# Assessing Kinematic Methods of Structuring Gait

## Overview

This tutorial is a comprehensive walkthrough of a research investigation presented at the Ontario Biomechanics Conference (OBC) 2025.

The study, titled “Assessing Different Kinematic Methods of Structuring Gait”, explores how gait events can be reliably detected using only kinematic data, without access to lab-based kinetic systems such as force plates or instrumented treadmills.

**Research Question: Are kinematic event detection methods able to reliably structure biomechanical waveforms in the gait cycle?**

The tutorial has been designed for biomechanics researchers, students, and clinicians interested in alternative gait cycle structuring approaches, especially in field-based or markerless motion capture settings. Pipelines for all event detection methods are provided.

## Introduction

In gait analysis, the ability to structure biomechanical waveforms into gait cycles is needed for analyzing different kinematic signals and calculating various gait measures.

Traditionally, gait cycles are structured using kinetic event detection methods based on force thresholds, which provide an objective and reliable signal for detecting ON (HS) and OFF(TO). However, kinetic data acquisition is not always available and with more in-field data collection it is not feasible.

Given these limitations, researchers have developed a range of kinematic event detection algorithms that rely on marker trajectories or derived joint kinematics to infer gait events. While these methods are promising, their accuracy and robustness - especially under varying walking speeds and among diverse populations- needs validation.

The aim of this tutorial is to present a step-by-step guide for using HAS-Motion software tools, specifically Visual3D, to process gait data and compare several kinematic-based methods against the kinetic-based gold standard.

## Dataset Description

**A public dataset of overground and treadmill walking kinematics and kinetics in healthy individuals**

Claudiane A Fukuchi [1], Reginaldo K Fukuchi [2], Marcos Duarte [1, 2]

Affiliations + expand

PMID: 29707431 PMCID: PMC5922232 DOI: 10.7717/peerj.4640

Link to Article

This investigation uses the publicly available dataset from Fukuchi et al. (2018), which provides overground and treadmill walking data for 42 healthy adults.

The dataset consists of 24 younger participants (mean age ~27.6 years) and 18 older particpants (mean age ~62.7 years). Each subject completed both treadmill and overground walking trials across a range of speeds. For each trial, 3D kinematic data was captured using a 26-marker lower-body configuration at a sampling rate of 150 Hz, and ground reaction forces were recorded from an instrumented treadmill at 300 Hz. Metadata for each participant includes height, weight, segment length, sex, leg dominance, and treadmill speed.

This tutorial focuses exclusively on **treadmill trials** and analyzes only **left-side events** (LHS and LTO). This simplifies gait cycle definitions to the interval between two identical events on the same foot, which is essential for consistent waveform normalization.

## Sample Data Download and Contents

To facilitate learning and application, we provide a premade ZIP archive containing all required files.

- Includes a Visual3D preprocessing pipeline (**Preprocessing_Pipeline**) that performs kinetic event detection and computes the necessary joint kinematics.
- **PREPROCESSED_PARTICIPANTS**: Preprocessed CMZ files for all 42 participants with the preprocessing pipelines applied.
- **FULL_PROCESSED_PARTICIPANTS**: Fully processed CMZ files for all 42 subjects where all methods and events have already been applied to the data.
- **METHOD PIPELINES + MASTER**: Folder containing individual Visual3D pipeline scripts for each kinematic method (e.g., Zeni Method 1+2, DeAsha, Hreljac, etc.) and a master pipeline script that applies all methods and computes the metric of cycle duration.
- **WBDSmodel.mdh**: Original model file that defines the lower limb segments using the 26 anatomical markers.
- **AutoFormatted_Gait_Data_Final**: Excel sheet containing resultant exported data from processing - listing method, trial and cycle instance+duration. This format is used for more accessible manipulation during analysis in Python.

Copies of the literature for the various kinematic methods used are also included in the ZIP archive:

- Zeni et al. (2018) Method 1 and 2
- DeAsha et al. (2012)
- Hreljac & Marshall (2000)
- OConnor et al. (2007)

# Workflow Outline

1. **Preprocessing in Visual3D**

The first stage involves preprocessing all participant data using the **Preprocessing_Pipeline.v3s** script. This step takes treadmill-only trials for each participant and puts them into a single CMZ file and computes joint kinematic data.

During this stage, ground reaction force (GRF) data is used to automatically detect kinetic gait events, establishing a **baseline** for later comparison. Subject-specific anthropometric data (height and weight) are manually input from the provided spreadsheet.

2. **Understanding and Applying Kinematic Method Pipelines**

The next phase involves applying several published and custom kinematic methods to detect gait events without relying on kinetic data.

Each method is built using the Visual3D pipeline script to extract features such as foot position, velocity, hip extension, and marker accelerations.

On top of the 9 methods (where HS and TO events are evaluated in their own methods, respectively), 2 additional methods were devised for this investigation:

- One based on hip surpassing heel
- the other on peak knee flexion angle, offering alternative kinematic cues not commonly discussed in literature.

3. **Apply and Generate Measures to Compare all Methods**

Once individual method pipelines were validated, they were consolidated into a master script (**FinalPipeline_ALL_METHODS_SEQUENCES.v3s**). This pipeline computes all gait event across all methods simultaneously and then defines gait cycles based on those events.

These durations are stored under the METRIC::PROCESSED:: folder in the application.

4. **Exporting to Python for Statistical Evaluation**

Finally, the computed cycle durations are exported to an Excel file where each row represents one gait cycle instance.

The structured data is then analyzed in Python using a Linear Mixed Model (LMM). In this model, the event detection method is treated as the fixed effect, while participant ID and trial number are modeled as nested random effects.

This statistical framework allows us to access the consistency and reliability of each method in reproducing gait cycle structure comparable to kinetic-based baselines.

# Preprocessing Using Visual3D

The first part of the analysis involves applying the preprocessing pipeline (Preprocessing_Pipeline.v3s). This script automates the loading of treadmill trials from each participant, builds a subject-specific CMZ file by combining trials, and applies a model to compute segment lengths, joint angles, and kinetic-based gait events.

The following shows the steps to complete the preprocessing. If you prefer to start with the already preprocessed folder, you may skip to the **NEXT SECTION**.

1. Open an empty workspace on the Visual3D application, and navigate to the pipeline workshop dialog.

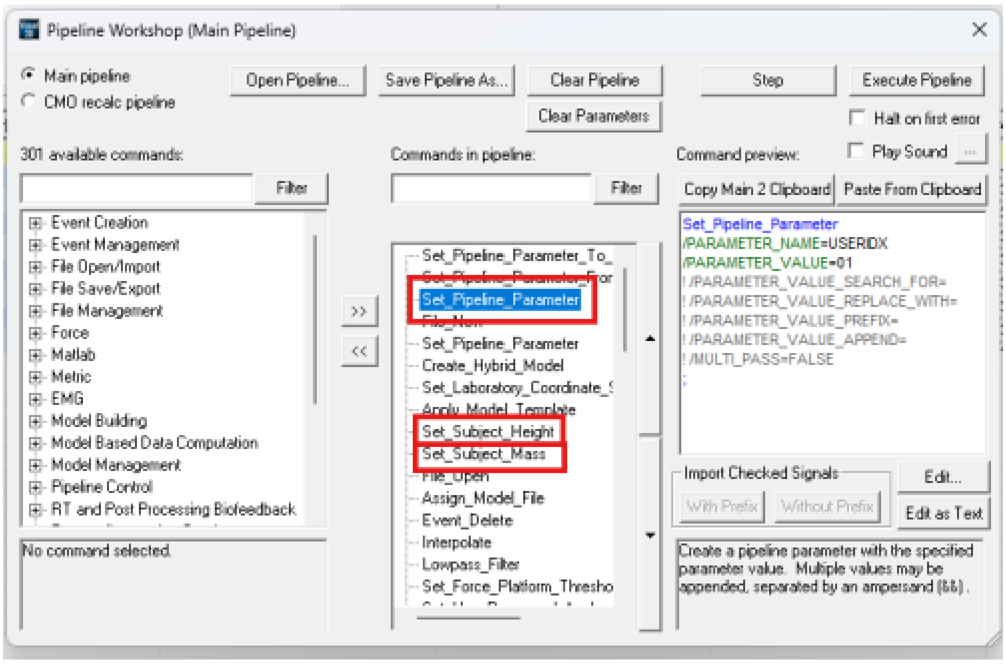
2. Click **Open Pipeline** and browse to select the Preprocessing_Pipeline.v3s file included in the ZIP.

3. This pipeline will be run once for each participant. The following changes must be applied prior to executing the pipeline each time:

- **Set_Pipeline_Parameter** command: /PARAMETER_VALUE= [PARTICIPANT NO.]
- **Set_Subject_Height** and **Set_Subject_Weight** to specific values found on Subject_Info excel sheet from sample data.

For example, for PARTICIPANT #1:

```
Set_Pipeline_Parameter
/PARAMETER_NAME=USERIDX
**/PARAMETER_VALUE=01**
! /PARAMETER_VALUE_SEARCH_FOR=
! /PARAMETER_VALUE_REPLACE_WITH=
! /PARAMETER_VALUE_PREFIX=
! /PARAMETER_VALUE_APPEND=
! /MULTI_PASS=FALSE
;
```




4. Now click **Execute Pipeline**, this will prompt the user to select the folder in which the raw c3d files are saved, browse for the **WBDSc3dWithGaitEvents** folder included within the original dataset.

5. Next, you will be prompted to select a model file - **WBDSmodel.mdh**.

6. Once the pipeline is executed, you will be prompted to save the participant CMZ to a location on your computer. Make sure they are placed within the same folder.

# Understanding and Applying Kinematic Methods

Each kinematic method was implemented based on the procedures outlined in their original literature references. These methods rely on features like joint angles, foot velocities, and segment positions. Below are summaries of each method and how they were adapted into pipeline commands:

**LHS/LTO Foot Position (Zeni Method 1) :** Heelstrike and toe off events are detected by identifying the peak forward and backward positions of the foot (specifically heel and toe markers) relative to the pelvis. The relative positions are calculated using **Compute_Model_Based_Data**, and event detection is done via **Event_Maximum** and **Event_Minimum** commands.

**LHS/LTO Foot Velocity (Zeni Method 2):** Method looks for zero-crossings in the foot velocity signal relative to the pelvis. HS is the point when the foot velocity changes from forward to backward; TO is the reverse. This is implemented by calculating the first derivation of the position signal, followed by the **Event_Threshold** detection.

**LHS Hip Extension (DeAsha Method):** Initial Contact (heelstrike) is inferred from the contralateral hip reaching its maximum extension. The hip flexion-extension angle is computed using the pelvis and thigh segments, and the event is marked using **Event_Minimum** commands on the sagittal plane angle.

**LHS/LTO Toe Acceleration (Hreljac Method):** This method uses peaks in heel and toe marker acceleration, confirmed by corresponding zero-crossings in jerk (derivative of acceleration).

**LHS/LTO Heel Velocity (OConnor Method):** Foot velocity is derived from filtered heel and toe marker trajectories. Events are identified by checking for local minima combined with constraints such as the vertical position of the heel being below 35% of its range.

**Hip Surpassing Heel Method:** An event is detected when the leading hip joint center passes the position of the opposite foot, signaling a change in support limb.

**Knee Angle Max Method**: A simpler method that structures cycles based on successive local maxima of the knee flexion signal, using **Event_Maximum**

Each of these pipeline scripts define events AND compute the cycle duration using this events by placing them in sequences. The table below shows a summary of these methods:

| **Method Name** | **Events Produced** | **Description** |
|---|---|---|
| Foot Position - Zeni Method 1 | LHS, LTO | Uses anterior-posterior position of the foot. |
| Foot Velocity - Zeni Method 2 | LHS, LTO | Identifies local minimum in foot's anterior-posterior velocity/ |
| Hip Extension - DeAsha Method | LHS | Points of peak hip extension. |
| Toe Acceleration - Hreljac Method | LHS, LTO | Peak in vertical acceleration of toe marker. |
| Heel Velocity - OConnor Method | LHS, LTO | Identifies local minimum in local heel velocity. |
| Hip Surpassing Heel | LHip_Passes_Heel | Identifies moment the leading hip joint center overtakes the opposite foot. |
| Max Knee Angle | LKnee_Angle_Max | Peak knee flexion-extension angle. |

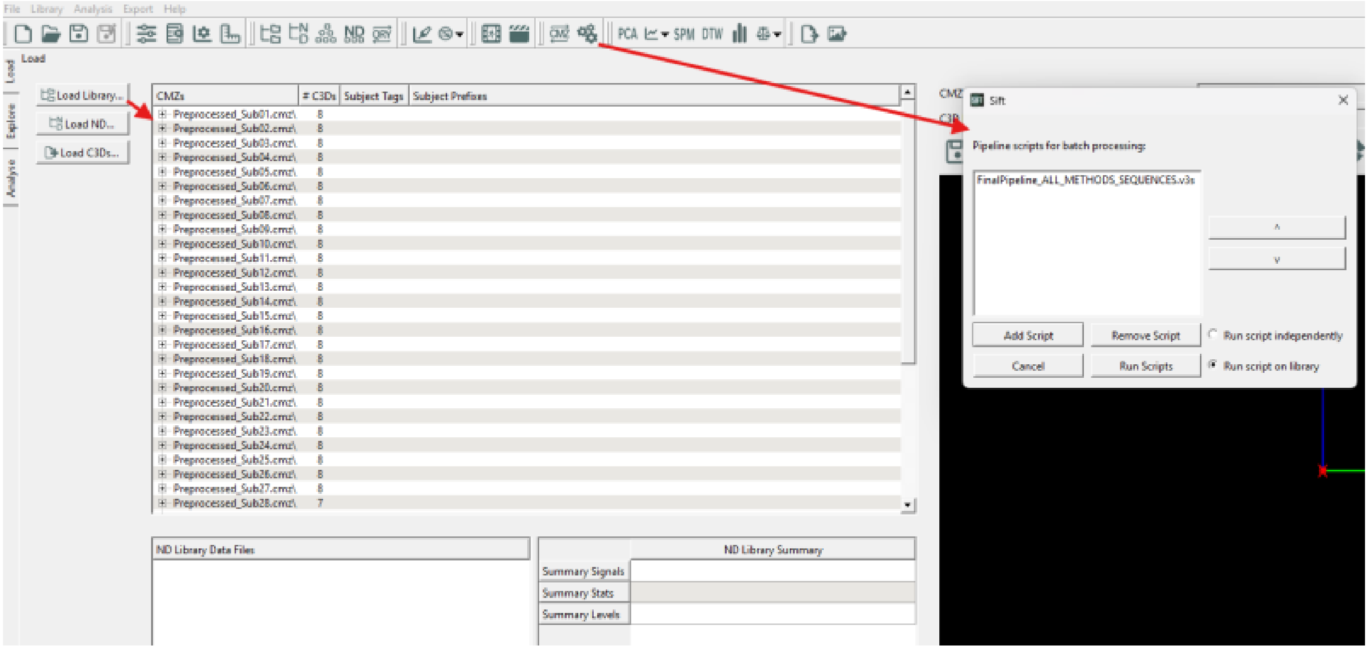
# Generating and Comparing Gait Cycles Across Methods

Once each individual pipeline was validated, a master script called **FinalPipeline_ALL_METHODS_SEQUENCES.v3s** was constructed. This script applies all methods in sequence to each participant's CMZ file and computes gait cycle durations for each method, the values for the durations will be stored under the METRIC::PROCESSED::FOLDER within each Visual3D workspace.

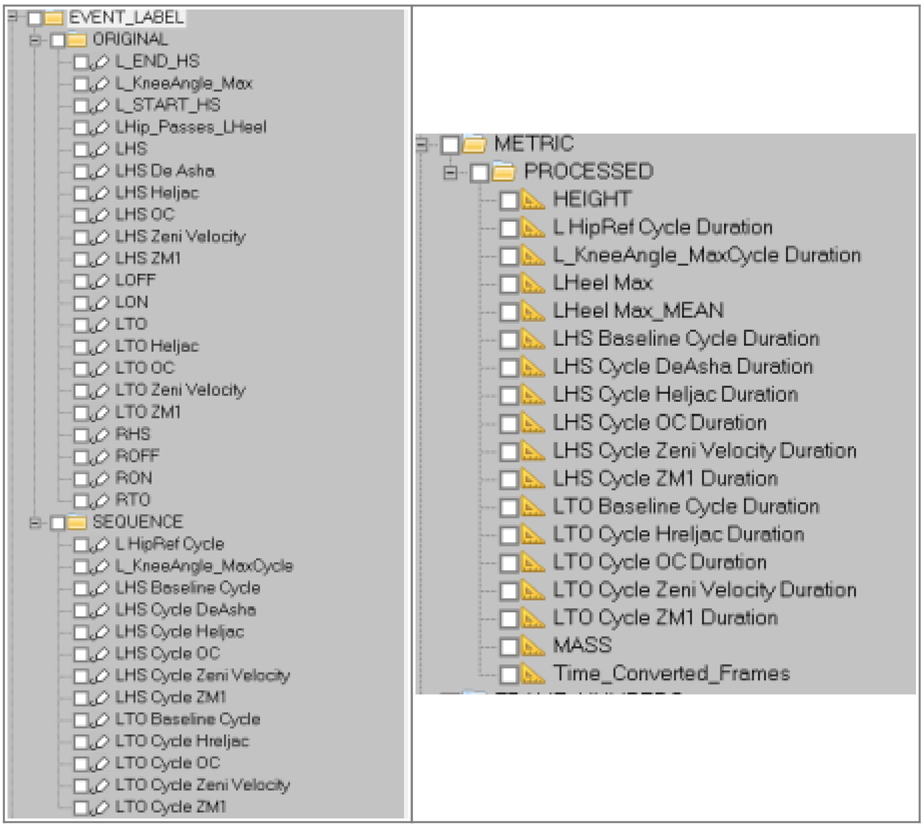
In order to apply this pipeline to all participant CMZs at once, **Sift** was used for it's batch processing capabilities while still allowing me to run a Visual3D pipeline script through the engine. The following steps were taken:

1. Open the Sift application to an empty workspace, select the **Load Library** and set **Library Path** to the folder where your .cmz files are located. Ensure you are using the files from the preprocessing step taken previously (or **PREPROCESSED_PARTICIPANTS** premade folder).

2. Select the **Run V3D Engine (Gear Icon)** button on the taskbar and select **Add Script**. Use the previously mentioned master pipeline here → click **Run Script**.




Once the pipeline has completed running on all the participant CMZs, you may open one on Visual3D to identify what was created.




# Exporting Data for Statistical Analysis

After processing, gait cycle duration data is exported using **Export_Data_To_ASCII**.

```
Export_Data_To_Ascii_File
! /FILE_NAME=
/SIGNAL_TYPES=
/SIGNAL_FOLDER=
```

```
! /SIGNAL_NAMES=
! /SIGNAL_COMPONENTS=
! /COMPONENT_SEQUENCE=
! /SIGNAL_PRECISION=
! /EVENT_SEQUENCE=
! /EXCLUDE_EVENTS=
! /USE_POINT_RATE=FALSE
! /NORMALIZE_DATA=FALSE
/NORMALIZE_POINTS=
! /EXPORT_MEAN_AND_STD_DEV=FALSE
! /EXPORT_MEAN_AND_STD_DEV_FOR_METRIC=FALSE
! /USE_P2D_FORMAT=FALSE
! /USE_XML_FORMAT=FALSE
! /USE_JSON_FORMAT=FALSE
! /USE_SHORT_FILENAME=FALSE
! /INCLUDE_TAGS_FOR_XML=FALSE
! /EXPORT_EMPTY_SIGNALS=FALSE
! /EXPORT_WITHOUT_HEADER=FALSE
! /EXPORT_COMPONENT_AS_TEXT=FALSE
! /EXPORT_NAN=FALSE
! /CREATE_FOLDER_PATH=TRUE
! /USE_SCIENTIFIC_NOTATION=FALSE
;
```

The command extracts the METRIC value of cycle durations for each method, for each of the participants. This is then opened in an excel file, and all participants were combined into one XML included in the sample data. The format of the sheet is the following:

| SUBJECT NO. | TRIAL | METHOD | CYCLE_INSTANCE | CYCLE_DURATION |
|---|---|---|---|---|
| Sub01 | walkT01 | Zeni Foot Position | 1 | 1.233 |

## Statistical Evaluation Using Linear Mixed Models

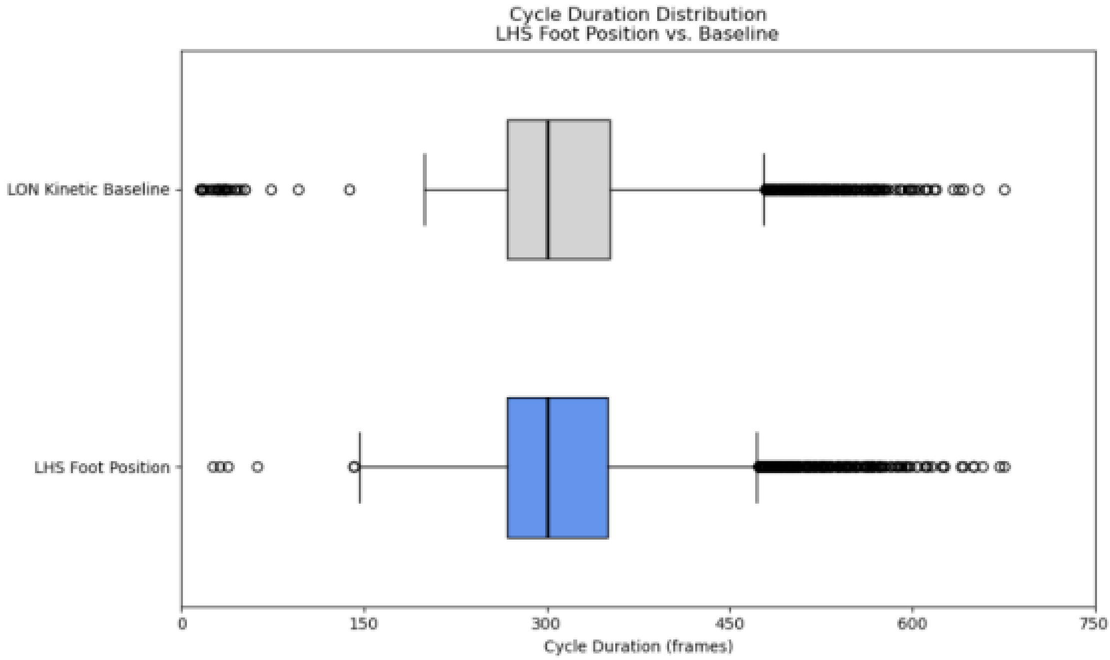
Once all gait events and cycle durations were processed in Visual3D, the data was ready for statistical analysis. To assess how well the different kinematic methods agreed with the kinetic gold standard, we used a statistical approach called a **Linear Mixed Model (LMM)** in Python (via the statsmodels library).

In this analysis, we looked at the **cycle duration** as the outcome we were measuring. We then compared this outcome across different **event detection methods** (which served as the fixed effect), while accounting for variation across participants and their trials (included as random effects).

This model was chosen because it is especially good at handling repeated measures, meaning it can recognize that the same person may walk differently across trials, and still lets us isolate whether the method itself is making a difference.

To visualize the results, we start with a boxplot that shows the cycle durations for one of the more reliable kinematic methods- **LHS Foot Position** (also known as Zeni Method 1)- compared to the **LON Kinetic Baseline**. You'll notice that the distributions of the two are quite similar, suggesting that this method might be reliably mimicking the gold standard.

However, this figure does not show us the whole story, since it doesn't take into account individual differences between participants and how they vary.



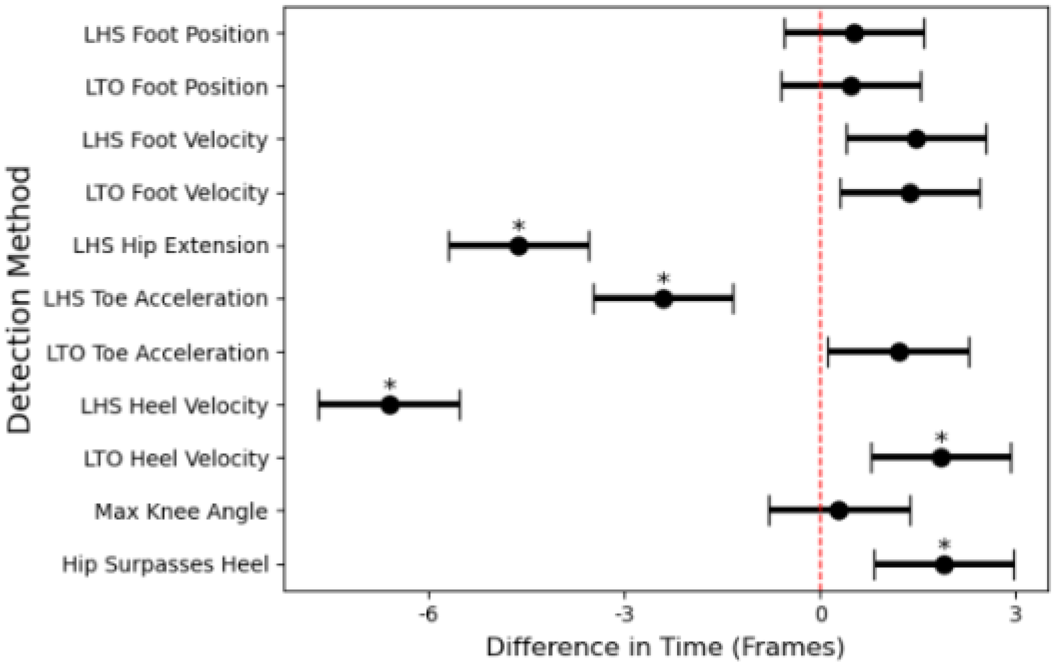

That's where the results plot from the Linear Mixed Model comes in. In this graph, we show how each method's average cycle duration compares to the baseline. Each point represents the difference in time between the method and the kinetic baseline, and the horizontal bars are confidence intervals.

If a method's bar overlaps with the red vertical line at zero, it means there's no **statistically significant difference**- that method produces similar cycle durations to the kinetic baseline. In this case, three methods stood out as being statistically similar to the kinetic reference:

1. LHS Foot Position
2. LTO Foot Position
3. Peak Knee Angle

These results suggest that those methods can structure gait cycles just as consistently as the gold standard kinetic approach.

On the other hand, several methods have an asterisk beside their points, meaning they were significantly different from the baseline using a Bonferroni correction. This indicates that while they may still detect events, they do not structure the gait cycle in a way that aligns closely with the kinetic reference.




## Conclusions

This tutorial demonstrates that while kinetic event detection remains the gold standard, kinematic methods—especially those based on relative foot position or knee angles—can offer reliable alternatives for structuring gait waveforms. These findings are especially relevant for field-based settings and markerless applications. By providing sample data, executable pipelines, and analysis scripts, this tutorial equips users with a practical workflow for comparing gait cycle structuring methods and generating reproducible results.

From:
https://has-motion.com/wiki/ - **Wiki Documentation Site**

Permanent link:
**https://has-motion.com/wiki/doku.php?id=visual3d:tutorials:events:kinematic_event_detection&rev=1748895939**

Last update: **2025/06/02 20:25**