














# Gaitmap—An Open Ecosystem for IMU-Based Human Gait Analysis and Algorithm Benchmarking

Arne Küderle , Martin Ullrich , Nils Roth , Malte Ollenschläger , Alzhraa A. Ibrahim ,  
Hamid Moradi , *Graduate Student Member, IEEE*, Robert Richer , *Graduate Student Member, IEEE*,  
Ann-Kristin Seifer , *Graduate Student Member, IEEE*, Matthias Zürl , Raul C. Sîmpetru , Liv Herzer ,  
Dominik Prossel, Felix Kluge , and Bjoern M. Eskofier , *Member, IEEE*

**Abstract—Goal:** Gait analysis using inertial measurement units (IMUs) has emerged as a promising method for monitoring movement disorders. However, the lack of public data and easy-to-use open-source algorithms hinders method comparison and clinical application development. To address these challenges, this publication introduces the gaitmap ecosystem, a comprehensive set of open source Python packages for gait analysis using foot-worn IMUs. **Methods:** This initial release includes over 20 state-of-the-art algorithms, enables easy access to seven datasets, and provides eight benchmark challenges with reference implementations. Together with its extensive documentation and tooling, it enables rapid development and validation of new algorithm and provides a foundation for novel clinical applications. **Conclusion:** The published software projects represent a pioneering effort to establish an open-source ecosystem for IMU-based gait analysis.

Manuscript received 4 August 2023; revised 15 November 2023 and 17 January 2024; accepted 17 January 2024. Date of publication 22 January 2024; date of current version 23 February 2024. This work was supported in part by the Innovative Medicines Initiative 2 Joint Undertaking (JU) through European Union’s Horizon 2020 Research and Innovation Program and the European Federation of Pharmaceutical Industries and Associations (EFPIA) through Mobilise-D Project under Grant 820820, in part by the Bavarian Ministry for Economy, Regional Development & Energy via the Medical Valley Award 2017 (FallRiskPD Project), and in part by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through the Project “Mobility\_APP” under Grant 438496663. The review of this article was arranged by Editor Paolo Bonato. (*Corresponding author: Arne Küderle.*)

Arne Küderle, Martin Ullrich, Nils Roth, Malte Ollenschläger, Hamid Moradi, Robert Richer, Ann-Kristin Seifer, Matthias Zürl, Raul C. Sîmpetru, Liv Herzer, Dominik Prossel, Felix Kluge, and Bjoern M. Eskofier are with the Machine Learning and Data Analytics Lab, Friedrich-Alexander Universität Erlangen-Nürnberg (FAU), 91054 Erlangen, Germany (e-mail: arne.kuederle@fau.de; martin.ullrich@fau.de; nils.roth@fau.de; malte.ollenschlaeger@fau.de; hamid.moradi@fau.de; robert.richer@fau.de; ann-kristin.seifer@fau.de; matthias.zuerl@fau.de; raul.simpetru@fau.de; liv.herzer@fau.de; dominik.prossel@kit.edu; felix.kluge@novartis.com; bjoern.eskofier@fau.de).

Alzhraa A. Ibrahim is with the Machine Learning and Data Analytics Lab, Friedrich-Alexander Universität Erlangen-Nürnberg (FAU), 91054 Erlangen, Germany, also with the Department of Molecular Neurology, FAU Erlangen, 91054 Erlangen, Germany, and also with the Computer Science Department, Faculty of Computers and Information, Assiut University, Assiut Governorate 71515, Egypt (e-mail: alzhraa.ahmed@fau.de).

This article has supplementary downloadable material available at <https://doi.org/10.1109/OJEMB.2024.3356791>, provided by the authors. Digital Object Identifier 10.1109/OJEMB.2024.3356791

We believe that this work can democratize the access to high-quality algorithm and serve as a driver for open and reproducible research in the field of human gait analysis and beyond.

**Index Terms**—Accelerometer, walking, biomarker, biomechanics, movement analysis.

**Impact Statement**—We present an elaborate open-source ecosystem for IMU-based gait analysis, including multiple algorithms, tooling, and benchmarks. This simplifies the development of applications as well as the creation and validation of new algorithms.

## I. INTRODUCTION

IN RECENT years, there has been a significant increase in research activity and advancements within the field of human gait analysis. In particular, the use of small and lightweight inertial measurement units (IMUs) has enabled the collection of gait data in a variety of settings, allowing for a wide range of applications in the fields of sports, rehabilitation, and health monitoring [1], [2], [3], [4]. Gait analysis based on IMUs often requires complex multistep algorithms to extract meaningful parameters from the recorded raw data. Multiple approaches for the various steps (e.g. stride segmentation or stride length estimation) of the analytic pipelines have been proposed in the literature [5], [6].

Unfortunately, many of the proposed algorithms are only evaluated on single datasets and are not compared to other approaches. Further, comparison metrics are often inconsistent, making it difficult to compare results across studies. To overcome that, multiple benchmark studies were performed to objectively compare algorithms for specific applications [7], [8], [9], [10], [11], [12], [13], [14], [15]. However, even the most comprehensive studies only compare a small subset of the reported algorithms. One of the core challenges is that authors need to implement most of the algorithms from scratch, as high quality source code is usually not publicly available. In combination with the lack of high quality public datasets, no conclusive algorithm ranking exists for the majority of IMU-based gait analysis tasks.

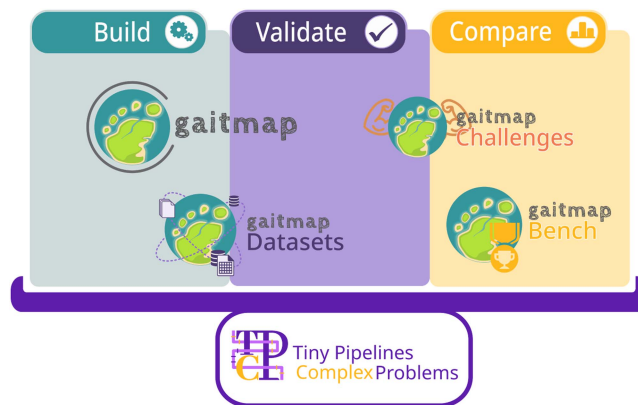
The lack of easy to use open source gait analysis algorithms leads to duplicated work across the field and a higher likelihood of implementation errors affecting published results. Further, it increases the barrier of entry into the field and prevents applications and research based on gait parameters without detailed understanding of the fundamental algorithms or access to proprietary implementations.

Mandates from public funding agencies [16], and government programs [17] might motivate some authors to share their work more openly. However, in most cases intrinsic motivation of individual researchers is the primary driving factor. Given this lack of direct incentive, the required effort often appears to be disproportionately large [18], [19], [20], [21]. Unfortunately, attempts to support researchers with the associated technical challenges are rare.

Still, notable exceptions exist in the field on IMU-based gait analysis, who published their code publicly without restrictions [14], [15], [22], [23], [24], [25], [26], [27], [28], [29]. But, the scope of most of these code publications is limited to providing the used algorithms “as-is”, with reproducibility and not reusability as the primary goal. Only a handful of projects attempt to build reusable software libraries. The most widely used is *OpenSim*, which implements a wide range of biomechanical simulation tools, including gait analysis using inverse-kinematics [30]. This approach focuses on full reconstruction of lower-limb mechanics using multi-sensor setups typically used in a laboratory setting. The *gaitpy* package [22] and its successor *scikit-digital-health* [31] provide an algorithm pipeline for hip-worn sensors focusing on fundamental spatial-temporal gait parameters. However, to the best of our knowledge, these two packages have not been widely adopted. This leaves a considerable gap in the availability of easy to use and high quality software libraries for IMU-based gait analysis, in particular for “few-node” setups applicable for long-term unsupervised monitoring.

Other scientific fields show that investing in open and well maintained tooling can have a significant impact on the scientific progress. As a prime example, *scikit-learn* [32] has revolutionized the field of machine learning by vastly simplifying the process of working with common algorithms. Its algorithm interface has become the de-facto standard for machine learning in Python, enhancing code standardization, correctness, and facilitating reuse and extension of research code by others. In addition, projects like *Kaggle*<sup>1</sup> and *Papers with Code*<sup>2</sup> provide accessible platforms for open benchmarking of machine learning methods. This makes it easier to identify and reuse the best available methods and perform meaningful comparisons of new methods.

Given the clear benefits of easy to use computational tools, we believe that a standardized and open ecosystem for IMU-based gait analysis would have a significant positive impact on the field. It would not only support the development and validation of algorithms, but also make state-of-the-art methods accessible for clinical research and support the development of end-user



**Fig. 1.** Overview of the *gaitmap* ecosystem, showing the four core packages (*gaitmap*, *gaitmap-datasets*, *gaitmap-challenges*, *gaitmap-bench*). These packages can support researchers in building gait analysis pipelines, validating pipelines and algorithms, and finally publishing and comparing reproducible benchmark results. The domain-agnostic *tcp* package is used as the foundation for all other packages.

products. With this publication and its accompanying software libraries, we hope to provide a starting-point for such an ecosystem with a focus on foot-worn IMUs. Specifically, we present the following contributions:

- 1) The *gaitmap*<sup>3</sup> library, which provides easy to use Python implementations of more than 20 algorithms for foot-worn IMUs based on 17 papers and a wide array of utility functions for working with IMU data.
- 2) The *gaitmap-datasets*<sup>4</sup> library, which provides easy access to existing public datasets.
- 3) An extensible platform (*gaitmap-challenges*, *gaitmap-bench*) to create fully reproducible benchmarks for gait analysis algorithms.<sup>5</sup> Currently, it provides eight concrete benchmark challenges with reference implementations.

## II. MATERIALS AND METHODS

### A. Overview Over the Gaitmap Ecosystem

The presented ecosystem consists of four core packages that all build on top the domain-agnostic *tcp* package [33] (Fig. 1). All packages are implemented in Python and follow modern best-practices in regard to structure and software-development and are build on top of widely used scientific Python packages (NumPy [33], SciPy [34], pandas [35], scikit-learn [32], Optuna [36], pomegranate [37], and tslearn [38]). The source of all *gaitmap* packages is published via Github ([github.com/mad-lab-fau/gaitmap](https://github.com/mad-lab-fau/gaitmap)) and their extensive documentation is hosted on ReadTheDocs ([gaitmap.readthedocs.io](https://gaitmap.readthedocs.io)).

The *gaitmap* package provides core algorithms for gait analysis using foot-worn IMUs aligned along a pipeline inspired by Rampff et al. [39] (Fig. 2). For each of the steps in the pipeline *gaitmap* provides a number of interchangeable algorithms (Table I). Each algorithm is implemented as a Python

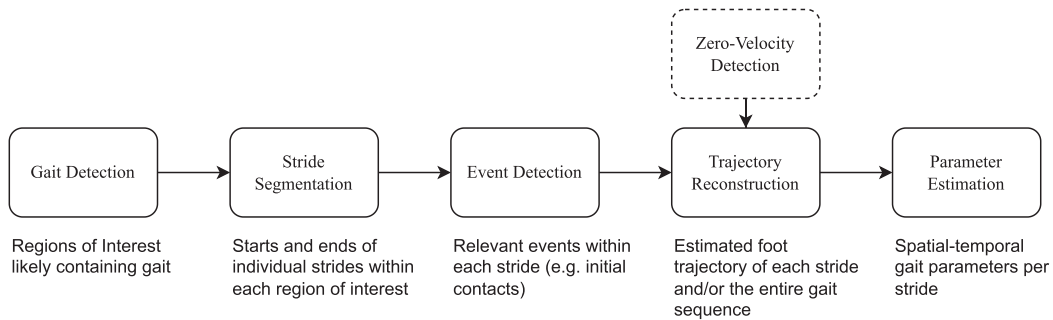
<sup>1</sup><https://www.kaggle.com>

<sup>2</sup><https://paperswithcode.com>

<sup>3</sup><https://github.com/mad-lab-fau/gaitmap>

<sup>4</sup><https://github.com/mad-lab-fau/gaitmap-datasets>

<sup>5</sup><https://github.com/mad-lab-fau/gaitmap-bench>



**Fig. 2.** Overview of the main gait analysis pipeline that can be built using the *gaitmap* package. The pipeline is inspired by Rampp et al. [39] with *Gait Detection* as an additional step. After the gait detection, each algorithm is expected to work on one of the detected regions of interests and has access to the results of the previous steps. The text under each box provides a short description of the output of each step.

class, providing attributes to configure the algorithm parameters (Appendix 1, (Supplementary material)). The algorithms can be used independently or combined into a pipeline to run an end-to-end analysis. For each step users can substitute algorithms with their own custom implementations. Beyond the core algorithms, the package provides a number of utility methods to preprocess data, simplify the development of custom algorithms, and to evaluate the output of algorithms against ground truth references.

These evaluation methods are used in the *gaitmap-challenges* package, which provides a standardized framework to run and evaluate gait analysis pipelines on specific challenges. Each benchmark challenge is either based on a train-test split or a cross-validation approach, allowing for proper comparison of machine learning and traditional algorithms alike. These challenges can be run fully locally on any *gaitmap*-compatible implementation to enable quick validation and standardized comparison of algorithms. All predefined challenges use *gaitmap-datasets* as the interface to load and standardize the publicly available datasets they are based on.

Finally, *gaitmap-bench* presents a collection of reproducible scripts implementing various challenges of the *gaitmap-challenges* package for a number of different algorithms. The results of these scripts are published on the *gaitmap-bench* website,<sup>6</sup> which provides a visual overview of the results and further information about the respective challenges. New scripts and results for other public algorithms can be submitted by anyone via GitHub.

## B. Data Standardization

To ensure that algorithms can be compared across multiple datasets, we propose standardization guidelines for the input and output datatypes. The two main aspects of this standardization concern the used units and coordinate systems. Regarding the units, we suggest using the International System of Units (SI) wherever possible (Table II). We only make an exception for angular values, where we suggest using degrees instead of radians, as they are more comprehensible. For the orientation of the sensor or the body/foot, we use a *quaternion* representation.

Regarding coordinate systems, the optimal choice will depend on the sensor position. For foot/leg-mounted IMUs we propose to differentiate between three coordinate systems: (1) the sensor frame, (2) the body frame, and (3) the world (or global) frame. The sensor frame is created by the physical axes of the IMU sensor. We expect that the  $x$  axis is pointing roughly forward, the  $y$  axis to the left, and the  $z$  axis upwards relative to the body of a participant. The body frame is defined by the three main body axes, namely  $ml$  (medial to lateral),  $pa$  (posterior to anterior), and  $si$  (superior to inferior) (Fig. 3). Directions of rotations are defined based on the International Society of Biomechanics (ISB) recommendations for the ankle joint [54]. The resulting body frame coordinate systems of the left and right foot are mirror images of each other. In result, the same signals can be expected for the same anatomical movements independent of the limb. This is beneficial when performing any form of event detection on the signal. Finally, the world frame is used to report the global displacement or change of orientation of the sensor or body over a period of time.

## C. Benchmarking

Validating and benchmarking is essential during algorithm development. The presented ecosystem makes this process simple and standardized by (1) providing scoring functions specific for gait parameters to compare algorithm results against a ground truth, (2) providing primitives to perform parameter optimization and cross-validation, and (3) providing a set of challenges that allow benchmarking algorithms on a specific dataset with a single method call (Appendix 1, (Supplementary material)). Together, this decreases the chance of implementation errors and reduces the workload for comparing algorithms across multiple datasets.

At the time of writing, eight of these challenges are provided (Table III) and reference implementations and results for all *gaitmap* algorithms exist as part of the *gaitmap-bench* repository. In this manuscript, we present two of these challenges as an example in more detail. Further information and detailed results can be found on the *gaitmap-bench* website<sup>7</sup> and Appendix 2, (Supplementary material).

<sup>6</sup><https://gaitmap-bench.readthedocs.io>

<sup>7</sup><https://gaitmap-bench.readthedocs.io>

**TABLE I**  
OVERVIEW OF THE CORE ALGORITHMS PROVIDED IN THE *GAITMAP* PACKAGE

Processing Step	Algorithm	Description
Preprocessing	<code>align_dataset_to_gravity</code>	Alignment of IMU signal with gravity.
	<code>align_heading_of_sensors</code>	Align the heading to the movement direction using PCA on the gyroscope signal and an integration of the acceleration signal (unpublished, inspired by [41]).
Gait Detection	<code>UllrichGaitSequenceDetection</code>	Detection of gait using harmonic frequencies from raw IMU signals [42]*.
Stride Segmentation	<code>BarthDtw</code>	Template matching using dynamic time warping (DTW) modified based on [43]*.
	<code>ConstrainedBarthDtw</code>	<code>BarthDtw</code> with additional local warping constraints. Inspired by local weightings explained in [44].
	<code>HmmStrideSegmentation</code>	Multi-activity Hidden-Markov-Model (HMM) based on [45]* implemented using [38].
Event Detection	<code>RampEventDetection</code>	Initial and terminal foot contact detection based on signal features [40]*.
	<code>FilteredRampEventDetection</code>	<code>RampEventDetection</code> with additional lowpass filtering for specific calculation steps.
	<code>HerzerEventDetection</code>	Initial and terminal foot contact based on signal features tuned for stair walking [46]*. Combines concepts from [40]* and [47].
Zero-Velocity Detection	<code>AredZuptDetector</code>	Window based thresholding on gyroscope norm to detect static regions [48].
	<code>NormZuptDetector</code>	Window based thresholding to detect static regions. Generalized version of ARED from [48].
	<code>ShoeZuptDetector</code>	Window based thresholding on combined gyroscope and accelerometer norm to detect static regions [48].
Trajectory Reconstruction	<code>RtsKalman</code>	Zero-Velocity aided Error Tracking Kalman Filter with smoothing to estimate the foot trajectory from IMU signals. Based on [49], [50] with implementation details from [51].
	<code>MadwickRtsKalman</code>	Variation of the <code>RtsKalman</code> method using the Madgwick algorithms for orientation estimation (unpublished, based on [24], [49]–[51]).
	<code>SimpleGyroIntegration</code>	Orientation estimation based on integration of the gyroscope signal.
	<code>MadgwickAHRS</code>	Sensor fusion algorithm for orientation estimation [24].
	<code>ForwardBackwardIntegration</code>	Position estimation using double-integration of the acceleration with dedrifting through forward-backwards integration [52] and [9]*.
	<code>PieceWiseLinearDedrifiedIntegration</code>	Position estimation using double-integration of the acceleration with linear drift model (unpublished, inspired by [53]).
Parameter Estimation	<code>TemporalParameterCalculation</code>	Calculation of temporal parameters (e.g. stride time) based on the detected events.
	<code>SpatialParameterCalculation</code>	Calculation of spatial parameters (e.g. stride length) based on the calculated trajectory [40], [54]*.

The name in the Algorithm column corresponds to the class/function name within the package. The last column lists the publications that the algorithm is based on. Algorithms marked with an asterisk (\*) are publications that were published by the authors of this manuscript directly or the associated research group. Besides the core algorithms, the package provides a number of helper methods, which are not listed here. For a full list see the official documentation page ([gaitmap.readthedocs.io](http://gaitmap.readthedocs.io)). The categories are listed in the order they are typically used in a gait analysis pipeline.

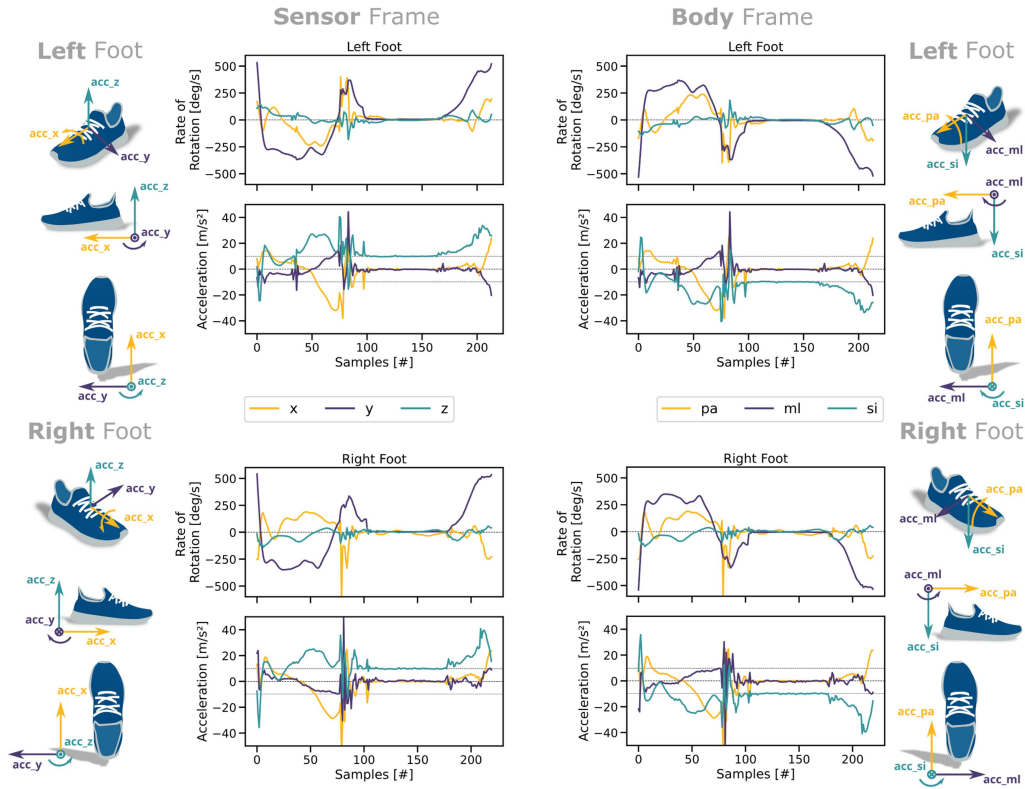
**TABLE II**  
OVERVIEW OF THE UNITS USED IN THE *GAITMAP* PACKAGE

Metric	Unit
Distance	Meter (m)
Time	Seconds (s)
Acceleration	Meter per second squared ( $m/s^2$ )
Angular Velocity	Degrees per second (deg/s)
Orientation	Unit quaternion in $(x, y, z, w)$ order

**1) Example 1 – Stride Segmentation:** We use the dataset by Barth et al. [42] (30 participants performing a  $4 \times 10$  m walk and 15 participants performing a 4 min free-walk) to benchmark

the extraction of individual strides from a continuous signal of foot-worn IMUs. We consider a stride as detected (“true-positive”), if both the start and end label are within 30 ms of the hand-labeled ground truth. Based on this output we calculated the precision, recall, and F1-score. The validation is performed within a 5-fold cross-validation to allow for parameter optimization on an independent training set.

On this challenge we compare the three stride segmentation algorithms implemented in *gaitmap* (*DTW*, *Constrained-DTW*, *HMM*). We tested each algorithm once with its default parameters and once with parameters automatically optimized for the dataset. For the *HMM* this involved retraining the underlying



**Fig. 3.** Illustration of the sensor and the body frame with example signals from an individual stride (toe-off to toe-off) of a healthy participant recorded with a foot-worn IMU sensor at 204.8 Hz. The direction of the arrows indicate the movement/rotation direction resulting in a positive sensor reading. The coordinate system for the sensor frame (left) is defined by the three axes  $x$ ,  $y$ , and  $z$  of the sensor. The coordinate system for the body frame (right) is defined by the main body axes  $ml$  (medio-lateral),  $pa$  (posterior-anterior), and  $si$  (superior-inferior) of the foot.

model from scratch. See Appendix 2, (Supplementary material) for more details.

**2) Example 2 – Full Pipeline Validation:** To evaluate the representative performance of entire gait analysis pipelines, we compare aggregated parameters over an entire gait test. These types of comparisons represent more closely the error ranges expected during actual usage of a gait analysis system and, hence, are an important addition to detailed validation of individual algorithms [57].

The specific challenge we present in this manuscript uses the dataset recorded by Kluge et al. [56] containing recordings of  $4 \times 10$  m walk tests at different speeds from 11 healthy and 4 patients with Parkinson’s disease. On each  $4 \times 10$  m recording we validate the estimated mean gait parameters over all strides against the motion capture (MoCap) reference using standard error metrics. In this manuscript the mean-absolute-error (MAE) of the mean gait speed within a gait test is presented as the main comparison metric. The evaluation is performed within a 5-fold cross-validation. For this challenge we compare the results of one pipeline (“Classic”) closely resampling the one presented in [56] and one pipeline (“Modern”) based on more recent algorithms presented in [24], [44], [45], [48], [49]. Note that both pipelines are not specifically optimized for this dataset, so the evaluation using cross-validation is not required. However, to ensure comparability with future algorithms, we still perform the evaluation this way. See Appendix 2, (Supplementary material) for more details.

### III. RESULTS AND DISCUSSION

#### A. Example Benchmarks

**1) Example 1 – Stride Segmentation:** Comparing the different algorithms, the HMM appears to be the best overall approach (Fig. 4). It is able to provide good results, even without retraining the model for the specific dataset, and in the retrained version it shows the best performance overall. However, compared to the other algorithms, the HMM experiences a larger drop in performance going from the  $4 \times 10$  m to the free-walking condition (Fig. 4, left). This highlights the need for validating algorithms under different conditions.

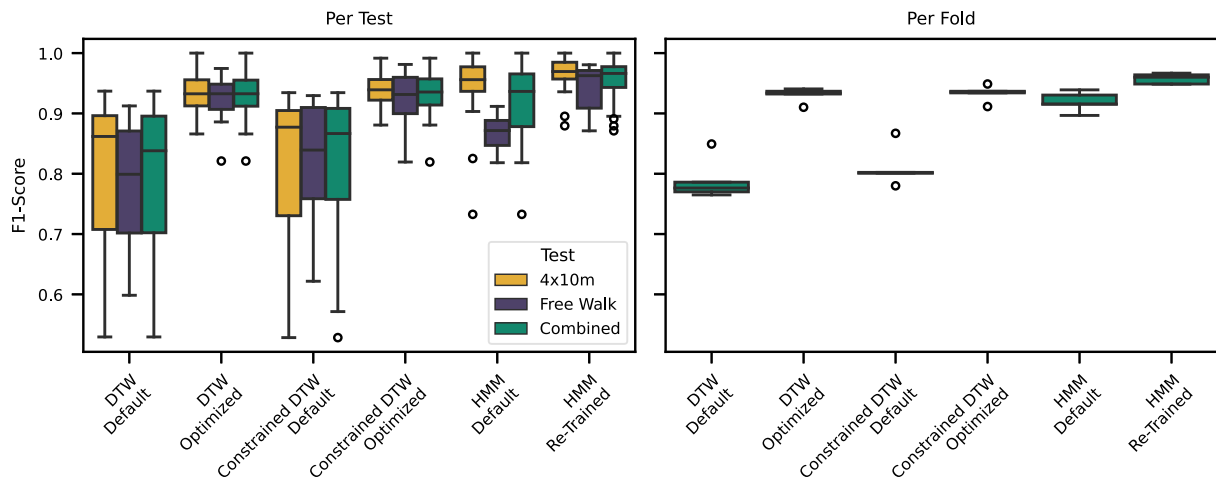
For all algorithms, optimizing the algorithmic parameters for the dataset increases the median F1-score and reduces the variation across data points. This underlines the importance of considering the hyperparameter optimization as part of an algorithmic approach. Fair comparison of algorithms is only possible if all algorithms are actively optimized for the specific task.

In their optimized/re-trained form, the algorithms implemented in *gaitmap* provide state-of-the-art performance with median F1-scores above 90%. However, the actual value differs between the per test and the per fold aggregation (Fig. 4). This shows that how exactly an error value is derived matters, and if they are not calculated absolutely identically, results can not be compared.

**TABLE III**  
OVERVIEW OF THE BENCHMARK CHALLENGES CURRENTLY IMPLEMENTED IN THE *GAITMAP-CHALLENGES* PACKAGE

Processing Step	Dataset	Description
Stride Segmentation	EgaitSegmentationValidation2014 [43]	Algorithms need to identify individual strides based on the definition by Barth et al. [43] during supervised 4x10m gait tests and unstructured simulated free-living activities. The sensors are mounted laterally at the shoes.
	EgaitSegmentationValidation2014 [43] (relabelled)	<b>Same as the previous challenge, but in addition to straight strides the algorithms also need to identify strides during stair walking and turns.</b>
	SensorPositionComparison2019 [56]	Algorithms need to identify individual strides based on the definition by Barth et al. [43] during various walking tasks in the laboratory. Two synchronized IMU sensors are mounted on the instep of the shoe.
Spatial Parameter	EgaitAdidas2014 [54]	Algorithms need to estimate the stride length of strides performed at different speeds within the capture volume of a MoCap system. Two sensors are mounted laterally on the shoe and predefined stride borders and events are provided.
	EgaitParameterValidation2013 [40]	Algorithms need to estimate the stride length of strides performed by geriatric patients while walking over a GAITRite carpet. Two sensors are mounted laterally on the shoe and predefined stride borders and events are provided.
	SensorPositionComparison2019 [56]	Algorithms need to estimate the stride length of strides performed at different speeds within the capture volume of a MoCap system. Two synchronized IMU sensors are mounted on the instep of the shoe and predefined stride borders and events are provided.
Full Pipeline	SensorPositionComparison2019 [56]	Algorithms need to estimate the mean and standard deviation of the stride length, gait velocity, and stride time of each of the performed gait tests. Only the raw IMU data of the two sensors mounted on the insteps of the shoes is provided.
	KlugeParameterValidation2017 [57]	<b>Algorithms need to estimate the mean and standard deviation of the stride length, gait velocity, and stride time of each of the performed 4x10m tests. Only the raw IMU data of the two sensors mounted laterally on the shoes is provided.</b>

The two challenges with descriptions highlighted in bold are presented in more detail in this manuscript. The Dataset column provides the class name of the dataset within the *gaitmap-datasets* package.

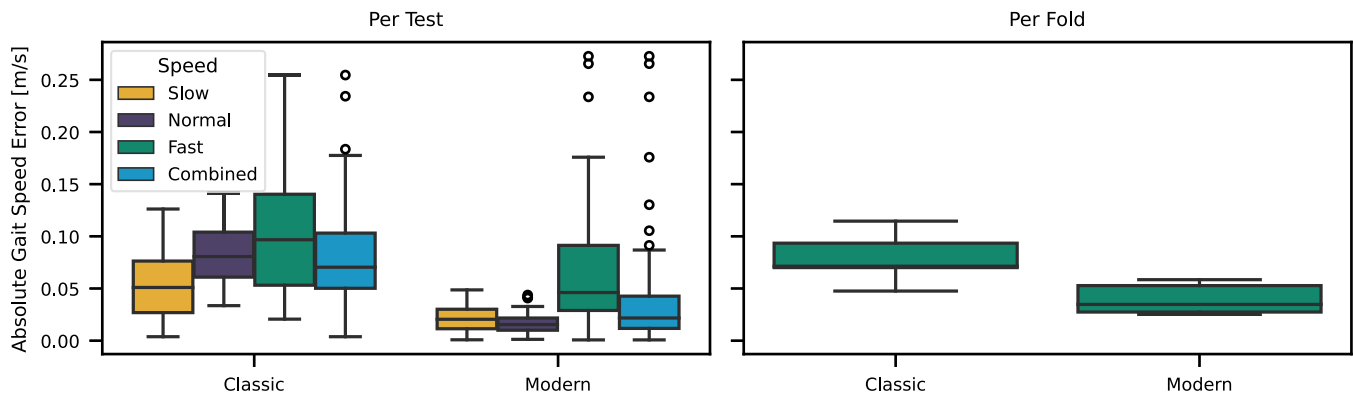


**Fig. 4.** F1-score of the stride segmentation benchmark (Example 1) comparing all algorithms implemented in *gaitmap* in their default and optimized/re-trained version. In the left plot, each data point represents one gait test (either  $4 \times 10$  m or 4-min walk). The data points are combined across all test-sets of the 5-fold cross-validation. In the right plot, each data point represents one test-set of the 5-fold cross-validation. All strides across all gait tests within each fold are grouped together before calculating the F1-score.

**2) Example 2 – Full Pipeline Validation:** The “Modern” pipeline outperforms the “Classic” pipeline independent of the aggregation method and speed condition (Fig. 5). However, the speed dependency shows that combining the error to a single value (per fold aggregation) is not sufficient to fully understand the performance of the algorithms. Both pipelines show a speed dependency of the error, but the “Modern” pipeline has additional outliers at higher gait speeds, which are not present

in the “Classic” pipeline. This indicates the need for in-depth analysis of individual data points to fully understand potential shortcomings of an algorithm.

The results from just these two example benchmarks show that a complete and fair comparison of algorithms requires full understanding of the comparison pipeline, access to algorithms’ source code for optimization, and the ability to look at algorithm results in multiple ways and at multiple aggregation levels to



**Fig. 5.** Absolute gait speed error of the full pipeline benchmark (Example 2) comparing the “Classic” and the “Modern” pipeline. In the left plot, each data point represents one  $4 \times 10$  m gait test at different speeds. The data points are combined across all test-sets of the 5-fold cross-validation. Boxplots are provided for each gait speed and all data points combined. In the right plot, each data point represents one test-set of the 5-fold cross-validation. The absolute errors of the individual tests are averaged over each fold.

understand the strengths and weaknesses of specific approaches. All of that is only possible, if data and code are shared fully and openly in an accessible way.

### B. Using and Extending the Ecosystem

One of the core aims of the presented ecosystem is to provide direct value to as many communities as possible. Hence, we split our software ecosystem into multiple independent open-source software packages with an increasing level of specificity. Ideally, this results in a larger user base of the more general packages, increasing the chance of building stable and sustainable communities around them. For example, the *tcp* package [33], which was developed as the foundation for all the other packages, has already been used in the fields of digital psychology [58] and is being used in further applications fields our research group.

The *gaitmap* library itself has been used internally for the past years and was the basis for the data analysis in at least nine papers [44], [55], [59], [60], [61], [62], [63], [64] and numerous student theses. Furthermore, the developed algorithm pipelines are used by project partners in industry and clinical research. Based on these experiences, we are confident that the presented ecosystem will be useful far beyond our research group. To make adoption as seamless as possible, we provide extensive documentation and tutorials and are open to provide support.

As all of our libraries are built with extensibility in mind, they can be adapted for individual needs. All core algorithms are sensor-agnostic and only require that the data can be standardized into the correct format. Custom algorithms, datasets, and challenges can be added without modifying the core libraries and the provided benchmarks can be run fully locally to compare custom proprietary algorithms with presented baselines. To accommodate cases, where publishing algorithms and data openly is not possible, we ensured that the full ecosystem can be used (from a technical and legal perspective) for research purposes without the need of publishing your own code. This makes it possible to build (internal) toolboxes and workflows on top of our work. For example, we successfully integrated the *gaitmap* algorithms into a custom plugin for the *MaD-GUI* [65] to allow

clinicians at the University Hospital Erlangen to analyze data from routine clinical gait recordings.

In case researchers can contribute their work back to the community, they can do so by adding their algorithms, datasets, or challenges to the libraries of the *gaitmap* ecosystem. Alternatively, they can use the *gaitmap* libraries as dependencies for their own packages and publish them independently, while still benefiting from the standardization and tooling we provide.

Beyond that, we are confident that our work demonstrates many best practices for developing scientific code and Python packages, and can therefore be used as a template for other projects.

### C. Limitations

To ensure the quality of the presented libraries, we decided to narrow the scope of this initial release to foot-worn IMUs. Given the current directions of the field of human gait analysis, similar projects including hip-worn or wrist-worn IMUs could be of high interest for the community. For this, the presented ecosystem could be extended to include new sensor positions or could serve as a blueprint for independent projects in the future.

Similarly, we focused on algorithms that were already used in our research group. This does not represent the entire field and the provided algorithms will likely not be the best for all use cases. Regarding datasets, we were limited by our attempt to build a fully open system. Most of the largest and interesting datasets are not publicly available. In particular, the inclusion of home-monitoring/real-world datasets and datasets including diverse participants with pathological gait would increase the usefulness of the presented benchmarks. We plan to extend the list of included algorithms and datasets in the future, and we believe that the visibility of this project can motivate others to support us in this process. In particular, recent promising machine learning based methods [14], [66], [67], [68], [69] are not included in this initial release. One of the main challenges for any open source ecosystem is to maintain and update the packages long term. With a strong focus on good software practices, testing, and documentation, we provide the foundation

for that. However, the only sustainable long term solution is to have an active community around the presented ecosystem to make it independent of individual developers and funding sources. Building such a community is a difficult task, and we do not know yet how to best achieve this.

Regarding all the presented limitations, we are looking forward to open discussions within the community to improve and extend the presented ecosystem, and hope that this publication can motivate more open collaboration on computational tools.

#### IV. CONCLUSION

This manuscript and the published code libraries represent the most comprehensive attempt yet to create an open ecosystem for analyzing human gait using IMUs and benchmarking the associated algorithms. The *gaitmap* ecosystem supports the development, evaluation, and usage of IMU-based gait analysis algorithms at every stage. This empowers researchers to write less and simpler code, allowing them to concentrate on their primary research questions. Moreover, it decreases the perceived burden of creating reproducible and publishable research code, facilitating broader evaluations and comparisons of algorithms across diverse datasets.

For clinical researchers and application developers this publication provides access to high quality implementations of state-of-the-art algorithms, which can be used to perform data analysis and build end-user applications for clinical and sports applications, reducing the required expertise and effort to do so.

In the long term, we hope the *gaitmap* ecosystem can grow into a community supported and independent project, facilitating open and reproducible research in human gait analysis and beyond.

#### AUTHOR CONTRIBUTIONS

AK lead the development of the *gaitmap* ecosystem, developed many of the core algorithms, and wrote the initial draft of the manuscript. AK, MU, MO, NR, AI, RR, MZ, FK, and BE were involved in planning and structuring the core software packages and the associated literature research. AK, MU, MO, NR, AI, FK, HM, RS, DP, LH, AS, MZ provided significant contributions to package development, the included algorithms, and documentation writing. All authors contributed to the writing of the manuscript and approved the final version.

#### CONFLICT OF INTEREST

The authors declare no conflicts of interest.

#### SUPPLEMENTARY MATERIALS

The supplementary material provides code examples on how to use the *gaitmap-challenges* library (Appendix 1) and further details on how the example benchmarks were performed (Appendix 2).

#### ACKNOWLEDGMENT

Content in this publication reflects the authors' view and neither IMI nor the European Union, EFPIA, or any Associated Partners are responsible for any use that may be made of the information contained herein.

#### REFERENCES

- [1] C. Buckley et al., "The role of movement analysis in diagnosing and monitoring neurodegenerative conditions: Insights from gait and postural control," *Brain Sci.*, vol. 9, no. 2, Feb. 2019, Art. no. 34. [Online]. Available: <https://www.mdpi.com/2076-3425/9/2/34>
- [2] Y. Celik, S. Stuart, W. Woo, and A. Godfrey, "Gait analysis in neurological populations: Progression in the use of wearables," *Med. Eng. Phys.*, vol. 87, pp. 9–29, Jan. 2021. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1350453320301697>
- [3] S. D. Din, A. Godfrey, C. Mazzà, S. Lord, and L. Rochester, "Free-living monitoring of Parkinson's disease: Lessons from the field: Wearable technology for Parkinson's disease," *Movement Disord.*, vol. 31, no. 9, pp. 1293–1313, Sep. 2016. [Online]. Available: <https://doi.wiley.com/10.1002/mds.26718>
- [4] B. Eskofier et al., "An overview of smart shoes in the Internet of Health Things: Gait and mobility assessment in health promotion and disease monitoring," *Appl. Sci.*, vol. 7, no. 10, Sep. 2017, Art. no. 986. [Online]. Available: <https://www.mdpi.com/2076-3417/7/10/986>
- [5] R. Caldas, M. Mundt, W. Potthast, F. B. de L. Neto, and B. Markert, "A systematic review of gait analysis methods based on inertial sensors and adaptive algorithms," *Gait Posture*, vol. 57, pp. 204–210, Sep. 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0966636217302424>
- [6] D. Jarchi, J. Pope, T. K. M. Lee, L. Tamjidi, A. Mirzaei, and S. Sanei, "A review on accelerometry-based gait analysis and emerging clinical applications," *IEEE Rev. Biomed. Eng.*, vol. 11, pp. 177–194, 2018.
- [7] V. N. Bobic, M. D. Djuric-Jovieic, S. M. Radovanovic, N. T. Dragaevic, V. S. Kostic, and M. B. Popovic, "Challenges of stride segmentation and their implementation for impaired gait," in *Proc. 40th Ann. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2018, pp. 2284–2287. [Online]. Available: <https://ieeexplore.ieee.org/document/8512836/>
- [8] N. Haji Ghassemi et al., "Segmentation of gait sequences in sensor-based movement analysis: A comparison of methods in Parkinson's disease," *Sensors*, vol. 18, no. 2, Jan. 2018, Art. no. 145. [Online]. Available: <https://www.mdpi.com/1424-8220/18/1/145>
- [9] J. Hannink, M. Ollenschläger, F. Kluge, N. Roth, J. Klucken, and B. M. Eskofier, "Benchmarking foot trajectory estimation methods for mobile gait analysis," *Sensors*, vol. 17, no. 9, Aug. 2017, Art. no. 1940. [Online]. Available: <https://www.mdpi.com/1424-8220/17/9/1940>
- [10] A. Soltani et al., "Algorithms for walking speed estimation using a lower-back-worn inertial sensor: A cross-validation on speed ranges," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1955–1964, 2021. [Online]. Available: <https://ieeexplore.ieee.org/document/9534902/>
- [11] M. Caruso et al., "Analysis of the accuracy of ten algorithms for orientation estimation using inertial and magnetic sensing under optimal conditions: One size does not fit all," *Sensors*, vol. 21, no. 7, Apr. 2021, Art. no. 2543. [Online]. Available: <https://www.mdpi.com/1424-8220/21/7/2543>
- [12] I. Skog, J. O. Nilsson, and P. Händel, "Evaluation of zero-velocity detectors for foot-mounted inertial navigation systems," in *Proc. Int. Conf. Indoor Positioning Indoor Navigation*, 2010, pp. 1–6.
- [13] G. P. Panebianco, M. C. Bisi, R. Stagni, and S. Fantozzi, "Analysis of the performance of 17 algorithms from a systematic review: Influence of sensor position, analysed variable and computational approach in gait timing estimation from IMU measurements," *Gait Posture*, vol. 66, pp. 76–82, Oct. 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0966636218304004>
- [14] B. Wagstaff, V. Peretroukhin, and J. Kelly, "Robust data-driven zero-velocity detection for foot-mounted inertial navigation," *IEEE Sensors J.*, vol. 20, no. 2, pp. 957–967, Jan. 2020. [Online]. Available: <https://ieeexplore.ieee.org/document/8852673/>
- [15] M. E. Micó-Amigo et al., "Assessing real-world gait with digital technology? Validation, insights and recommendations from the Mobilise-D consortium," *J. NeuroEngineering Rehabil.*, vol. 20, no. 1, Jun. 2023, Art. no. 78, doi: [10.1186/s12984-023-01198-5](https://doi.org/10.1186/s12984-023-01198-5).
- [16] European Commission, Directorate-General for Research and Innovation, Horizon Europe, open science — Early knowledge and data sharing, and open collaboration, Publications Office of the European Union, 2021, doi: [10.2777/18252](https://doi.org/10.2777/18252).
- [17] C. Gentemann, "Why NASA and federal agencies are declaring this the year of open science," *Nature*, vol. 613, no. 7943, pp. 217–217, Jan. 2023. [Online]. Available: <https://www.nature.com/articles/d41586-023-00019-y>
- [18] C. L. Borgman, "The conundrum of sharing research data," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 63, no. 6, pp. 1059–1078, 2012. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/asi.22634>

- [19] V. Stodden, "The scientific method in practice: Reproducibility in the computational sciences," MIT Sloan Research Paper No. 4773-10, Feb. 9, 2010. [Online]. Available: <http://dx.doi.org/10.2139/ssrn.1550193>
- [20] R. J. LeVeque, I. M. Mitchell, and V. Stodden, "Reproducible research for scientific computing: Tools and strategies for changing the culture," *Comput. Sci. Eng.*, vol. 14, no. 4, pp. 13–17, Jul. 2012. [Online]. Available: <https://ieeexplore.ieee.org/document/6171147/>
- [21] J. Swarts, "Open-source software in the sciences: The challenge of user support," *J. Bus. Tech. Commun.*, vol. 33, no. 1, pp. 60–90, Jan. 2019, doi: [10.1177/1050651918780202](https://doi.org/10.1177/1050651918780202).
- [22] M. Czech and S. Patel, "GaitPy: An open-source python package for gait analysis using an accelerometer on the lower back," *J. Open Source Softw.*, vol. 4, no. 43, Nov. 2019, Art. no. 1778. [Online]. Available: <https://joss.theoj.org/papers/10.21105/joss.01778>
- [23] R. D. Gurchiek et al., "Open-source remote gait analysis: A post-surgery patient monitoring application," *Sci. Rep.*, vol. 9, no. 1, Nov. 2019, Art. no. 17966. [Online]. Available: <https://www.nature.com/articles/s41598-019-54399-1>
- [24] S. O. Madgwick, A. J. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *Proc. IEEE Int. Conf. Rehabil. Robot.*, 2011, pp. 1–7. [Online]. Available: <https://ieeexplore.ieee.org/document/5975346/>
- [25] J.-O. Nilsson, I. Skog, P. Handel, and K. Hari, "Foot-mounted INS for everybody—An open-source embedded implementation," in *Proc. IEEE/ION Position Location Navigation Symp.*, 2012, pp. 140–145. [Online]. Available: <https://ieeexplore.ieee.org/document/6236875/>
- [26] I. A. Vajs, V. N. Bobic, M. D. Duric-Jovicic, and M. M. Jankovic, "Open-source application for real-time gait analysis using inertial sensors," in *Proc. 28th Telecommun. Forum*, 2020, pp. 1–4. [Online]. Available: <https://ieeexplore.ieee.org/document/9306636/>
- [27] C. Ensink, K. Smulders, J. Warnar, and N. Keijsers, "Sint maartenskliniek - IMU gait analysis (validation study, release tag v1.1.0)," Jul. 2023. [Online]. Available: <https://zenodo.org/record/8198714>
- [28] J. Trautmann et al., "TRIPOD with IMU, pressure-distribution and photoelectric data for gait analysis," *Data*, vol. 6, no. 9, Sep. 2021, Art. no. 95. [Online]. Available: <https://www.mdpi.com/2306-5729/6/9/95>
- [29] U. Gottlieb, T. Balasukumaran, J. R. Hoffman, and S. Springer, "Agreement of gait events detection during treadmill backward walking by kinematic data and inertial motion units," *Sensors*, vol. 20, no. 21, Jan. 2020, Art. no. 6331. [Online]. Available: <https://www.mdpi.com/1424-8220/20/21/6331>
- [30] M. Al Borno et al., "OpenSense: An open-source toolbox for inertial-measurement-unit-based measurement of lower extremity kinematics over long durations," *J. NeuroEngineering Rehabil.*, vol. 19, no. 1, Feb. 2022, Art. no. 22. [Online]. Available: <https://doi.org/10.1186/s12984-022-01001-x>
- [31] L. Adamowicz, Y. Christakis, M. D. Czech, and T. Adamusiak, "SciKit digital health: Python package for streamlined wearable inertial sensor data processing," *JMIR mHealth uHealth*, vol. 10, no. 4, Apr. 2022, Art. no. e36762.
- [32] F. Pedregosa et al., "Scikit-learn: Machine learning in python," Jun. 2018. [Online]. Available: <https://arxiv.org/abs/1201.0490>
- [33] C. R. Harris et al., "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020. [Online]. Available: <https://www.nature.com/articles/s41586-020-2649-2>
- [34] P. Virtanen et al., "SciPy 1.0: Fundamental algorithms for scientific computing in python," *Nature Methods*, vol. 17, no. 3, pp. 261–272, Mar. 2020. [Online]. Available: <https://www.nature.com/articles/s41592-019-0686-2>
- [35] W. McKinney, "Data structures for statistical computing in python," in *Proc. 9th Python Sci. Conf.*, 2010, pp. 56–61.
- [36] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," Jul. 2019. [Online]. Available: <https://arxiv.org/abs/1907.10902>
- [37] J. Schreiber, "Pomegranate: Fast and flexible probabilistic modeling in python," *J. Mach. Learn. Res.*, vol. 18, no. 164, pp. 1–6, 2018.
- [38] R. Tavenard et al., "Tslearn, a machine learning toolkit for time series data," *J. Mach. Learn. Res.*, vol. 21, no. 118, pp. 1–6, 2020. [Online]. Available: <https://jmlr.org/papers/v21/20-091.html>
- [39] A. Rapp, J. Barth, S. Schuelein, K.-G. Gassmann, J. Klucken, and B. M. Eskofier, "Inertial sensor-based stride parameter calculation from gait sequences in geriatric patients," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 4, pp. 1089–1097, Apr. 2015. [Online]. Available: <https://ieeexplore.ieee.org/document/6949634/>
- [40] W. Kong, S. Sessa, M. Zecca, and A. Takahashi, "Anatomical calibration through post-processing of standard motion tests data," *Sensors*, vol. 16, no. 12, Dec. 2016, Art. no. 2011. [Online]. Available: <https://www.mdpi.com/1424-8220/16/12/2011>
- [41] M. Ullrich et al., "Detection of gait from continuous inertial sensor data using harmonic frequencies," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 7, pp. 1869–1878, Jul. 2020.
- [42] J. Barth et al., "Stride segmentation during free walk movements using multi-dimensional subsequence dynamic time warping on inertial sensor data," *Sensors (Switzerland)*, vol. 15, no. 3, pp. 6419–6440, Mar. 2015. [Online]. Available: <https://www.mdpi.com/1424-8220/15/3/6419>
- [43] M. Müller, *Fundamentals of Music Processing Using Python and Jupyter Notebooks*, 2nd ed. Berlin, Germany: Springer, 2021.
- [44] N. Roth et al., "Hidden Markov model based stride segmentation on unsupervised free-living gait data in Parkinson's disease patients," *J. NeuroEngineering Rehabil.*, vol. 18, no. 1, Dec. 2021, Art. no. 93. [Online]. Available: <https://jneuroengrehab.biomedcentral.com/articles/10.1186/s12984-021-00883-7>
- [45] L. Herzer, "Gait event detection algorithms for free-living stair ambulation," Bachelor Thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg, Jun. 2021. [Online]. Available: <https://www.mad.tf.fau.de/person/liv-herzer-2/>
- [46] J. Figueiredo, P. Felix, L. Costa, J. C. Moreno, and C. P. Santos, "Gait event detection in controlled and real-life situations: Repeated measures from healthy subjects," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 10, pp. 1945–1956, Oct. 2018. [Online]. Available: <https://ieeexplore.ieee.org/document/8452990/>
- [47] I. Skog, P. Handel, J.-O. Nilsson, and J. Rantakokko, "Zero-velocity detection," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 11, pp. 2657–2666, Nov. 2010.
- [48] D. Simon Colomar, J. O. Nilsson, and P. Handel, "Smoothing for ZUPT-aided INSs," in *Proc. Int. Conf. Indoor Positioning Indoor Navigation*, 2012, pp. 13–15.
- [49] C. Tunca, N. Pehlivan, N. Ak, B. Arnrich, G. Salur, and C. Ersoy, "Inertial sensor-based robust gait analysis in non-hospital settings for neurological disorders," *Sensors (Switzerland)*, vol. 17, no. 4, Apr. 2017, Art. no. 825. [Online]. Available: <https://www.mdpi.com/1424-8220/17/4/825>
- [50] J. Solà, "Quaternion kinematics for the error-state Kalman filter," 2017. [Online]. Available: <https://arxiv.org/abs/1711.02508>
- [51] M. Zok, C. Mazzà, and U. Della Croce, "Total body centre of mass displacement estimated using ground reactions during transitory motor tasks: Application to step ascent," *Med. Eng. Phys.*, vol. 26, no. 9, pp. 791–798, Nov. 2004. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1350453304001195>
- [52] N. Kitagawa and N. Ogihara, "Estimation of foot trajectory during human walking by a wearable inertial measurement unit mounted to the foot," *Gait Posture*, vol. 45, pp. 110–114, Mar. 2016. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0966636216000151>
- [53] C. M. Kanzler et al., "Inertial sensor based and shoe size independent gait analysis including heel and toe clearance estimation," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2015, pp. 5424–5427.
- [54] G. Wu et al., "ISB recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motionpart I: Ankle, hip, and spine," *J. Biomech.*, vol. 35, no. 4, pp. 543–548, Apr. 2002. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0021929001002226>
- [55] A. Küderle, N. Roth, J. Zlatanovic, M. Zrenner, B. Eskofier, and F. Kluge, "The placement of foot-mounted IMU sensors does affect the accuracy of spatial parameters during regular walking," *PLoS One*, vol. 17, no. 6, Jun. 2022, Art. no. e0269567. [Online]. Available: <https://journals.plos.org/plosone/article?id=, doi: 10.1371/journal.pone.0269567>.
- [56] F. Kluge, H. Gaßner, J. Hannink, C. Pasluosta, J. Klucken, and B. M. Eskofier, "Towards mobile gait analysis: Concurrent validity and test-retest reliability of an inertial measurement system for the assessment of spatio-temporal gait parameters," *Sensors*, vol. 17, no. 7, Jul. 2017, Art. no. 1522. [Online]. Available: <https://www.mdpi.com/1424-8220/17/7/1522>
- [57] C. Kirk et al., "Mobilise-D insights to estimate real-world walking speed in multiple conditions with a wearable device," *Sci. Rep.*, vol. 14, 1754, 2024. [Online]. Available: <https://doi.org/10.1038/s41598-024-51766-5>
- [58] R. Richer, L. Abel, A. Küderle, B. M. Eskofier, and N. Rohleder, "CARWatch a smartphone application for improving the accuracy of cortisol awakening response sampling," *Psychoneuroendocrinol.*, vol. 151, May 2023, Art. no. 106073. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0306453023000513>

- [59] N. Roth, A. Küderle, D. Prossel, H. Gassner, B. M. Eskofier, and F. Kluge, "An inertial sensor-based gait analysis pipeline for the assessment of real-world stair ambulation parameters," *Sensors*, vol. 21, no. 19, Jan. 2021, Art. no. 6559. [Online]. Available: <https://www.mdpi.com/1424-8220/21/19/6559>
- [60] N. Roth et al., "Real-world stair ambulation characteristics differ between prospective fallers and non-fallers in Parkinson's disease," *IEEE J. Biomed. Health Inform.*, vol. 26, no. 9, pp. 4733–4742, Sep. 2022. [Online]. Available: <https://ieeexplore.ieee.org/document/9808112/>
- [61] M. Ullrich et al., "Detection of unsupervised standardized gait tests from real-world inertial sensor data in Parkinson's disease," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 2103–2111, 2021. [Online]. Available: <https://ieeexplore.ieee.org/document/9567680/>
- [62] M. Ullrich et al., "Fall risk prediction in Parkinson's disease using real-world inertial sensor gait data," *IEEE J. Biomed. Health Inform.*, vol. 27, no. 1, pp. 319–328, Jan. 2023. [Online]. Available: <https://ieeexplore.ieee.org/document/9924545/>
- [63] V. Koch et al., "Inertial sensor acquired foot elevation parameters validated by motion capture in a cohort of patients with hereditary spastic paraplegias," *Gait Posture*, vol. 100, pp. 79–80, Mar. 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0966636222007081>
- [64] H. Moradi, N. Roth, A.-K. Seifer, and B. M. Eskofier, "Detection of distorted gait and wearing-off phenomenon in Parkinson's disease patients during levodopa therapy," in *Proc. IEEE-EMBS Int. Conf. Biomed. Health Inform.*, 2022, pp. 01–04. [Online]. Available: <https://ieeexplore.ieee.org/document/9926873/>
- [65] M. Ollenschläger et al., "MaD GUI: An open-source python package for annotation and analysis of time-series data," *Sensors*, vol. 22, no. 15, Jan. 2022, Art. no. 5849. [Online]. Available: <https://www.mdpi.com/1424-8220/22/15/5849>
- [66] R. Romijnders, E. Warmerdam, C. Hansen, G. Schmidt, and W. Maetzel, "A deep learning approach for gait event detection from a single shank-worn IMU: Validation in healthy and neurological cohorts," *Sensors*, vol. 22, no. 10, Jan. 2022, Art. no. 3859. [Online]. Available: <https://www.mdpi.com/1424-8220/22/10/3859>
- [67] V. Guimarães, I. Sousa, and M. V. Correia, "A deep learning approach for foot trajectory estimation in gait analysis using inertial sensors," *Sensors*, vol. 21, no. 22, Nov. 2021, Art. no. 7517. [Online]. Available: <https://www.mdpi.com/1424-8220/21/22/7517>
- [68] J. Hannink, T. Kautz, C. F. Pasluosta, K.-G. Gasmann, J. Klucken, and B. M. Eskofier, "Sensor-based gait parameter extraction with deep convolutional neural networks," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 1, pp. 85–93, Jan. 2017. [Online]. Available: <https://ieeexplore.ieee.org/document/7778173/>
- [69] D. Weber, C. Gühmann, and T. Seel, "Neural networks versus conventional filters for inertial-sensor-based attitude estimation," May 2020. [Online]. Available: <https://arxiv.org/abs/2005.06897>