Thus a lower performance after shuffling signified the dependence of our model on the associated feature for classification and consequently, a greater importance of the respective body part. We repeated our permutation process for the test set 10 times and averaged performance metrics over these repetitions for a robust result. Fig. 2 plots the $\mathrm{AUC}_{str}$ after permuting features relative to each body part for the optimal task generalization model i.e. MSResNet.

Both knees followed by heels and big toes were the most informative features with the least $\mathrm{AUC}_{str}$ after permutation;

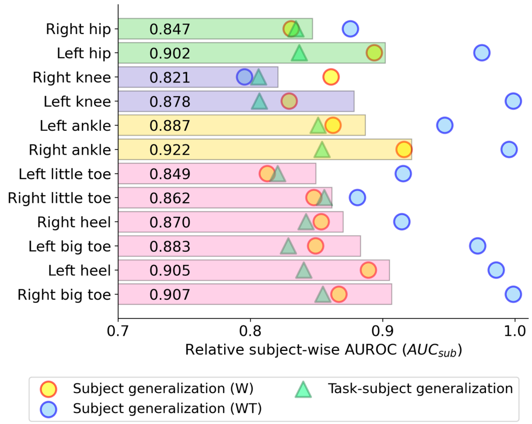

Fig. 3. Permutation feature importance in subject- and task-subject generalization. The ratio of model's AUC$_{sub}$ after permuting a feature relative to its original AUC$_{sub}$ (0.87, 0.77 and 0.93 for subject generalization in W, WT and task-subject generalization (resp.)). Yellow/blue circles and green triangles denote the ratio in subject generalization W/WT and task-subject generalization resp; bars depict the average ratio across the three designs. A lower ratio indicates higher importance. Hip, knee, ankle and feet keypoints are grouped, where features are sorted in decreasing order of importance within each group.

however, left hip positions were the least predictive of labels. Overall, we observed that right-side features were more dominant than their left-side counterparts, for instance, features from the right knee were more important than left knee. Fig. 3 plots the ratio of AUC$_{sub}$ after shuffling with respect to the original AUC$_{sub}$ of the subject- and task-subject generalization models. On average, right knee and hip followed by both little toes were the most relevant features, whereas, right ankle was the least important. It was interesting to observe that there were a few features, namely, right big toe, ankle and left knee, that had little effect on model performance for subject generalization in WT. In task-subject generalization (green triangles), all features seemed to be highly important as permuting any body part resulted in a loss of significant chunk in accuracy. This might indeed occur due to it being a highly complex classification paradigm and therefore, all features together were essential to diagnose the heterogeneity present in new subjects in an unseen trial. Altogether, knee coordinates followed by several feet features seemed to be the most important for our analysis.

**(ii) Shapley additive explanations:** SHAP [38] is based on a classic notion in game theory for optimal credit allocation, namely, Shapley values, where our classifier is considered analogous to a multi-player cooperative game with features as different players interacting together to produce the classification label as an outcome. Thus it is a local model-agnostic approach to assess feature importance by computing fair contribution of each player in our game. This kind of local explainability helps to understand individual stride characteristics that led to an accurate or erroneous prediction, which is indeed vital to facilitate targeted interventions in a medical setting. Fig. 4 applies a SHAP decision plot to depict the highest performing task-subject generalization model's (CNN) output trajectory for a single test-set stride that was correctly anticipated to belong to a PwMS.

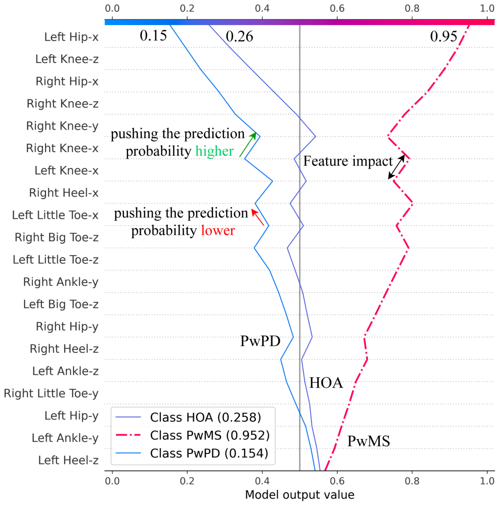

Fig. 4. SHAP for the top-performing task-subject generalization model. Multi-output decision plot for a randomly selected stride, that was correctly classified to belong to a PwMS.

The $x$-axis of this plot illustrates the model's output value, i.e., the probability of stride getting classified (vs. not) as HOA, PwMS or PwPD; the $y$-axis lists the model's top-20 features in the descending order of importance. Note that this importance is calculated only over the stride examined. SHAP essentially evaluated the affect on model performance in presence vs. absence of each feature. Moving from bottom to top, SHAP values for each feature drove the model's output from the base value (average model output over the training samples) to the overall prediction output; features pushing the model output higher increased the class prediction probability and otherwise. Observe that the outlined stride was correctly classified as from MS cohort with the highest predicted probability of 0.95, via an aggregated impact from nearly all features. This matched our remark in permutation feature importance where all body parts together were critical for task-subject generalization. Moreover, knee coordinates seemed to dominate top features in Fig. 4, similar to what we had observed in permutation feature importance.

## C. Association With Lower Extremity Function

We attempt to inspect a potential association between our top DL model predictions and the corresponding lower extremity physical function of subjects. We used the short physical performance battery (SPPB) assessment [39] as a common measure to evaluate the lower extremity functioning in all our older adults. SPPB integrates the performance of subjects in gait speed, chair stand and balance tests to create a summary score between 0 (worst) and 12 (best); lower scores indicate severe mobility limitations and higher scores indicate better performance. We had subjects with minimal to moderate frailty, i.e. SPPB: $9.85 \pm 2.35$ $[6 - 12]$, in our data. Fig. 5 visualizes the predictions made by the highest performing subject generalization in W model (ResNet) with respect to the frailty level in corresponding subjects, as measured by SPPB.

The markers, i.e. circles, squares and triangles, represent actual HOA, PwMS and PwPD, respectively, where marker

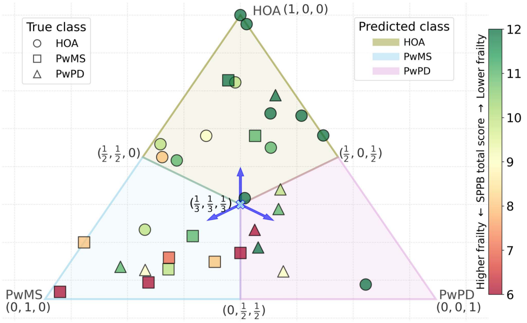

Fig. 5. Visualizing the predictions of subject generalization (W) model with respect to the corresponding lower extremity strength of subjects.

face-color shades denote the subject's SPPB. Moreover, marker positions are the barycentric coordinates representing the 3D predicted probability vector on an equilateral triangle. Consequently, our triangle is trisected in 3 equal parts, where background color depicts the predicted class by model, i.e, markers on green, blue and purple segments are predicted as HOA, PwMS and PwPD, respectively; and centroid represents an equal probability for each class. Therefore, a perfect classification would correspond to all circles in the north, squares in the south-west and triangles in the south-east vertex of the triangle. Not surprisingly, we observed that HOAs with a lower SPPB (higher frailty) had greater number of strides misclassified as belonging to MS cohort and likewise, PwMS with a higher SPPB (lower frailty) had majority of strides incorrectly predicted as from the healthy cohort. We also see from Fig. 5 that it is hard to distinguish between MS and PD subjects, which was again expected given the existent heterogeneity in these neurological disorders.

## VIII. Discussion

This study proposed a novel framework using multi-view visual data driven DL for MS and PD gait dysfunction prediction. Our system provides a convenient, low-cost, accurate, and rapid remote monitoring tool for neurological gait classification. Our architecture does not need any certified professional in charge, and being contact-less, it provides convenience and automaticity in the gait assessments of older adults in the wild. Our workflow is end-to-end open source, available at https://github.com/kaurrachneet6/VGA4MS.git. A few other works have explored vision data to categorize neurological gait [5], [19]. In contrast to our comprehensive comparison with 16 diverse models across four different designs, namely, task-, subject- (W and WT) and task-subject generalization, others [5], [19] have only examined CNN and ResNet, respectively, for a subject generalization in W trials. Further, [5] performed manual feature engineering after the extraction of 2D positions, which is now automated in our work. Our work thoroughly explored the interpretability of optimal models via post hoc analysis, which was missing in past studies. In comparison with prior work examining the binary classification of PwMS versus only controls using wearable-derivable measures [13], where 80% accuracy has been achieved when generalizing across new subjects, the ternary classification approach explored here, provides a proof of concept of a gait classification framework capable of identifying different origins of neurological gait disorders at a similar level of performance.

We observed that convolutional models were highest-performing across all generalization frameworks, which should provide guidance for future work in neurological gait classification. However, no one specific architecture was found to yield the best performance across different features, tasks, and frameworks. These findings suggested the importance of exploring different DL architectures in future work examining gait to extract as much information as possible from the input data. From a clinical perspective, stride-wise classification allowed for the use of a single stride, or brief duration walking trial, to serve as the basis for disease monitoring, which might be well suited for clinical settings with limited space and time. We used OpenPose-extracted position coordinates as input to our DL models instead of raw images as trained models with the latter might be sensitive to subject's footwear, clothing, and background, whereas OpenPose is robust to most of these factors. Our ablation study results demonstrated the importance of feet features in neurological gait classification, particularly in our task-subject generalization framework, which generalizes to new participants in different walking tasks. Further, through an analysis of permutation feature importance, we found the importance of right knee features, which might be partly due to the right-side dominance of participants in this study or positioning of video camera on right side of treadmill. SHAP visualizations (Fig. 4) provided a compact and efficient view of our model explanations to highlight relevant features for practitioners. These interpretable explanations not only helped to understand, but also trust the findings from our system.

The current study explored an automated gait screening model but the small sample size and gender differences between groups recruited for this study limits making generalized interpretations. While 3D joint coordinate trajectories used for classification were not compared with a lab-based motion capture system, prior work suggests an accuracy of 30 mm or less with removal of failed body segment recognition [40]. Since we relied on cross-validation to gauge the performance of subject- and task-subject generalization models, evaluating on a holdout data set would be essential to establish robustness. Exploration and analysis on the optimal number of continuous strides needed for best results is a crucial next step. Future research should examine inclusion of more types of pathological populations and the effect of number and position of digital cameras on the performance. Finally, evaluating the utilization of a 3D body mesh instead of sparse 3D coordinates might help improve the pose estimation block of our system. Future work might also involve exploring recent hybrid intelligence-driven and graph neural network-based approaches [41], [42].

## IX. Conclusion

The expression of neurological conditions over time and aging is heterogeneous, making the identification of sudden changes in PwMS and PwPD particularly difficult. We presented a novel vision and DL pipeline for classification of PwMS and PwPD using gait dynamics. In this study, we extracted 3D multi-view fused body keypoint positions from the recorded gait videos and demonstrated the benefits of DL architectures to differentiate neurological gait. Further, we evaluated the effectiveness of this framework to generalize across different walking tasks and subjects. Our entire code is open source and available at https://github.com/kaurrachneet6/VGA4MS.git. The studied digital camera-based framework provides a potential in-home gait monitoring tool to aid in diagnosis. This might in turn benefit both patients as well as clinicians to curtail MS and PD therapy expenses; and further facilitate low-cost and data-driven telemedicine systems for PwMS and PwPD.

## References

[1] J. Verghese et al., “Neurological gait abnormalities and risk of falls in older adults,” *J. Neurol.*, vol. 257, no. 3, pp. 392–398, 2010.

[2] J. Verghese et al., “Epidemiology of gait disorders in community-residing older adults,” *J. Amer. Geriatrics Soc.*, vol. 54, no. 2, pp. 255–261, 2006.

[3] D. J. Thurman et al., “Practice parameter: Assessing patients in a neurology practice for risk of falls (an evidence-based review): Report of the quality standards subcommittee of the american academy of neurology,” *Neurology*, vol. 70, no. 6, pp. 473–479, 2008.

[4] I. Papavasileiou, W. Zhang, X. Wang, J. Bi, L. Zhang, and S. Han, “Classification of neurological gait disorders using multi-task feature learning,” in *Proc. IEEE/ACM Int. Conf. Connected Health: Appl., Syst. Eng. Technol.*, 2017, pp. 195–204.

[5] V. Dentamaro, D. Impedovo, and G. Pirlo, “Gait analysis for early neurodegenerative diseases classification through the kinematic theory of rapid human movements,” *IEEE Access*, vol. 8, pp. 193966–193980, 2020.

[6] J. Hobart et al., “Measuring the impact of MS on walking ability: The 12-item ms walking scale (MSWS-12),” *Neurology*, vol. 60, no. 1, pp. 31–36, 2003.

[7] J. Behrens et al., “Using perceptive computing in multiple sclerosis-the short maximum speed walk test,” *J. Neuroengineering Rehabil.*, vol. 11, no. 1, pp. 1–10, 2014.

[8] T. Hanakawa et al., “Enhanced lateral premotor activity during paradoxical gait in Parkinson’s disease,” *Ann. Neurol.: Official J. Amer. Neurological Assoc. Child Neurol. Soc.*, vol. 45, no. 3, pp. 329–336, 1999.

[9] H. Zhao, Z. Wang, S. Qiu, Y. Shen, and J. Wang, “IMU-based gait analysis for rehabilitation assessment of patients with gait disorders,” in *Proc. 4th Int. Conf. Syst. Inform.*, 2017, pp. 622–626.

[10] A. P. Creagh et al., “Smartphone-and smartwatch-based remote characterisation of ambulation in multiple sclerosis during the two-minute walk test,” *IEEE J. Biomed. Health Inform.*, vol. 25, no. 3, pp. 838–849, Mar. 2021.

[11] Z. Luo et al., “Computer vision-based descriptive analytics of seniors’ daily activities for long-term health monitoring,” *Mach. Learn. Healthcare*, vol. 2, pp. 1–18, 2018.

[12] R. Kaur, S. Menon, X. Zhang, R. Sowers, and M. E. Hernandez, “Exploring characteristic features in gait patterns for predicting multiple sclerosis,” in *Proc. 41st Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2019, pp. 4217–4220.

[13] R. Kaur, Z. Chen, R. Motl, M. E. Hernandez, and R. Sowers, “Predicting multiple sclerosis from gait dynamics using an instrumented treadmill: A machine learning approach,” *IEEE Trans. Biomed. Eng.*, vol. 68, no. 9, pp. 2666–2677, Sep. 2021.

[14] A. Kalron, Z. Dvir, L. Frid, and A. Achiron, “Quantifying gait impairment using an instrumented treadmill in people with multiple sclerosis,” *Int. Scholarly Res. Notices Neurol.*, vol. 2013, 2013, Art. no. 867575.

[15] L. Comber et al., “Gait deficits in people with multiple sclerosis: A systematic review and meta-analysis,” *Gait Posture*, vol. 51, pp. 25–35, 2017.

[16] M. Psarakis et al., “Wearable technology reveals gait compensations, unstable walking patterns and fatigue in people with multiple sclerosis,” *Physiol. Meas.*, vol. 39, no. 7, 2018, Art. no. 075004.

[17] D. Xue et al., “Vision-based gait analysis for senior care,” *CoRR*, vol. abs/1812.00169, 2018. [Online]. Available: https://arxiv.org/abs/1812.00169

[18] Ł. Kidziński et al., “Deep neural networks enable quantitative movement analysis using single-camera videos,” *Nature Commun.*, vol. 11, no. 1, pp. 1–10, 2020.

[19] R. Mehrizi, X. Peng, S. Zhang, R. Liao, and K. Li, “Automatic health problem detection from gait videos using deep neural networks,” 2019. [Online]. Available: http://arxiv.org/abs/1906.01480

[20] M. D. Souza et al., “Assessment of disability in multiple sclerosis using the kinect-camera system: A. proof-of-concept study (p3. 139),” *Neurol.*, vol. 82, no. 10, Supplement, P3.139, Apr. 2014. [Online]. Available: https://n.neurology.org/content/82/10_Supplement/P3.139

[21] F. Gholami, D. A. Trojan, J. Kövecses, W. M. Haddad, and B. Gholami, “A microsoft kinect-based point-of-care gait assessment framework for multiple sclerosis patients,” *IEEE J. Biomed. Health Inform.*, vol. 21, no. 5, pp. 1376–1385, Sep. 2017.

[22] K. A. Welsh et al., “Detection of dementia in the elderly using telephone screening of cognitive status,” *Neuropsychiatry Neuropsychol., Behav. Neurol.*, vol. 6, no. 2, pp. 103–110, 1993.

[23] J. F. Kurtzke, “Rating neurologic impairment in multiple sclerosis: An expanded disability status scale (EDSS),” *Neurology*, vol. 33, no. 11, pp. 1444–1444, 1983.

[24] M. M. Hoehn et al., “Parkinsonism: Onset, progression, and mortality,” *Neurology*, vol. 50, no. 2, pp. 318–318, 1998.

[25] M. J. Socie et al., “Footfall placement variability and falls in multiple sclerosis,” *Ann. Biomed. Eng.*, vol. 41, no. 8, pp. 1740–1747, 2013.

[26] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, “OpenPose: Realtime multi-person 2D pose estimation using part affinity fields,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 172–186, Jan. 2021.

[27] T.-Y. Lin et al., “Microsoft COCO: Common objects in context,” in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2014, pp. 740–755.

[28] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: https://arxiv.org/abs/1409.1556

[29] “Definition and applicability of the anthropometric data: Hip height,” [Online]. Available: http://personal.cityu.edu.hk/meachan/Online%20Anthropometry/Chapter2/Ch2-5.htm

[30] N. Srivastava et al., “Dropout: A simple way to prevent neural networks from overfitting,” *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[31] A. Vaswani et al., “Attention is all you need,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[32] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[33] R. Liu, F. Wang, B. Yang, and S. J. Qin, “Multiscale kernel based residual convolutional neural network for motor fault diagnosis under nonstationary conditions,” *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 3797–3806, Jun. 2020.

[34] S. Bai, J. Z. Kolter, and V. Koltun, “An empirical evaluation of generic convolutional and recurrent networks for sequence modeling,” *CoRR*, vol. abs/1803.01271, 2018. [Online]. Available: https://arxiv.org/abs/1803.01271

[35] T. Salimans and D. P. Kingma, “Weight normalization: A simple reparameterization to accelerate training of deep neural networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, vol. 29, pp. 901–909.

[36] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computat.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[37] K. Cho, B. van Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” in *Proc. 2014 Conf. Empirical Methods Natural Lang. Process.*, Doha, Qatar: Association for Computational Linguistics, 2014, pp.1724–1734.

[38] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 4768–4777.

[39] J. M. Guralnik et al., “A short physical performance battery assessing lower extremity function: Association with self-reported disability and prediction of mortality and nursing home admission,” *J. Gerontol.*, vol. 49, no. 2, pp. M85–M94, 1994.

[40] N. Nakano et al., “Evaluation of 3D markerless motion capture accuracy using openpose with multiple video cameras,” *Front. Sports Act. Living*, vol. 2, 2020, Art. no. 842492. [Online]. Available: https://www.frontiersin.org/article/10.3389/fspor.2020.00050

[41] Z. Guo, Y. Shen, S. Wan, W. Shang, and K. Yu, “Hybrid intelligence-driven medical image recognition for remote patient diagnosis in internet of medical things,” *IEEE J. Biomed. Health Informat.*, early access, Dec. 31, 2021, doi: 10.1109/JBHI.2021.3139541.

[42] Z. Guo and H. Wang, “A deep graph neural network-based mechanism for social recommendations,” *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2776–2783, Apr. 2021.