



**Politecnico
di Torino**

Politecnico di Torino

Master's Degree in Biomedical Engineering

Graduation Session July 2026

Validation of Multi-Camera and Single-Camera Markerless Gait Analysis Approaches: OpenCap versus BRIDGE and the Effects of OpenSim Inverse Kinematics and Marker Augmentation

Supervisors:

Prof. Andrea Cereatti

Dr. Ing. Diletta Balta

Candidate:

Alessio Fusca

Academic Year 2025/2026

Abstract

Markerless (ML) gait analysis with multi or single camera approaches is increasingly investigated as an accessible alternative to marker-based (MB) motion capture, as it avoids the placement of physical markers on the subject’s skin and simplifies data acquisition. Current dual camera solutions include OpenCap, an open-source application from Stanford University, which reconstructs 3D body keypoints from two iPhones, predicts 3D virtual markers through marker augmentation (AGM) and estimates joint kinematics through OpenSim inverse kinematics (IK). Conversely, single-camera approaches based on statistical body-model fitting provide 3D joint centers from RGB/RGB-D data, but not the anatomical landmarks required for the Plug-in Gait (PiG) biomechanical protocol to estimate 3D joint kinematics. The objectives of this thesis were: (i) to validate OpenCap against a MB system and assess the added value of OpenSim IK over direct *PiG-like* computation from OpenCap AGM-derived virtual markers; (ii) to validate BRIDGE (Biomechanical Reconstruction through depth-Informed moDEL fittinG for gait Estimation), a novel subject-specific RGB-D fitting approach based on a SMPL statistical model; (iii) to investigate whether AGM improves the sagittal kinematics of BRIDGE and (iv) to evaluate its frontal kinematics performance.

Seven healthy participants wearing colored socks performed six self-selected-speed walking trials. Data were acquired simultaneously using a lateral RGB-D camera (Azure Kinect, 30 Hz), two iPhones positioned at 30° to the walking direction (60 Hz), and a 12-camera MB system (120 Hz). Three static views (frontal, sagittal, posterior) were acquired using Azure Kinect. Five pipelines were used: (i) MB PiG, as reference; (ii) BRIDGE, which reconstructs a subject-specific SMPL lower-limb model from three static RGB-D images, fits it to dynamic frames using an Articulated Iterative Closest Point algorithm, extracts hip, knee, and ankle joint centers, and computes sagittal joint angles using these centers together with color-based foot orientation; (iii) BRIDGE + AGM, in which BRIDGE joint centers were used to predict virtual anatomical markers through a pretrained AGM, enabling *PiG-like* estimation of sagittal and frontal lower-limb joint angles; (iv) OpenCap, based on standard OpenSim IK for sagittal kinematics along with hip ab/adduction; and (v) OpenCap *PiG-like*, in which OpenCap 3D virtual

markers were processed with the *PiG-like* workflow to compute sagittal and frontal lower-limb joint angles. Comparisons with the MB reference were performed after offset removal using root mean square error (RMSE).

OpenCap showed RMSE values below 5° for hip and knee flexion/extension and hip ab/adduction. OpenSim IK improved ankle flexion estimates (OpenCap RMSE $< 8^\circ$ vs 10° for OpenCap *PiG-like*). BRIDGE achieved good hip and knee angles accuracy (RMSE $\leq 5^\circ$) and the best ankle flexion results (RMSE $\leq 6^\circ$) compared to others, supporting the explicit color-based foot-orientation estimation. BRIDGE + AGM did not improve sagittal-plane kinematics compared with BRIDGE, probably because the pretrained AGM was developed to exploit bilateral lower-limb information; however, it enabled frontal-plane estimation, with hip ab/adduction and knee varus/valgus RMSE $< 4^\circ$. Ankle rotation remained challenging, with errors around 7° for both BRIDGE + AGM and OpenCap *PiG-like*. Overall, these results suggest that dedicated AGM retraining and the application of OpenSim IK to BRIDGE+AGM may further improve full 3D ML gait analysis.

Acknowledgements

I would like to express my sincere gratitude to Prof. Andrea Cereatti for giving me the opportunity to work on this thesis and for his guidance throughout the project.

My deepest thanks go to Dr. Diletta Balta for her constant availability and valuable advice. Her support was essential during the development of this work.

I would also like to thank all the members of the DET office for making this experience both pleasant and stimulating.

Lastly, I would like to thank my parents, my family and my friends. Their support and presence have accompanied me during this academic journey and have been fundamental in reaching this milestone.

Table of Contents

List of Acronyms	VI
1 Introduction	1
1.1 The importance of the human motion monitoring	1
1.2 Aim of the thesis	2
2 Background	4
2.1 Human motion analysis	4
2.1.1 Gait analysis	4
2.1.2 Kinematic analysis	6
2.1.3 Lower-limb kinematics	6
2.1.4 Segment reference frames and joint angles	6
2.2 Marker-Based motion capture	7
2.2.1 Marker types	7
2.2.2 Optical motion capture process	8
2.2.3 Biomechanical models and Plug-in Gait	9
2.2.4 Limitations	9
2.3 Markerless motion capture	10
2.4 Working principles of RGB-D technology	10
2.4.1 Depth image reconstruction	11
2.4.2 3D point cloud reconstruction from depth data	13
2.5 Markerless approaches for gait analysis	14
2.5.1 Deterministic approaches	14
2.5.2 AI approaches	18
2.5.3 Hybrid pipelines	20
3 Materials and methods	22
3.1 Study design and methodological overview	22
3.2 Participants	23
3.3 Experimental setup and data collection	23
3.3.1 Laboratory setup	23

3.3.2	Subject preparation	24
3.3.3	Data collection	25
3.4	Marker-based reference system	25
3.4.1	Vicon motion capture system and data processing	25
3.4.2	PiG lower body model and marker set	27
3.4.3	Segment reference frames	29
3.4.4	Joint angles calculation	33
3.5	OpenCap markerless pipeline	35
3.5.1	Overview of OpenCap	36
3.5.2	Two-dimensional pose estimation with HRNet	38
3.5.3	LSTM-based marker enhancement	39
3.5.4	OpenSim processing	40
3.6	BRIDGE and BRIDGE+AGM pipelines	41
3.6.1	SMPL body model	42
3.6.2	Subject-specific model calibration	48
3.6.3	AICP-based fitting procedure	49
3.6.4	Estimation of joint center and keypoint trajectories	52
3.6.5	Estimation of joint kinematics	54
3.6.6	LSTM-based marker augmentation	55
3.7	Data processing and temporal normalization	58
3.8	Performance evaluation and statistical analysis	62
4	Results	63
4.1	Original time-normalized curves	65
4.2	Curves after offset removal	71
5	Discussion	77
5.1	Validation of OpenCap and effect of OpenSim inverse kinematics . .	77
5.2	BRIDGE performance and the role of marker augmentation	78
5.3	Limitations	80
5.4	Future developments	81
	Bibliography	82

List of Acronyms

AE	Absolute Error
AGM	Marker Augmentation
AI	Artificial Intelligence
AICP	Articulated Iterative Closest Point
AJC	Ankle joint center
ASIS	Anterior Supine Iliac Spine
BRIDGE	Biomechanical Reconstruction through depth-Informed moDel fittinG for gait Estimation
DLC	DeepLabCut
HJC	Hip joint center
ICP	Iterative Closest Point
IK	Inverse Kinematics
IR	Infrared
KJC	Knee joint center
LBS	Linear Blend Skinning
LSTM	Long Short-Term Memory
MB	Marker-Based
ML	Markerless
PiG	Plug-in Gait
PSIS	Posterior Supine Iliac Spine
RGB	Red-Green-Blue
RGB-D	RGB-Depth
RMSE	Root Mean Square Error
SMPL	Skinned Multi-Person Linear
ToF	Time of Flight

Chapter 1

Introduction

1.1 The importance of the human motion monitoring

Human motion monitoring represents a fundamental area of research, with increasing relevance in clinical biomechanics, rehabilitation, and sports science. In particular, gait analysis focuses on the quantitative and objective assessment of human movement, providing essential information for clinical screening, early diagnosis of neurological disorders, and the optimization of therapeutic strategies.

Traditionally, instrumented gait analysis relies on marker-based (MB) optical stereophotogrammetry systems, which are currently considered the gold standard for motion capture [1]. This technique enables highly accurate tracking of markers placed on the subject's skin, achieving sub-millimeter spatial precision and high temporal resolution. Despite these advantages, stereophotogrammetric systems present several practical limitations, including long preparation times, difficulties in marker placement, potential physical and psychological discomfort for participants, as well as the need for highly specialized personnel and well-equipped laboratories. Furthermore, the high costs associated with these systems limit their widespread adoption, particularly outside controlled environments. As a result, they are not suitable for use in outpatient settings or for home-based assessments.

To overcome these limitations, alternative technologies have been developed over the years. Among these, markerless (ML) motion capture systems have gained increasing attention, as they enable the estimation of kinematic data without the need for physical markers. These systems offer several practical advantages, including reduced time and costs, greater portability, and ease of use, making them particularly suitable for applications such as large-scale screening and remote monitoring. However, despite their potential, the use of ML systems for quantitative gait analysis still requires thorough validation, particularly in terms of accuracy,

robustness, and reliability when compared with established MB approaches [2].

Recent advances in computer vision and video-based pose estimation have further accelerated the development of ML approaches [3]. In this context, the use of affordable and widely accessible devices has emerged as a promising direction.

Currently, motion analysis systems can be categorized into single-camera and multi-camera approaches, each characterized by different trade-offs in terms of accuracy, complexity, and ease of deployment. Within this framework, two main technological solutions can be identified.

On one hand, RGB-Depth (RGB-D) sensors, typically available at relatively low cost (approximately 200–400 €/€), combine a conventional Red-Green-Blue (RGB) camera with an infrared (IR) depth sensor, enabling the acquisition of both colour and depth information. By integrating these data, it is possible to obtain enriched 2D+ representations of the scene, allowing for accurate motion capture even with a single device and reducing the need for multi-camera setups.

On the other hand, purely video-based approaches rely on standard RGB cameras, including those embedded in smartphones, to estimate human motion without requiring dedicated hardware. These methods can operate either in single-view configurations, where 3D motion is inferred from 2D observations through learning-based models, or in multi-view setups, where multiple synchronized cameras are used to improve reconstruction accuracy. In this context, systems such as OpenCap leverage multi-view video data acquired from mobile devices to reconstruct human kinematics, improving accessibility and enabling motion analysis in non-laboratory environments.

1.2 Aim of the thesis

The objective of this thesis is the validation of ML approaches for lower-limb joint kinematic estimation through a systematic comparison with a MB reference system based on the Plug-in Gait (PiG) biomechanical protocol. Specifically, the study was designed as a validation analysis on healthy subjects, aimed at quantifying the accuracy of joint angle estimation obtained through alternative methodologies based on computer vision, statistical body models, and biomechanical modelling.

To this end, an experimental protocol was designed and conducted at the BiomedLab motion analysis laboratory of Politecnico di Torino. The analysis compares the reference MB PiG kinematics with four ML pipelines: a dual-camera approach based on OpenCap, a direct *PiG-like* computation from OpenCap-derived virtual markers, a single-camera RGB-D fitting approach referred to as BRIDGE, and a version of this pipeline based on marker augmentation (AGM), referred to as BRIDGE+AGM.

The OpenCap approach represents a dual-camera ML solution based on an

open-source application developed at Stanford University. It reconstructs 3D body keypoints from two iPhone videos, predicts 3D virtual anatomical markers through AGM, and estimates joint kinematics using OpenSim inverse kinematics (IK). In this thesis, OpenCap was evaluated both in its standard OpenSim-IK workflow and through a direct *PiG-like* computation based on the virtual markers provided by its AGM step, in order to assess the added value of OpenSim IK over direct *PiG-like* MB angle computation.

In parallel, the main contribution of this thesis lies in the validation of BRIDGE (Biomechanical Reconstruction through depth-Informed moDel fittinG for gait Estimation), a novel subject-specific RGB-D fitting approach based on a Skinned Multi-Person Linear (SMPL) statistical model. The proposed pipeline builds upon a previously developed framework for SMPL fitting and keypoint extraction, originally implemented by Balta in his doctoral work [4]. In BRIDGE, a subject-specific lower-limb SMPL model is reconstructed from three static RGB-D views and fitted to dynamic walking frames using an Articulated Iterative Closest Point (AICP) algorithm. Hip, knee, and ankle joint centers are then extracted, and sagittal joint angles are computed using these centers together with color-based foot orientation.

A further objective of this thesis is to investigate whether AGM can improve the sagittal kinematic estimation of BRIDGE and extend its applicability to frontal-plane joint angles. For this purpose, BRIDGE joint centers were used as input to an adapted version of a pre-trained AGM model to predict virtual anatomical markers. The resulting BRIDGE+AGM pipeline enables a *PiG-like* estimation of sagittal and frontal lower-limb joint angles, allowing the assessment of hip ab/adduction, knee varus/valgus, and ankle rotation in addition to sagittal-plane kinematics.

Overall, this thesis aims to assess the accuracy and completeness of accessible ML approaches for gait analysis, with particular attention to the contribution of AGM and biomechanical IK in improving lower-limb joint angle estimation.

Chapter 2

Background

2.1 Human motion analysis

Human motion analysis is the quantitative study of human movement, aimed at describing, evaluating, and interpreting motor functions in both healthy and pathological subjects. It plays a crucial role in clinical, sports, and research contexts, enabling objective assessment of movement patterns, diagnosis of musculoskeletal and neurological disorders, and monitoring of rehabilitation outcomes. In clinical settings, motion analysis is widely used to support decision-making processes, such as treatment planning and evaluation of surgical or rehabilitative interventions. The ability to obtain objective and repeatable measurements of movement is essential for improving patient-specific care and tracking functional recovery over time.

2.1.1 Gait analysis

Gait analysis is a specific branch of human motion analysis that integrates tools and methodologies for the measurement and estimation of variables associated with walking. These variables can be broadly categorized into kinematic and dynamic quantities, providing a comprehensive description of locomotor function.

Gait analysis primarily focuses on steady-state walking conditions and is commonly based on the assumption of quasi-periodicity. Under this assumption, walking can be effectively characterized through the statistical description of a single gait cycle, typically in terms of mean values and variability across cycles.

According to Perry [5], the gait cycle is defined as the time interval between two successive initial contacts of the same foot with the ground, and it is considered the fundamental unit for the analysis of human walking. For clarity, it is useful to distinguish between a *stride* and a *step*: a stride corresponds to a full gait cycle and refers to the motion of a single limb, while a step represents the interval between the initial contact of one foot and the initial contact of the contralateral foot.

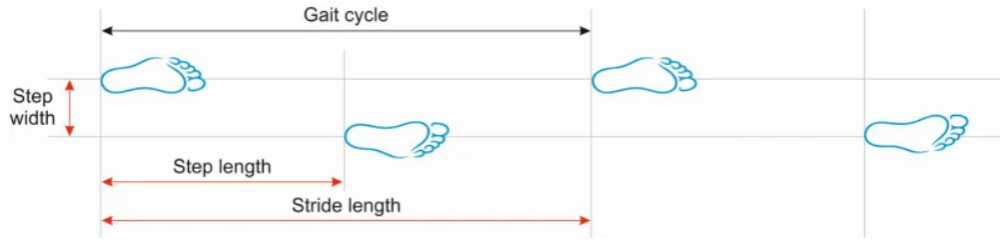


Figure 2.1: Difference between stride and step. Reproduced from [6].

Within this framework, the gait cycle is functionally divided into two main phases: the stance phase and the swing phase, typically described with reference to a single lower limb. The stance phase corresponds to the period during which the foot is in contact with the ground, while the swing phase refers to the period during which the foot is off the ground and advances forward in preparation for the next contact.

For a more detailed description, each phase can be further subdivided into specific sub-phases. In particular, the stance phase includes initial contact (heel strike), loading response, mid-stance, terminal stance, and pre-swing (toe-off), while the swing phase is divided into initial swing, mid-swing, and terminal swing. These subdivisions provide a more precise characterization of the biomechanical and functional events occurring throughout the gait cycle.

When considering both lower limbs simultaneously, the gait cycle can also be described in terms of support phases. In this context, two periods of double support can be identified, during which both feet are in contact with the ground, alternating with periods of single support, during which only one limb supports the body while the contralateral limb is in the swing phase.

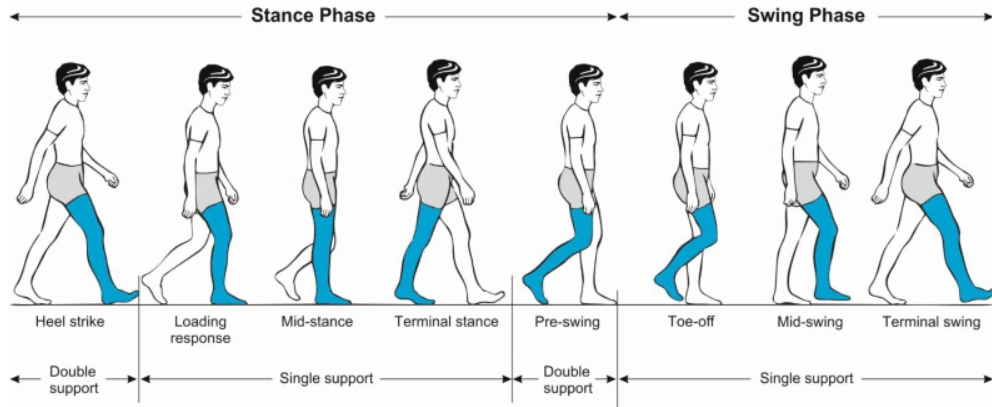


Figure 2.2: Phases of the normal gait cycle. Reproduced from [6].

2.1.2 Kinematic analysis

Kinematic analysis describes human movement in terms of its geometric properties, without considering the forces that generate it. In gait analysis, the human body is commonly modeled as a system of rigid segments connected by joints, and motion is described through positions, orientations, and their variation over time. Joint kinematics is typically expressed through joint angles, which provide a compact representation of the relative motion between adjacent segments [7]. During walking, these angles are evaluated over the gait cycle and represented as continuous trajectories, usually normalized as a percentage of the gait cycle to enable comparison across subjects and trials [8]. Such representations allow the identification of characteristic movement patterns and facilitate the analysis of inter-subject variability.

Overall, kinematic analysis provides a fundamental tool for the quantitative assessment of human movement, supporting both clinical evaluation and research applications.

2.1.3 Lower-limb kinematics

Lower-limb kinematics focuses on the motion of the main body segments involved in human locomotion, namely the foot, shank, thigh, and pelvis. The relative motion between body segments defines the kinematics of the ankle, knee and hip joints, which are essential for the generation and control of walking [7].

Joint motion is commonly described with respect to anatomical planes: the sagittal plane includes flexion and extension, which represent the main movements during walking; the frontal plane describes abduction and adduction, relevant for balance and lateral stability; and the transverse plane accounts for internal and external rotations. In gait analysis, hip, knee, and ankle joint angles are commonly used to characterize lower-limb motion and to investigate differences in gait performance across subjects or populations [8]. In particular, joint angles at the knee and ankle have been shown to provide valuable insights into mobility and frailty in older adults [9].

2.1.4 Segment reference frames and joint angles

The description of joint kinematics relies on the definition of local reference frames associated with each body segment. These reference frames provide a consistent way to represent the position and orientation of segments in three-dimensional space and form the basis for the computation of joint motion [7].

In this context, the human body is commonly modeled as a system of rigid segments, each associated with a local coordinate system. Joint kinematics is defined

from the relative motion between adjacent segments and is typically expressed through joint angles describing their relative orientation [7, 8].

Since joint angles depend directly on the definition of segment reference frames, the identification of anatomical landmarks represents a critical step in motion analysis. Differences in landmark placement or coordinate system definition may lead to variations in the computed kinematics, reducing consistency and reproducibility across measurements.

For this reason, standardized biomechanical models have been developed to define common conventions for segment reference frames and joint angle computation. Among them, the PiG model is widely used in clinical gait analysis, providing a reference framework that supports comparability across studies and measurement systems.

2.2 Marker-Based motion capture

Optoelectronic stereophotogrammetric multi-camera systems are considered the gold standard for motion analysis, as they enable the tracking of markers placed on the subject's body [3, 10]. This approach allows for the reconstruction of the three-dimensional position of each marker over time, providing a detailed description of movement. Based on this information, it is possible to perform a kinematic analysis of the anatomical segments to which the markers are attached, provided that they are carefully positioned to ensure their visibility to the camera system.

2.2.1 Marker types

Markers can be classified as either passive or active. Passive markers, widely used in commercial motion capture systems, are typically spherical elements coated with retro-reflective materials, capable of reflecting the IR radiation emitted by the cameras. Their position is reconstructed from the reflected light, making this approach simple and minimally invasive. However, it presents several limitations, particularly in the marker identification phase during thresholding: markers are not always uniformly illuminated, which can make trajectory tracking challenging. As a result, it is often necessary to rely on the spatial configuration of the markers and temporal coherence to resolve ambiguities. Additional issues include occlusions and spurious reflections.

Active markers consist of light-emitting diodes (LEDs), typically operating in the IR spectrum, which directly emit a detectable signal and do not require external illumination. Although this enables their use in low-light conditions, occlusions and trajectory-tracking issues may still occur. In addition, active marker systems require electrical power, often leading to the presence of cables.

A further solution is represented by time-modulated active markers, in which LEDs are activated according to a temporal sequence synchronized with the camera frame rate, so that only one marker is visible in each frame. This approach removes ambiguities in marker identification, but increases system complexity because it requires synchronization between cameras and markers and dedicated activation hardware. Moreover, the effective acquisition frequency of each marker is reduced, as it corresponds to the camera frame rate divided by the number of markers used.

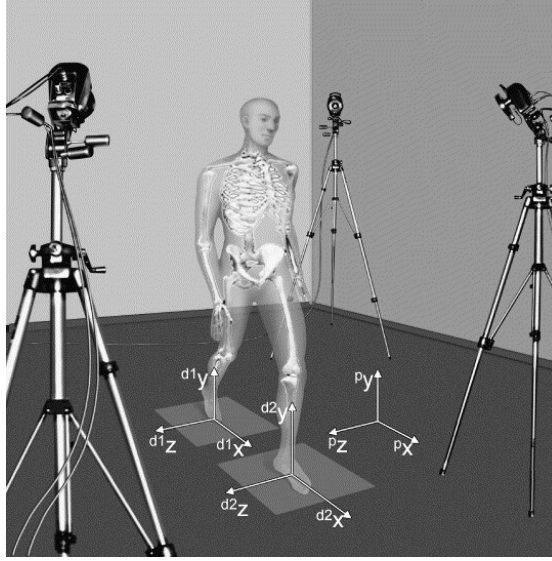


Figure 2.3: Example of optoelectronic stereophotogrammetric multi-camera system. Reproduced from [11].

2.2.2 Optical motion capture process

The optical motion capture process consists of several sequential steps. First, a calibration procedure is performed, during which a global reference system is defined, and both intrinsic (focal length and lens distortion) and extrinsic (position and orientation) camera parameters are estimated. This is followed by the image acquisition and marker detection phase, in which markers are identified in each camera view, typically through thresholding and labeling procedures, allowing the extraction of their 2D positions in the image plane. Subsequently, a 3D reconstruction step is performed using triangulation to combine information from multiple cameras. In order to accurately reconstruct the 3D position of a marker, it must be visible from at least two distinct cameras, although in practice a higher number of cameras (typically 8–12) is used to improve robustness. This step relies on the camera parameters obtained during the calibration phase. Once the 3D

marker coordinates have been reconstructed, they represent the spatial trajectories of the markers over time. These trajectories constitute the input for subsequent processing steps, in which biomechanical models are applied to estimate segment motion and joint kinematics.

2.2.3 Biomechanical models and Plug-in Gait

The computation of joint kinematics from reconstructed marker trajectories requires the adoption of biomechanical models to define body segments and their relative motion. In MB motion capture systems, the human body is typically represented as a set of rigid segments connected by joints. These models define key aspects of the analysis, including the identification of anatomical landmarks for marker placement, the construction of segment reference frames, and the computation of joint angles describing the relative motion between adjacent segments.

Among the available models, the PiG model is one of the most widely used in clinical gait analysis and is commonly implemented in commercial stereophotogrammetric systems. It specifies a predefined marker set and a standardized procedure to estimate segment kinematics and joint angles based on anatomical landmark definitions [12].

The adoption of standardized models such as PiG ensures consistency and repeatability in the analysis, enabling comparisons across subjects and experimental sessions. However, the accuracy of the estimated kinematics is influenced by factors such as marker placement, soft tissue artifacts, and the assumptions underlying the biomechanical model.

As a result, the choice of the biomechanical model has a direct impact on the resulting kinematic measurements. Different modelling approaches may lead to variations in the estimated joint angles, making model selection a critical aspect in the interpretation and comparison of motion capture data.

2.2.4 Limitations

Optoelectronic stereophotogrammetric systems present several inherent limitations. One of the main issues is related to marker occlusions, which occur when markers are temporarily hidden from the cameras by body segments or external objects, compromising trajectory continuity. Another significant limitation is the soft tissue artifact, caused by the relative motion between the skin and the underlying bone, which can introduce substantial errors in the estimation of anatomical segment motion [13]. Additional challenges concern the correct identification and labeling of markers, which may become difficult in the presence of similar trajectories or rapid movements, particularly in passive marker systems. Furthermore, the accuracy of 3D reconstruction strongly depends on the camera calibration and configuration:

errors in intrinsic or extrinsic parameters can propagate to the reconstructed marker positions [14]. From a practical perspective, these systems are characterized by high costs and are typically available only in specialized laboratory environments. They also operate within limited acquisition volumes, which can restrict the range of analyzable movements. Moreover, the experimental setup requires time-consuming subject preparation, including careful marker placement and system calibration, which may limit its applicability in clinical or real-world scenarios.

2.3 Markerless motion capture

ML motion capture refers to techniques used to estimate human motion without physical markers attached to the subject’s body. Unlike MB systems, which rely on markers placed on anatomical landmarks, ML approaches use visual data acquired from cameras to automatically detect and track body motion.

In a typical ML pipeline, images are acquired from one or more cameras and processed using computer vision algorithms, often based on machine learning [15] or deep learning techniques [16]. These algorithms identify anatomical keypoints, such as joints, which are then used to reconstruct the human pose in two or three dimensions, depending on the system configuration.

ML systems can be classified according to the acquisition setup and sensor type. Monocular approaches rely on a single camera and infer three-dimensional pose through learning-based or model-based methods, whereas multi-camera systems use synchronized views to reconstruct three-dimensional keypoints through triangulation. Similarly, RGB-based methods require depth information to be inferred or reconstructed, while RGB-D sensors directly provide per-pixel depth measurements, making them suitable for single-camera three-dimensional motion capture.

These systems reduce preparation time, improve subject comfort, and allow motion analysis outside laboratory environments. However, their performance may still be affected by accuracy limitations, occlusions, environmental conditions, and the robustness of the underlying algorithms.

2.4 Working principles of RGB-D technology

The introduction of RGB-D cameras represented an important advancement in motion analysis and computer vision, as these sensors allow color (RGB) and depth information to be acquired simultaneously. For this reason, they are widely used in ML motion capture, where depth data provide additional spatial information that supports three-dimensional reconstruction.

RGB-D devices became widely accessible with the release of the Microsoft Kinect in 2010, which was originally developed for gaming applications and relied on structured light for depth estimation. Later sensors introduced different depth-sensing principles, such as time-of-flight (ToF) technology in the second-generation Kinect and stereoscopic vision in devices such as the Intel RealSense series. More recent systems, including the Azure Kinect used in this work, provide improved performance and integration capabilities, supporting their application in fields such as biomechanics [17].

2.4.1 Depth image reconstruction

A depth image represents the distance between the camera and the observed scene, typically with a grayscale map in which each pixel corresponds to a depth value. Depth images can be reconstructed using different techniques, including fixed structured light, stereoscopic vision, and time-of-flight (ToF) approaches.

Fixed structured light

The Kinect v1 introduced basic gesture recognition capabilities by leveraging structured light technology for depth sensing. This approach consists of projecting a known infrared pattern (a grid or a set of stripes) onto the scene using the IR emitter of the RGB-D camera. When the pattern interacts with the surfaces of the observed objects, it becomes distorted. An IR depth sensor captures the deformed pattern, which is then compared with the original one. Based on this comparison, the system estimates the depth information through a triangulation process, allowing the computation of the distance between the camera and the objects in the scene [18].

Stereoscopic vision

Stereoscopic vision represents one of the approaches used in RGB-D sensors for depth estimation, based on principles analogous to human binocular vision. In this case, the system consists of two RGB cameras positioned at a known distance from each other (baseline), which simultaneously acquire images of the same scene from slightly different viewpoints. This approach has been widely adopted in devices such as Intel RealSense cameras Figure 2.5. Depth estimation first requires a calibration phase, during which the intrinsic parameters (such as focal length, principal point coordinates, and optical distortion) and extrinsic parameters (translation vector and rotation matrix) of the cameras are determined. Subsequently, the acquired images undergo a rectification process, which transforms them into standard stereo images, in which homologous points are aligned along the same image rows in both the left and right views [19]. This is achieved by exploiting the calibration

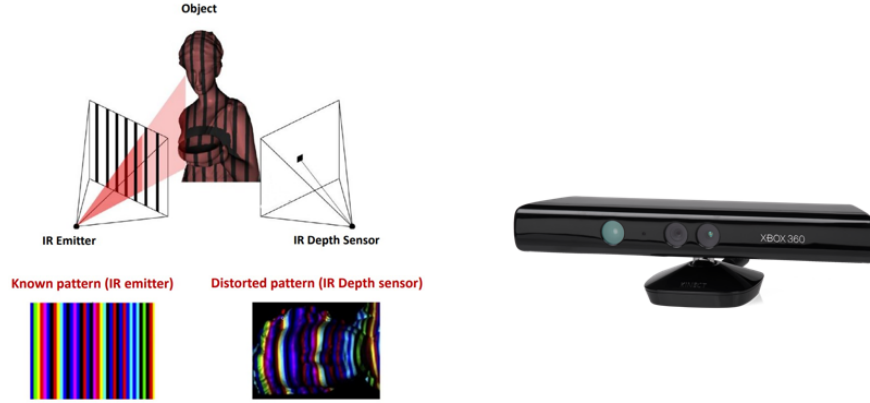


Figure 2.4: Structured light principle. Left: Known IR pattern projected onto the scene (left) and its deformation captured by the IR sensor (right), used for depth estimation through triangulation. Right: Kinect v1.

parameters, which allow compensation for the geometric differences between the two views. Depth is then reconstructed through a correspondence process between homologous points in the images called stereo matching: the difference in position of the same point in the two views, defined as disparity, is inversely proportional to the distance of the object from the camera. Starting from the disparity map and knowing the geometric parameters of the system, it is possible to obtain the depth map of the scene through triangulation. Although this approach does not require active signal emission, and is therefore considered passive, it can be sensitive to lighting conditions and to the presence of poorly textured surfaces, which make the identification of correspondences between images more challenging.

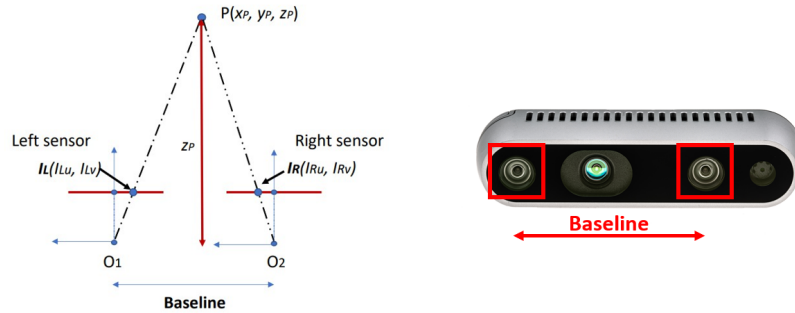


Figure 2.5: Stereoscopic vision. Left: Schematic representation of the stereoscopic vision setup, illustrating two cameras separated by a baseline and the projection of a 3D point onto their respective image planes, from which depth can be estimated through triangulation. Right: Intel RealSense D435.

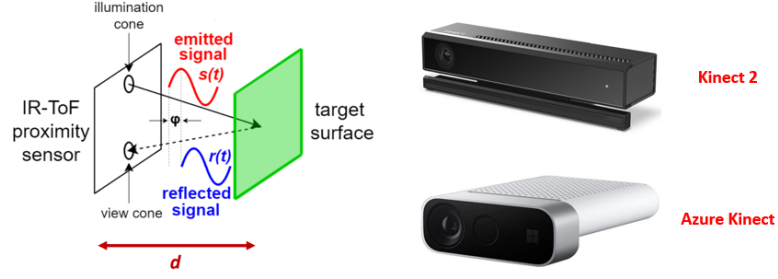


Figure 2.6: Left: Principle of ToF depth sensing. Right: Examples of RGB-D devices adopting this technology: Kinect 2 and Azure Kinect.

Time of Flight (ToF)

RGB-D cameras based on ToF technology estimate depth by exploiting the propagation of IR light. Specifically, the sensor emits an IR signal and captures its reflection from objects in the scene; the distance is computed by measuring the phase shift between the emitted and received signals, which is proportional to the light's round-trip time. This approach enables real-time operation, making it particularly effective for applications in computer vision and human body tracking. Compared to structured light approaches, ToF systems are less sensitive to ambient illumination and do not require complex processing to decode projected patterns, resulting in faster and more robust measurements [18]. They also provide an extended operating range and more consistent accuracy across varying distances and environmental conditions. This evolution is evident when comparing Kinect 2 and Azure Kinect. While both devices rely on ToF technology, Azure Kinect represents a further advancement, offering higher depth resolution as well as improved spatial and temporal accuracy. It also introduces more flexible acquisition modes, with different fields of view and operating ranges, and provides enhanced synchronization with external devices. Moreover, it leverages more advanced reconstruction and body tracking algorithms, making it better suited for research applications [20].

2.4.2 3D point cloud reconstruction from depth data

RGB-D cameras enable the reconstruction of 3D point clouds representing the objects within the camera's field of view. A point cloud is a collection of points in a three-dimensional coordinate system, where each point describes the spatial position of a surface element. The reconstruction process begins with camera calibration, which provides the intrinsic parameters of the sensor, including the focal lengths and the coordinates of the principal point. Each pixel (p_x, p_y) , associated with a depth value $D(p_x, p_y)$, can be projected into 3D space. This mapping allows the

computation of the corresponding 3D coordinates (X, Y, Z) , where Z represents the measured depth, while X and Y are obtained by scaling the pixel coordinates with respect to the focal lengths f_x and f_y and translating them relative to the principal point (pp_x, pp_y) . By applying this transformation to all pixels, a dense set of 3D points is generated, forming the point cloud.

$$X = \frac{p_x - pp_x}{f_x} \cdot D(p_x, p_y)$$

$$Y = \frac{p_y - pp_y}{f_y} \cdot D(p_x, p_y)$$

$$Z = D(p_x, p_y)$$

2.5 Markerless approaches for gait analysis

After describing the main principles underlying ML motion capture and RGB-D technology, it is useful to discuss how visual and depth data can be processed to perform gait analysis. Several ML approaches have been proposed for this purpose, differing in terms of input data, assumptions about the human body, and algorithms used for pose estimation and motion reconstruction. In general, these methods can be grouped into three main categories: deterministic approaches, AI-based approaches, and hybrid pipelines, which combine elements of both.

2.5.1 Deterministic approaches

Deterministic approaches rely on predefined mathematical relationships, anatomical proportions, and geometric principles to estimate the position of the joint centers and the orientations of the segments. These methods are widely used because, for a given set of input data and assumptions, they provide reproducible results that are directly related to predefined anatomical and mathematical relationships. A further advantage of deterministic approaches is that they do not require large training datasets. This represents an important benefit compared with AI-based methods, such as machine learning and deep learning approaches, which generally achieve better performance when trained on large and representative datasets. In clinical applications, collecting large amounts of data can be particularly challenging, especially when pathological populations are involved. Therefore, deterministic methods remain valuable in contexts where data availability is limited and interpretability is important. Within human motion analysis, these approaches can be broadly divided into two main categories: model-based and model-free methods [21].

Model-based approaches rely on a predefined representation of the human body, whereas model-free methods extract motion-related features directly from image data, for example from the human silhouette, without explicitly fitting a complete biomechanical model. In ML gait analysis, model-based methods use an a priori anatomical or biomechanical model to guide the reconstruction of human movement from images, videos, or depth data. The body is typically represented as a set of interconnected segments and joints, whose geometry and motion are constrained by anatomical relationships. The model may take different forms, such as a stick figure, a volumetric representation, or a surface-based model, depending on the level of detail required. The core principle of these approaches is the fitting of the predefined model to the observed data. During this process, the position, orientation, and in some cases the shape of the model are iteratively adjusted in order to match the visual or depth information acquired by the cameras. The fitting procedure is usually formulated as an optimization problem, in which the difference between the model prediction and the observed data is minimized. Once the model has been aligned with the recorded motion, biomechanical variables such as joint centers, segment orientations, joint angles, velocities, and accelerations can be estimated from the reconstructed movement.

On the other hand, model-free approaches rely on visual features extracted directly from images, most commonly the human silhouette. After the silhouette has been segmented from the background, its shape and spatial distribution are analysed to infer the approximate position of body segments and joint centers. Anatomical knowledge is still used, but in a less explicit way than in model-based approaches: rather than adapting a complete articulated model to the observed motion, model-free methods exploit general human body proportions to interpret the silhouette. For example, the relative lengths of adjacent body segments can be used to estimate joint positions. Therefore, the reconstructed movement is derived directly from image-based features, supported by simplified anatomical assumptions. Compared with model-based methods, model-free approaches are generally simpler and less computationally demanding, since they do not require the optimization of a complete 2D or 3D body model. However, their accuracy can be affected by the quality of silhouette extraction, body orientation with respect to the camera, self-occlusions, and variability in individual anthropometry. For this reason, they can provide a useful approximation of body kinematics, but they are generally less suitable when a detailed biomechanical reconstruction is required. In such cases, model-based approaches are usually preferable.

Among model-based strategies, early multi-view systems exploited the visual hull technique to obtain a volumetric approximation of the subject from multiple synchronized RGB cameras. In a typical visual-hull pipeline, the subject is acquired

simultaneously from different viewpoints, and a binary silhouette is extracted from each camera view after background segmentation. Each silhouette defines a cone-shaped volume in 3D space, corresponding to all possible points that would project inside that silhouette. The intersection of the visual cones obtained from all views generates the visual hull, an approximate 3D volume enclosing the subject. This reconstructed volume can then be used for motion tracking by fitting an articulated anatomical model to the visual hull sequence. During tracking, the model pose is iteratively updated to minimize the discrepancy between the model representation and the reconstructed volume. In this context, articulated iterative closest point (AICP) registration is particularly suitable because it adapts the standard iterative closest point (ICP) algorithm, which aligns two sets of points by minimizing the distance between corresponding points [22], to articulated structures such as the human body. Unlike standard ICP, AICP accounts for multiple interconnected segments and enforces joint constraints, providing an alignment that is more consistent with biomechanical movement patterns [23]. This procedure allows the estimation of segment poses over time, from which joint center trajectories and joint kinematics can subsequently be derived. For example, Mündermann and colleagues proposed a multi-camera ML motion capture method based on visual hull reconstruction, subject-specific model generation, and AICP registration, enabling the tracking of body segment motion without physical markers [21]. Similar multi-camera ML technologies were later compared with MB systems during gait analysis [24].

In parallel, model-free methods were also developed to estimate gait-related variables directly from image features, without explicitly fitting a complete biomechanical model. Goffredo et al. proposed a completely ML 2D gait analysis system based on video sequences acquired with a single RGB camera, to extract kinematic information in a low-cost and non-invasive way [25]. Leu et al. later proposed a two-camera RGB system, in which the subject silhouette was segmented in each view and used to estimate joint center positions [26]. Although these methods are relatively simple and computationally efficient, their dependence on average anthropometric proportions may limit their applicability in pathological populations, where asymmetries, deformities, or altered movement patterns can make joint center localization less reliable.

Other deterministic approaches adopted simplified 2D model-based formulations. Surer et al. proposed a ML method to estimate the sagittal kinematics of the shank-foot complex during the stance phase of walking using a single RGB camera. Their approach was based on a multi-rigid-body representation of the shank, rearfoot, and forefoot, with the model aligned to image data through cross-correlation [27]. Although promising, this method was restricted to the ankle-foot complex

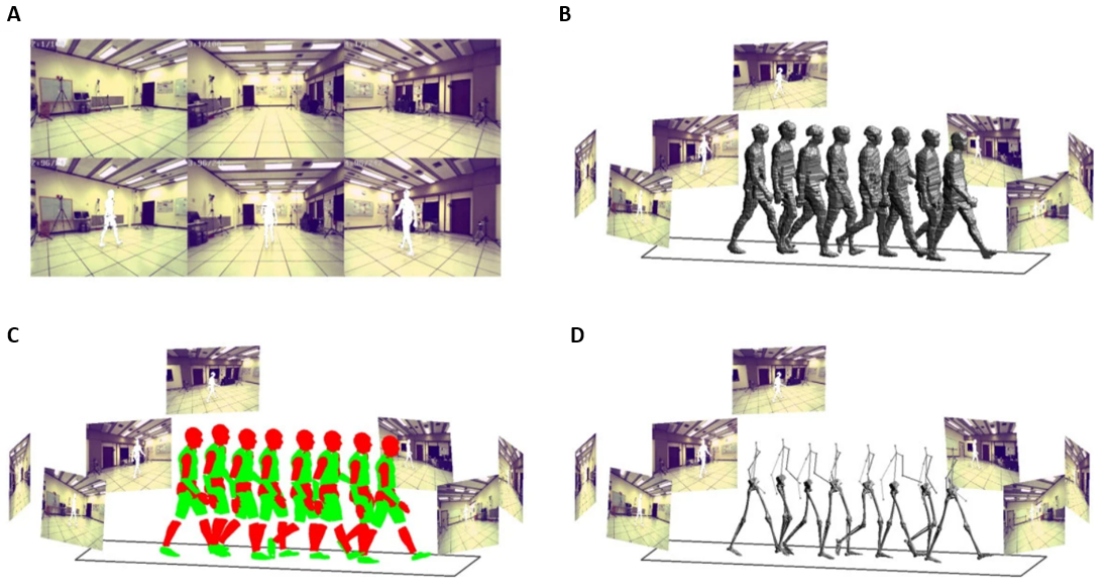


Figure 2.7: (A) Multiple synchronized camera views are acquired and the subject is segmented from the background in each image. (B) The extracted silhouettes are projected into 3D space and combined to reconstruct a visual hull representing the subject's approximate body volume. (C) A subject-specific articulated model is generated and fitted to the reconstructed volume. (D) The fitted model is tracked over time to estimate body segment poses and derive the corresponding skeletal motion. Adapted from [21].

and to the stance phase, making its extension to the entire gait cycle challenging because of segment deformation, occlusions, and foot alignment during swing. More recent studies moved towards single-camera model-based protocols for gait analysis. Castelli et al. proposed a 2D ML methodology based on a single RGB camera and a multi-segment lower-limb model to estimate sagittal-plane kinematics during gait [28]; however, this method was validated only on healthy subjects. Pantzar-Castilla et al. later extended this line of research by proposing a model-based approach using a single RGB-D camera in children with cerebral palsy [29]. Although depth information was used to support motion reconstruction, the need for a green background to facilitate subject segmentation limited the portability of the system. In addition, validation against a MB reference system was performed only for knee joint kinematics, leaving the accuracy of a complete lower-limb analysis only partially assessed.

To improve clinical applicability, Balta et al. proposed a 2D model-based ML protocol for clinical gait analysis using a single RGB-D camera in 18 patients with cerebral palsy [30]. The protocol relied on a 2D lower-limb model, whose joint center

positions were estimated by registering the model to dynamic 2D images through an ICP procedure. The method was validated against a conventional 3D MB gait analysis protocol, assessing both accuracy and reliability for spatio-temporal parameters and sagittal-plane lower-limb joint angles. Despite these advances, residual errors may still occur in 2D ML protocols, mainly because two-dimensional modelling cannot fully account for out-of-plane movements or changes in the relative position between the subject and the camera during gait. Moreover, the manual identification of anatomical landmarks during model calibration may introduce further variability. These limitations have motivated the development of extended 3D approaches based on statistical body models, aimed at estimating three-dimensional lower-limb joint center trajectories and improving the robustness of sagittal-plane kinematics in individuals with cerebral palsy and foot deformities [4].

2.5.2 AI approaches

AI-based approaches have become increasingly relevant in ML gait analysis, mainly due to the development of deep learning methods for human pose estimation. Although earlier machine learning techniques have also been explored, most recent approaches rely on convolutional neural networks (CNNs), which can learn complex spatial features directly from images or video frames. Unlike deterministic methods, which are mainly based on predefined geometric and anatomical assumptions, deep learning-based approaches learn the relationship between visual input and body pose from large annotated datasets. Depending on the specific method, their output may consist of two-dimensional image keypoints, three-dimensional skeletal landmarks, or parameters used by subsequent reconstruction algorithms, such as pose and shape parameters for statistical body models. Usually, these methods can be considered model-free, since they detect anatomical landmarks directly from images without explicitly fitting a complete body model. In contrast, RGB-D skeletal tracking systems, such as the Azure Kinect body tracking SDK, rely on a predefined skeletal representation and can therefore be considered model-based tracking approaches. Several deep learning-based pose estimation frameworks have been proposed for human movement analysis, including OpenPose, DeepLabCut (DLC), AlphaPose, MediaPipe Pose/BlazePose and MoveNet. However, the following sections focus on selected representative methods that are particularly relevant for ML gait analysis, namely OpenPose, DLC, and Azure Kinect body tracking SDK.

OpenPose

OpenPose is one of the most widely used deep learning-based frameworks for ML human pose estimation. It was originally developed for real-time multi-person 2D



Figure 2.8: Typical output of OpenPose for multi-person 2D pose estimation. Adapted from [16].

pose estimation from RGB images and video sequences [16]. The method follows a bottom-up strategy: candidate body keypoints are first detected in the image, and then associated with individual subjects through Part Affinity Fields, which encode the spatial relationships between adjacent body parts. This architecture allows OpenPose to estimate the pose of multiple people in the same frame while maintaining real-time performance.

In gait analysis, OpenPose has been widely adopted because it only requires conventional video recordings and can therefore support low-cost and non-invasive motion assessment. It has been used, for example, in workflows based on videos acquired with a mobile phone camera, where 2D landmarks are extracted to calculate knee flexion–extension [31]. Its performance has also been evaluated against three-dimensional MB motion capture for the estimation of spatio-temporal and sagittal-plane kinematic gait parameters, showing its potential for accessible video-based gait analysis while also highlighting limitations related to two-dimensional measurements and camera viewpoint [32].

DeepLabCut

DLC is a deep learning-based framework originally developed for ML pose estimation in animals, and later adopted in a wide range of movement analysis applications, including human motion analysis [33]. Unlike general-purpose human pose estimators, it allows the user to define the body points to be tracked, rather than relying on a fixed set of predefined anatomical keypoints. The method is based on transfer learning: a network pre-trained on large image datasets is fine-tuned using a relatively small number of manually labeled frames from the

specific experimental setup. This makes DLC particularly flexible, since it can be adapted to different subjects, viewpoints, body landmarks, and movement tasks. In gait analysis, this flexibility is useful when the standard keypoint set provided by general human pose estimation algorithms is not sufficient or when specific anatomical landmarks need to be tracked. Recent work has specifically investigated the use of DLC for gait analysis by comparing a pre-trained OpenPose model, a pre-trained DLC model, and a custom-trained DLC model. In this context, the custom-trained DLC configuration, especially when combined with the refinement function, showed improved performance for locomotion assessment, highlighting the importance of task-specific training and refinement when applying ML pose estimation to gait analysis [34]. However, the accuracy of DLC strongly depends on the quality and consistency of manual annotations, the representativeness of the training dataset, and the similarity between training and testing conditions.

Azure Kinect body tracking SDK

The Azure Kinect Body Tracking SDK is an AI-based ML tracking system that exploits RGB-D data to estimate the three-dimensional position of body joint centers in real time. Since the system returns a predefined skeletal representation composed of 32 joints (Figure 2.9), it can be considered closer to a model-based approach than to purely model-free keypoint detection methods. In addition to the 3D position of the joints, the SDK associates a confidence level, which provides an indication of the reliability of the tracking result [35].

The tracking pipeline relies on deep learning and combines both color and depth information. First, an initial body pose is estimated using a CNN, which processes the acquired RGB-D data to identify the main body joints. This initial estimate is then refined to improve the consistency of the reconstructed pose. Subsequently, an optimization procedure registers the predefined skeletal model to the acquired data. Finally, biomechanical constraints are applied to preserve anatomically plausible joint relationships and to reduce unrealistic body poses.

2.5.3 Hybrid pipelines

Hybrid pipelines combine deterministic components with learning-based algorithms in order to exploit the strengths of both approaches. In ML gait analysis, deep learning methods are often used to detect anatomical landmarks or body keypoints from images and videos, while kinematic models are subsequently applied to reconstruct three-dimensional motion. For example, 2D keypoints estimated from multiple camera views can be triangulated and used as input for IK or musculoskeletal modelling. Hybrid approaches can also follow the reverse strategy, in which a deterministic model is first used to obtain an initial structured

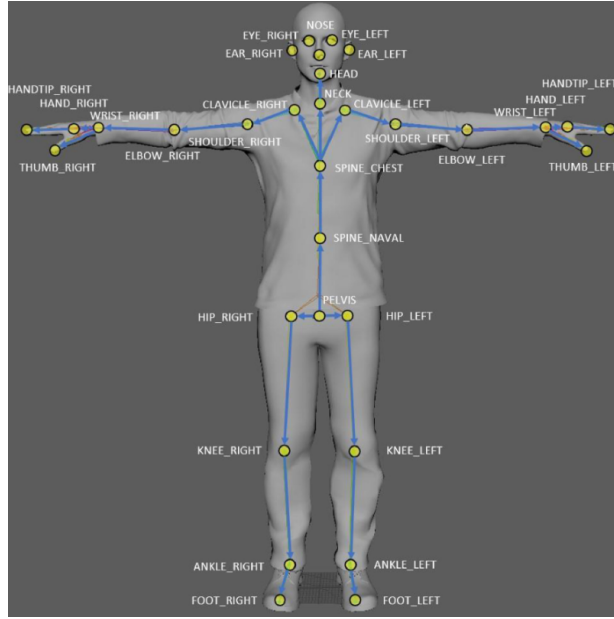


Figure 2.9: Predefined skeletal representation used by the Azure Kinect Body Tracking SDK. The model includes 32 joints distributed across the head, trunk, upper limbs, and lower limbs, whose three-dimensional positions are estimated in real time from RGB-D data. Reproduced from [36].

representation of the human body. From this representation, joint trajectories or virtual markers can be extracted and then processed by learning-based methods to refine or enhance the estimated motion. Nevertheless, the final accuracy of hybrid pipelines depends on both the quality of the initial reconstruction and the generalization capability of the learning-based model, making validation against MB reference systems essential before their use in clinical applications.

Chapter 3

Materials and methods

3.1 Study design and methodological overview

The study compared lower-limb kinematic outcomes obtained from ML approaches with those derived from a conventional MB motion capture system. The MB system was used as the reference method for three-dimensional gait analysis, and marker trajectories acquired during walking trials were processed according to the PiG biomechanical protocol to estimate lower-limb joint kinematics.

Two main ML approaches were considered, leading to four markerless processing pipelines. The first approach was based on OpenCap, an open-source application developed at Stanford University that estimates human movement from synchronized recordings acquired with two iPhones. OpenCap reconstructs 3D body keypoints, predicts 3D virtual anatomical markers through AGM, and estimates joint kinematics using OpenSim IK. In this thesis, OpenCap was evaluated both in its standard OpenSim-IK workflow and through a parallel OpenCap *PiG-like* workflow, in which virtual marker trajectories provided by the OpenCap AGM step were exported and processed to compute joint angles according to *PiG-like* biomechanical conventions.

The second approach was based on BRIDGE, a subject-specific RGB-D fitting pipeline based on the SMPL statistical body model. In this pipeline, a subject-specific lower-limb SMPL model is reconstructed from static RGB-D acquisitions and fitted to dynamic walking frames using an AICP algorithm. From the fitted model, hip, knee, and ankle joint centers are extracted, and sagittal joint angles are computed through trigonometric calculations using these centers together with color-based foot orientation.

In addition, a version of this pipeline based on AGM was evaluated and referred to as BRIDGE+AGM. In this workflow, the joint centers obtained from BRIDGE were used as input to an adapted pretrained Long Short-Term Memory LSTM

neural network to predict virtual anatomical markers. These markers enabled a *PiG-like* estimation of sagittal and frontal lower-limb joint angles, extending the analysis beyond the sagittal-plane kinematics directly available from BRIDGE.

Overall, the methodological workflow consists of four main stages: experimental data acquisition, MB PiG reference processing, ML data processing through different pipelines, and quantitative comparison of the resulting kinematic outputs.

3.2 Participants

Seven healthy participants were recruited for this study, including three males and four females. Participants had no known balance disorders that could affect gait. All subjects were able to walk independently and performed the walking trials at a self-selected speed. The main anthropometric characteristics of the participants are reported in Table 3.1.

Table 3.1: Anthropometric characteristics of the participants. Values are reported as mean $\pm$ standard deviation.

Characteristic	Value
Number of participants	7
Age (years)	28 ± 3
Height (cm)	175 ± 8
Body mass (kg)	67 ± 10

3.3 Experimental setup and data collection

3.3.1 Laboratory setup

Data collection was performed in the motion analysis laboratory of the Polito-BIOMed Lab at Politecnico di Torino. The laboratory was equipped with a conventional optoelectronic motion capture system composed of 12 IR cameras, operating at a sampling frequency of 120 Hz. This system was used as the MB reference for the analysis.

For the OpenCap acquisition, two smartphones, an iPhone 12 and an iPhone 17, were used at a frame rate of 60 Hz and mounted on tripods. According to the recommended setup, the two devices were positioned in front of the participant with respect to the direction of progression, each forming an angle of approximately 30° with the walking direction. This configuration provided two oblique views of the walking task, allowing the triangulation of the detected keypoints and the reconstruction of three-dimensional information.

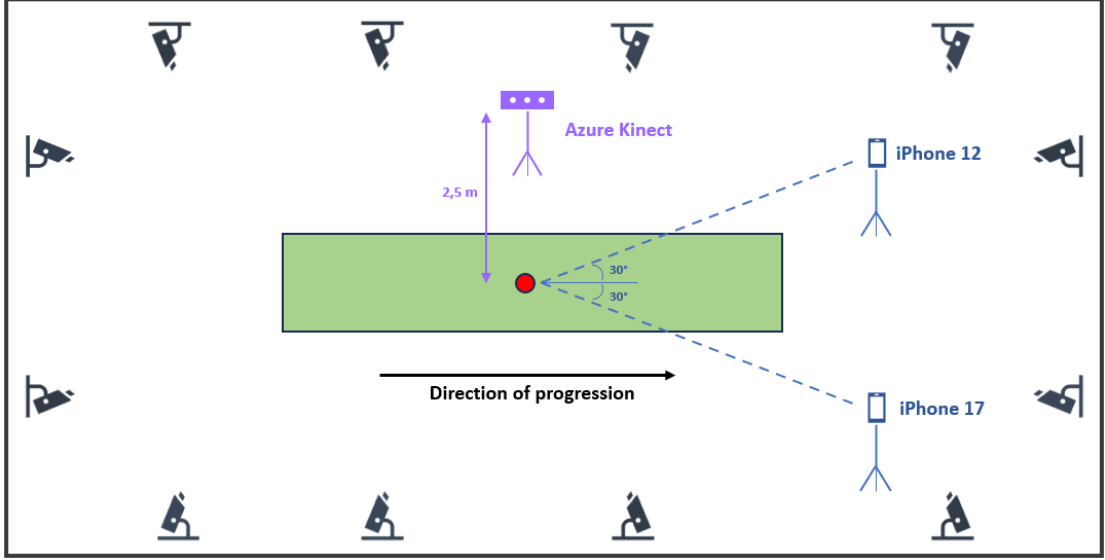


Figure 3.1: Top-view of the experimental setup. The configuration shown refers to the acquisition of a left gait cycle. For right gait cycle acquisitions, participants walked in the opposite direction, and the smartphones were repositioned accordingly on the opposite side of the walkway.

In addition, an Azure Kinect RGB-D camera was used to acquire the data required for the SMPL-based pipeline. The camera recorded RGB images with a resolution of 720×1280 pixels at 30 fps and depth images with a resolution of 640×576 pixels, in unbinned mode, at 30 fps. The Azure Kinect was mounted on a tripod and positioned laterally with respect to the direction of progression, at a distance of 2.5 meters from the center of the walkway and 5 meters from the background wall. The image coordinate system was aligned with the sagittal plane, defined by the direction of progression and the vertical direction. The RGB camera was operated with a manual exposure time of 8.33 ms to reduce motion blur that could occur with automatic exposure.

The overall arrangement of the acquisition systems is schematically represented in Figure 3.1, which shows the relative position of the MB cameras, the two smartphones used for OpenCap, and the Azure Kinect RGB-D camera with respect to the walkway and the direction of progression.

3.3.2 Subject preparation

Participants were asked to wear tight-fitting clothing in order to reduce marker occlusions and facilitate the identification of body segments during video acquisition. They were also asked to avoid black shorts, since dark clothing can reduce the

contrast in RGB images and may also affect the quality of depth estimation in RGB-D cameras. This precaution was adopted to facilitate body segmentation and improve the reliability of ML pose estimation. To facilitate foot identification through a color-based filtering procedure, participants wore two socks of different colors: a red sock on the right foot and a blue sock on the left foot.

Reflective markers were then placed on the pelvis and lower limbs according to the marker set required by the PiG lower body model, described in subsection 3.4.2. This configuration allowed the acquisition of the three-dimensional marker trajectories required to reconstruct body segments and compute the reference joint kinematics.

3.3.3 Data collection

Before data collection, both the MB stereophotogrammetric system and the dual-camera smartphone system used by OpenCap were calibrated. For each subject, static acquisitions were performed for the three acquisition setups before the dynamic walking trials. In the MB stereophotogrammetric system, the static trial was used to label the markers according to the lower-body PiG protocol. In OpenCap, a neutral standing pose was acquired to scale the musculoskeletal model to the subject's anthropometry. For BRIDGE, four static RGB-D views were recorded with the subject standing upright, including frontal, posterior, left lateral, and right lateral views. The subject-specific model of each lower limb was reconstructed using the frontal and posterior views together with the corresponding lateral view. After the static acquisitions, dynamic gait trials were acquired synchronously with the three acquisition systems. Participants were instructed to walk at a comfortable speed along a straight 5 m walkway. Ten gait trials were acquired for each subject, consisting of five left and five right gait cycles. For the subsequent analyses, only the first three valid acquisitions for each side were retained, resulting in three left and three right gait cycles per subject.

3.4 Marker-based reference system

The Vicon optoelectronic motion capture system was used as the reference system for the evaluation of the ML pipelines considered in this thesis. The three-dimensional marker trajectories acquired during the walking trials were processed using Vicon Nexus and subsequently analysed through the PiG lower body biomechanical model, in order to obtain the reference lower-limb joint kinematics.

3.4.1 Vicon motion capture system and data processing

The MB motion capture system used in this study was a Vicon optoelectronic system, operated through Vicon Nexus 2.15 software. Vicon Nexus is a platform

designed for the acquisition, processing, and analysis of human motion, and is widely used in biomechanical and clinical motion capture applications.

The system employed in the present study consisted of 12 Vero v2.2 IR cameras, which were used to detect the retroreflective markers attached to the subject. The cameras acquired data at a sampling frequency of 120 Hz. Motion reconstruction was performed using 16 passive retroreflective markers with a diameter of 14 mm, positioned according to the marker set required by the PiG lower body protocol.

Before data acquisition, the cameras were calibrated using an active wand, consisting of a rigid wand equipped with active markers placed at known distances from each other. This procedure allowed the system to determine the relative position and orientation of the cameras and to define the capture volume. The origin of the global laboratory reference frame was then set by placing the wand at the center of the capture volume.



Figure 3.2: Components of the MB stereophotogrammetric system. From left to right: Vero v2.2 IR camera, 14 mm passive marker, and Vicon Active Wand.

From an operational point of view, the workflow in Vicon Nexus followed a standardized sequence. First, the system was prepared by applying camera masks to exclude unwanted reflections in the environment, which could otherwise be incorrectly detected as markers. After calibration of the capture volume, the subject was created in Nexus from a predefined template associated with the selected biomechanical model and marker set. The anthropometric measurements required by the model were then entered, including body mass, height, inter-ASIS distance, leg length, knee width, and ankle width. The retroreflective markers were subsequently placed on the subject.

Before the dynamic walking trials, a static acquisition was performed. This trial was used to verify the correct visibility of all markers, to ensure that no additional markers or unwanted reflections were present, and to initialize the subject-specific biomechanical model. The static trial also allowed the estimation of the static offsets associated with the foot segment. Once this step was completed, the dynamic walking trials were acquired. After acquisition, the data were post-processed in Vicon Nexus. First, the central gait cycle was selected in order to ensure the best possible visibility of the markers within the capture volume. Marker trajectories

were then labeled by assigning each trajectory to the corresponding marker. When necessary, gaps caused by temporary marker occlusions were corrected using procedures based on interpolation. Once marker trajectories consistent with the recorded movement had been obtained, filtering was applied to reduce noise and improve signal smoothness. Specifically, the marker trajectories were filtered using the Woltring filter, which uses a fifth-order spline as the interpolating function. The processed trajectories were then used as input for the dynamic PiG pipeline. The dynamic processing allowed the coordinates of the main lower-limb joint centers to be estimated, including the hip joint center (HJC), knee joint center (KJC), and ankle joint center (AJC), and enabled the calculation of joint angles over time. Finally, marker trajectories, joint center coordinates, and joint angles were exported and subsequently used for comparison with the ML pipelines.

Since the IR sensor of the Azure Kinect operates in a spectral range compatible with that used by the Vicon system, around 850 nm, simultaneous acquisition with the two systems may generate interference. This can degrade the quality of the depth images and increase the number of invalid pixels. To reduce this issue, the acquisitions were synchronized through a trigger signal, with the aim of limiting the temporal overlap between the infrared illumination of the Vicon system and the depth acquisition of the Azure Kinect.

In the adopted setup, the Vicon system was configured as the master device, whereas the Azure Kinect was used in subordinate mode, consistently with the external synchronization strategy supported by the Azure Kinect DK [37]. In this configuration, the trigger signal provided by the Vicon system allowed the timing of the RGB-D camera acquisition to be controlled according to the settings defined in the GPO configuration file, as described in the Vicon Nexus documentation [38]. The use of synchronization output signals for triggering external devices is also described in the Vicon Systems Setup Guide [39]. This solution made it possible to acquire MB and RGB-D data synchronously, while reducing interference between the two systems.

3.4.2 PiG lower body model and marker set

The reference lower-limb kinematics were obtained using the PiG lower body model implemented in Vicon Nexus. The PiG model is based on the Newington-Helen Hayes gait model, and uses a predefined marker set together with subject-specific anthropometric measurements to reconstruct the pelvis and lower-limb segments and estimate joint kinematics [12]. In this study, only the lower body configuration was used, since the analysis focused on hip, knee, and ankle joint motion during gait.

The marker set consisted of 16 passive retroreflective markers placed on the

Table 3.2: Marker set used for the PiG lower body model.

Segment	Marker	Anatomical location
Pelvis	LASI	Left anterior superior iliac spine
Pelvis	RASI	Right anterior superior iliac spine
Pelvis	LPSI	Left posterior superior iliac spine
Pelvis	RPSI	Right posterior superior iliac spine
Thigh	LTHI	Lower lateral third of the left thigh
Thigh	RTHI	Upper lateral third of the right thigh
Knee	LKNE	Left lateral femoral epicondyle
Knee	RKNE	Right lateral femoral epicondyle
Shank	LTIB	Lower third of the left shank
Shank	RTIB	Upper third of the right shank
Ankle	LANK	Left lateral malleolus
Ankle	RANK	Right lateral malleolus
Foot	LHEE	Left calcaneus, at the same height as the toe marker
Foot	RHEE	Right calcaneus, at the same height as the toe marker
Foot	LTOE	Left second metatarsal head
Foot	RTOE	Right second metatarsal head

pelvis and lower limbs, as reported in Table 3.2. Four markers were placed on the pelvis, corresponding to the left and right anterior superior iliac spines (ASIS) and the left and right posterior superior iliac spines (PSIS). On each lower limb, markers were positioned on the thigh, knee, shank, ankle, heel, and second metatarsal head. This configuration allowed the PiG model to define the pelvis, thigh, shank, and foot segments and to track their motion during the dynamic walking trials.

As specified in the *Plug-in Gait Reference Guide*, a different placement was adopted for the thigh and shank markers on the right and left limbs. The left thigh and tibia markers were placed over the lower lateral third of the corresponding segments, whereas the right thigh and tibia markers were placed over the upper lateral third. This asymmetric placement is part of the standard PiG marker set and facilitates the identification of the two limbs during marker labeling and tracking.

In addition to marker trajectories, the PiG lower body model requires a set of anthropometric measurements to scale the model to the subject and to estimate joint center locations. In the present study, body mass, height, inter-ASIS distance, leg length, knee width, and ankle width were entered in Vicon Nexus before processing. These measurements contributed to the definition of the subject-specific biomechanical model and were used, together with the static acquisition, to estimate the main lower-limb joint centers.

The static trial played a central role in model calibration. During this acquisition, the subject stood in an upright position while all lower-body markers were visible. The marker positions recorded in the static trial were used to initialize the pelvis and lower-limb segments, define the reference configuration of the model, and estimate the static offsets associated with the foot segment. Once the static model had been defined, the dynamic PiG pipeline was applied to the walking trials to reconstruct segment motion and compute the lower-limb joint kinematics.

3.4.3 Segment reference frames

The PiG lower body model reconstructs the pelvis and lower-limb segments by defining, for each segment, a local coordinate system composed of an origin and three orthogonal axes. In general, one anatomical direction derived from marker or joint center positions is used as the dominant axis, while a second direction is used to define the anatomical plane of the segment. The remaining axes are then obtained through cross products, ensuring a right-handed coordinate system [12].

For clarity, the construction of the segment reference frames can be summarized using unit vectors. Given a vector $\mathbf{v}$, its normalized form is defined as:

$$\hat{\mathbf{v}} = \frac{\mathbf{v}}{\|\mathbf{v}\|}.$$

Pelvis

The origin is defined as the midpoint between the left and right ASIS markers:

$$\mathbf{O}_P = \frac{\mathbf{LASI} + \mathbf{RASI}}{2}.$$

The lateral axis is directed from the right to the left side of the pelvis, while the PSIS markers are used to define the pelvic plane; their midpoint is considered:

$$\mathbf{S} = \frac{\mathbf{LPSI} + \mathbf{RPSI}}{2}.$$

The pelvis axes can therefore be expressed as:

$$\hat{\mathbf{y}}_P = \frac{\mathbf{LASI} - \mathbf{RASI}}{\|\mathbf{LASI} - \mathbf{RASI}\|},$$

$$\hat{\mathbf{z}}_P = \frac{(\mathbf{O}_P - \mathbf{S}) \times \hat{\mathbf{y}}_P}{\|(\mathbf{O}_P - \mathbf{S}) \times \hat{\mathbf{y}}_P\|},$$

$$\hat{\mathbf{x}}_P = \hat{\mathbf{y}}_P \times \hat{\mathbf{z}}_P.$$

In this convention, the pelvis y axis is directed laterally towards the subject's left side, the z axis points approximately upwards, and the x axis points approximately forwards.

Femur

The femur coordinate system is defined with its origin at the KJC. The longitudinal axis is directed from the KJC to the HJC. The anatomical plane of the thigh is defined using the HJC, the knee marker, and the thigh marker. The anterior axis is obtained as the direction perpendicular to this plane, while the mediolateral axis is computed as the cross product between the longitudinal and anterior axes.

For the right femur, the auxiliary vectors are defined as:

$$\mathbf{v}_1^r = \mathbf{RTHI} - \mathbf{RKNE}, \quad \mathbf{v}_2^r = \mathbf{RHJC} - \mathbf{RKNE}.$$

The right femur reference frame is then computed as:

$$\mathbf{O}_{F,r} = \mathbf{RKJC},$$

$$\hat{\mathbf{z}}_{F,r} = \frac{\mathbf{RHJC} - \mathbf{RKJC}}{\|\mathbf{RHJC} - \mathbf{RKJC}\|},$$

$$\hat{\mathbf{x}}_{F,r} = \frac{\mathbf{v}_2^r \times \mathbf{v}_1^r}{\|\mathbf{v}_2^r \times \mathbf{v}_1^r\|},$$

$$\hat{\mathbf{y}}_{F,r} = \hat{\mathbf{z}}_{F,r} \times \hat{\mathbf{x}}_{F,r}.$$

For the left femur, the auxiliary vectors are defined as:

$$\mathbf{v}_1^l = \mathbf{LTHI} - \mathbf{LKNE}, \quad \mathbf{v}_2^l = \mathbf{LHJC} - \mathbf{LKNE}.$$

The left femur reference frame is computed as:

$$\mathbf{O}_{F,l} = \mathbf{LKJC},$$

$$\hat{\mathbf{z}}_{F,l} = \frac{\mathbf{LHJC} - \mathbf{LKJC}}{\|\mathbf{LHJC} - \mathbf{LKJC}\|},$$

$$\hat{\mathbf{x}}_{F,l} = \frac{\mathbf{v}_1^l \times \mathbf{v}_2^l}{\|\mathbf{v}_1^l \times \mathbf{v}_2^l\|},$$

$$\hat{\mathbf{y}}_{F,l} = \hat{\mathbf{z}}_{F,l} \times \hat{\mathbf{x}}_{F,l}.$$

Tibia

The tibia is represented by two anatomical reference frames, as defined in the PiG model: a torsioned tibia and an untorsioned tibia. Both reference frames have their origin at the AJC. The torsioned tibia describes the distal orientation of the shank and is used for the calculation of ankle joint angles. Conversely, the untorsioned tibia represents the proximal orientation of the shank and is used for the calculation of knee joint angles.

For the torsioned tibia, the longitudinal axis is directed from the AJC to the KJC. The anatomical plane of the shank is defined using the KJC, the AJC, and the tibial marker. The anterior axis is obtained as the direction perpendicular to this plane, while the mediolateral axis is computed as the cross product between the longitudinal and anterior axes.

For the right torsioned tibia, the auxiliary vectors are defined as:

$$\mathbf{v}_1^r = \mathbf{RTIB} - \mathbf{RAJC}, \quad \mathbf{v}_2^r = \mathbf{RKJC} - \mathbf{RAJC}.$$

The right torsioned tibia reference frame is then computed as:

$$\begin{aligned} \mathbf{O}_{T,r} &= \mathbf{RAJC}, \\ \hat{\mathbf{z}}_{T,r} &= \frac{\mathbf{RKJC} - \mathbf{RAJC}}{\|\mathbf{RKJC} - \mathbf{RAJC}\|}, \\ \hat{\mathbf{x}}_{T,r} &= \frac{\mathbf{v}_2^r \times \mathbf{v}_1^r}{\|\mathbf{v}_2^r \times \mathbf{v}_1^r\|}, \\ \hat{\mathbf{y}}_{T,r} &= \hat{\mathbf{z}}_{T,r} \times \hat{\mathbf{x}}_{T,r}. \end{aligned}$$

For the left torsioned tibia, the auxiliary vectors are defined as:

$$\mathbf{v}_1^l = \mathbf{LTIB} - \mathbf{LAJC}, \quad \mathbf{v}_2^l = \mathbf{LKJC} - \mathbf{LAJC}.$$

The left torsioned tibia reference frame is computed as:

$$\begin{aligned} \mathbf{O}_{T,l} &= \mathbf{LAJC}, \\ \hat{\mathbf{z}}_{T,l} &= \frac{\mathbf{LKJC} - \mathbf{LAJC}}{\|\mathbf{LKJC} - \mathbf{LAJC}\|}, \\ \hat{\mathbf{x}}_{T,l} &= \frac{\mathbf{v}_1^l \times \mathbf{v}_2^l}{\|\mathbf{v}_1^l \times \mathbf{v}_2^l\|}, \\ \hat{\mathbf{y}}_{T,l} &= \hat{\mathbf{z}}_{T,l} \times \hat{\mathbf{x}}_{T,l}. \end{aligned}$$

The untorsioned tibia is obtained by rotating the torsioned tibia around its longitudinal axis by the tibial torsion angle τ . This correction aligns the frontal plane of the proximal shank with the knee flexion/extension axis. Therefore, the untorsioned tibia is used to describe the proximal motion of the shank when calculating knee joint angles.

Foot

The foot coordinate system is defined with its origin at the AJC. During the static trial, PiG uses the heel marker to estimate the static foot offsets. First, a heel-based foot segment is constructed using the line between the toe and heel markers as the primary axis, corresponding to the foot longitudinal axis. This axis is defined as the local z axis. The local y axis is taken from the untorsioned tibia, while the local x axis, pointing downwards, is obtained by cross product.

A second foot segment, referred to as the uncorrected foot, is then constructed using the line between the AJC and the toe marker as the primary axis. The remaining axes are defined using the same procedure described for the heel-based foot segment, with the y axis derived from the untorsioned tibia and the x axis obtained by cross product.

The static foot offsets are calculated from the relative orientation between these two foot segments. Specifically, PiG computes the YXZ Cardan angles required to rotate the uncorrected foot segment into the heel-based foot segment. This calculation is performed for each frame of the static trial, and the mean angles are then retained. The plantar-flexion offset is defined as the rotation around the local y axis, whereas the rotation offset is defined as the rotation around the local x axis. The rotation around the local z axis is ignored.

During dynamic trials, the foot is first reconstructed as the uncorrected foot. Its longitudinal axis is directed from the AJC towards the toe marker:

$$\mathbf{O}_{Foot} = \mathbf{AJC},$$

$$\hat{\mathbf{z}}_{Foot,0} = \frac{\mathbf{TOE} - \mathbf{AJC}}{\|\mathbf{TOE} - \mathbf{AJC}\|}.$$

The mediolateral direction is derived from the untorsioned tibia, and the remaining axes are obtained by enforcing orthogonality:

$$\hat{\mathbf{y}}_{Foot,0} = \hat{\mathbf{y}}_T^{untorsioned},$$

$$\hat{\mathbf{x}}_{Foot,0} = \hat{\mathbf{y}}_T^{untorsioned} \times \hat{\mathbf{z}}_{Foot,0}.$$

The final foot coordinate system is then obtained by applying the static offsets calculated during the static trial. Specifically, the uncorrected foot is first rotated around its y axis by the plantar-flexion offset and then around its updated x axis by the rotation offset:

$$\mathbf{R}_{Foot} = \mathbf{R}_x(\text{RotOff}) \mathbf{R}_y(\text{PlantFlexOff}) \mathbf{R}_{Foot,0}.$$

3.4.4 Joint angles calculation

Once the local segment reference frames have been defined, joint angles are calculated by comparing the relative orientation between the proximal segment, or parent segment, and the distal segment, or child segment, with respect to the joint of interest. In the PiG model, angular outputs are derived from Cardan angles using a YXZ sequence, as specified in the *Plug-in Gait Reference Guide* [12]. The use of ordered rotations to describe lower-limb joint kinematics is also consistent with the conventional gait analysis framework described by Kadaba et al. [8]. The proximal segment is considered as the fixed reference, whereas the distal segment is considered as the moving element. The relative orientation between the two segments can therefore be expressed by the relative rotation matrix:

$$\mathbf{R}_{rel} = \mathbf{R}_{prox}^T \mathbf{R}_{dist}.$$

From this matrix, the three angular components are extracted, corresponding to flexion/extension, abduction/adduction, and internal/external rotation, respectively. More specifically, the first rotation is associated with flexion/extension, the second with abduction/adduction, and the third with internal/external rotation. This representation allows joint kinematics to be described in the three anatomical planes, while maintaining a convention consistent with the PiG model.

For the hip joint, the proximal segment is represented by the pelvis, whereas the distal segment is represented by the femur. Hip angles are therefore calculated by comparing the orientation of the femur with respect to the pelvis:

$$\mathbf{R}_{Hip} = \mathbf{R}_{Pelvis}^T \mathbf{R}_{Femur}.$$

The decomposition of this matrix provides the hip flexion/extension, adduction/abduction, and internal/external rotation components.

For the knee joint, the proximal segment is represented by the femur, whereas the distal segment is represented by the untorsioned tibia. This choice is consistent with the PiG convention, since the untorsioned tibia represents the proximal orientation of the tibia and is therefore used to describe knee kinematics:

$$\mathbf{R}_{Knee} = \mathbf{R}_{Femur}^T \mathbf{R}_{Tibia}^{untorsioned}.$$

The decomposition of the relative matrix provides the knee flexion/extension, varus/valgus, and internal/external rotation components. In the PiG model, the frontal-plane knee component is generally described as varus/valgus, or adduction/abduction.

For the ankle joint, the proximal segment is represented by the torsioned tibia, whereas the distal segment is represented by the foot. The torsioned tibia is used because it represents the distal orientation of the tibia, that is, the orientation associated with the ankle motion axis:

$$\mathbf{R}_{Ankle} = \left(\mathbf{R}_{Tibia}^{torsioned} \right)^T \mathbf{R}_{Foot}.$$

In this case as well, the angles are derived using the YXZ sequence, providing the dorsi/plantar flexion, inversion-eversion, and internal/external rotation components of the foot with respect to the tibia. However, the PiG guide specifies that, since the foot segment is defined with a different orientation with respect to the tibial segment, a $+90^\circ$ offset is applied to the ankle flexion component. This correction only affects the flexion component and does not change the interpretation of the other angular components.

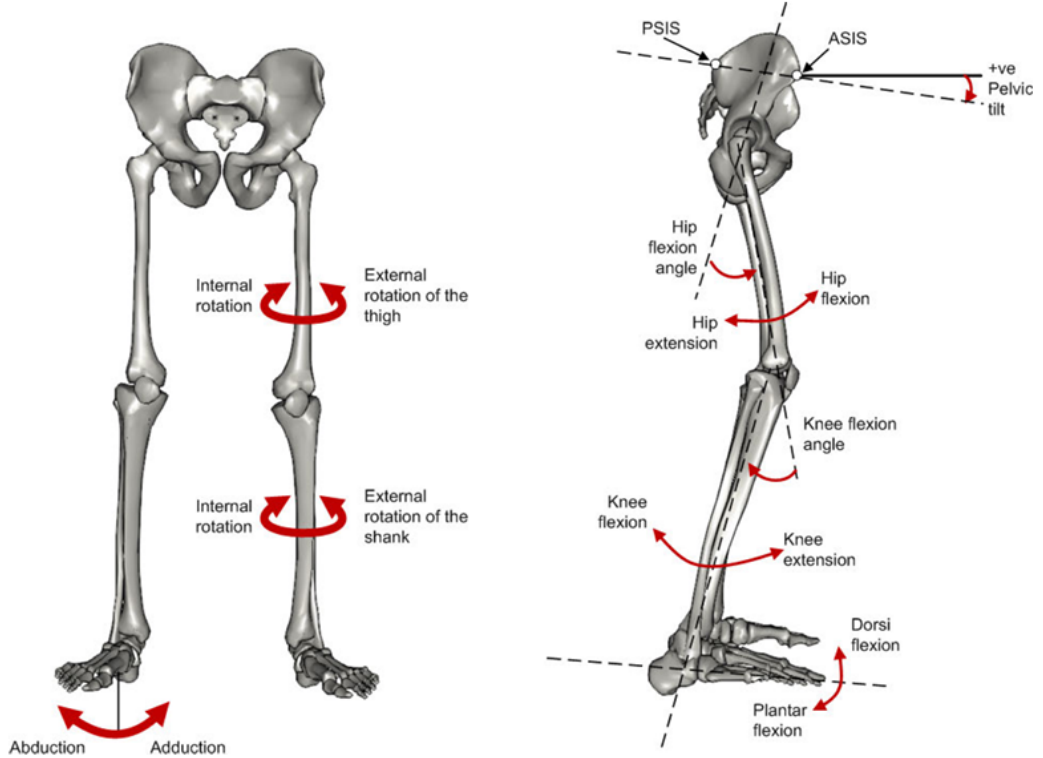


Figure 3.3: Schematic representation of the lower-limb joint angle conventions used in the Plug-in Gait model. Reproduced from [12].

The sign conventions reported in this section refer to the native PiG outputs exported from Vicon Nexus. For the final comparison across motion capture pipelines, joint angle signals were subsequently expressed according to the convention adopted by Benedetti et al. [40], as described in section 3.7.

The segment pairs used for joint angle calculation are summarized in Table 3.3, while the sign convention adopted for the interpretation of the native PiG angular components is reported in Table 3.4.

Table 3.3: Plug-in Gait lower-limb joint angle calculation.

Joint	Proximal segment	Distal segment
Hip	Pelvis	Femur
Knee	Femur	Untorsioned tibia
Ankle	Torsioned tibia	Foot

Table 3.4: Sign convention for Plug-in Gait lower-limb joint angles.

Joint	Positive value	Negative value
Hip	Flexion	Extension
	Adduction	Abduction
	Internal rotation	External rotation
Knee	Flexion	Extension
	Varus / adduction	Valgus / abduction
	Internal rotation	External rotation
Ankle	Dorsiflexion	Plantarflexion
	Inversion	Eversion
	Internal rotation	External rotation

3.5 OpenCap markerless pipeline

OpenCap was used in this thesis as a video-based ML pipeline for estimating lower-limb kinematics during gait. The platform allows human movement to be acquired using smartphones and progressively transforms video data into biomechanical outputs by combining computer vision algorithms, deep learning, and musculoskeletal modelling. In the original work by Uhlich et al., OpenCap is presented as an open-source platform for computing human movement kinematics and dynamics from videos acquired with two or more smartphones [41].

In the present study, OpenCap was applied to walking trials acquired with two synchronized iPhones. The pipeline was used to obtain lower-limb joint angles computed through OpenSim IK, and to export the 3D virtual marker

trajectories generated after the AGM step. These two levels of output were considered separately: on the one hand, the joint angles directly provided by OpenCap represented the standard output of the platform; on the other hand, the OpenCap-derived virtual markers were used in a parallel *PiG-like* pipeline to compute joint angles according to biomechanical conventions consistent with the MB PiG workflow. This second analysis made it possible to evaluate the added value of OpenSim IK over direct *PiG-like* computation from OpenCap AGM-derived virtual markers.

The overall workflow adopted for the OpenCap analysis of dynamic trials is summarized in Figure 3.4. The diagram highlights both the standard OpenCap processing path, leading to joint angles computed through OpenSim IK, and the parallel *PiG-like* processing path applied to the post-augmentation virtual marker trajectories.

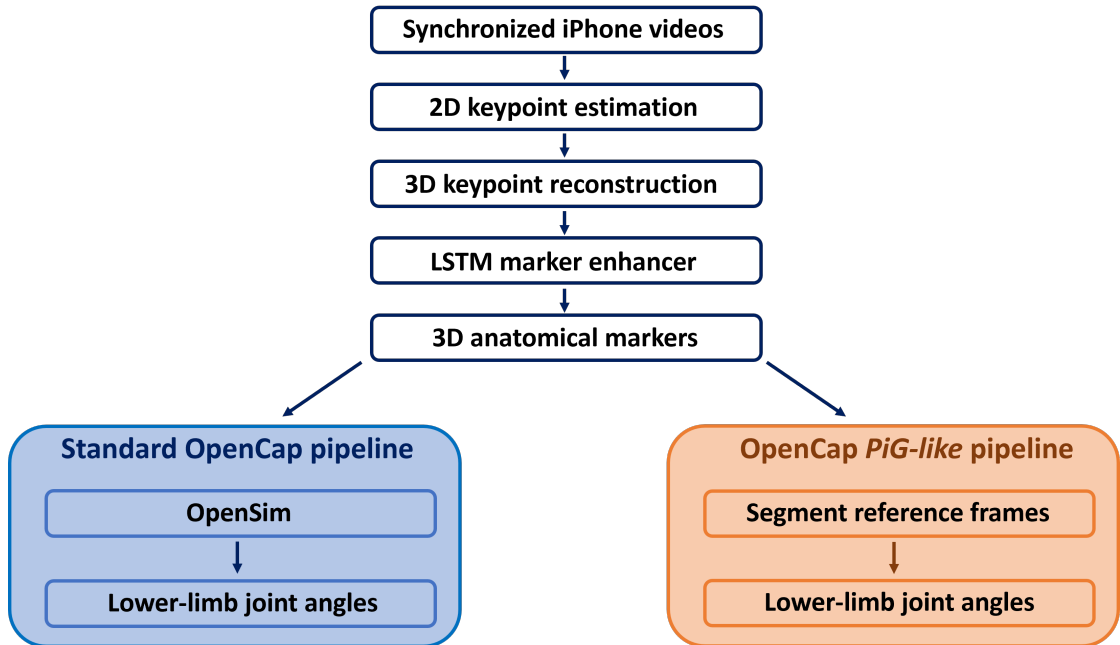


Figure 3.4: Schematic representation of the OpenCap workflow adopted in this thesis.

3.5.1 Overview of OpenCap

OpenCap is a ML platform designed to estimate human movement from videos acquired with smartphones. Unlike MB stereophotogrammetric systems, it does not require physical markers to be placed on the subject, but uses multi-camera video recordings to estimate body pose and reconstruct motion in three dimensions. The

platform integrates a web application for acquisition management, pose estimation algorithms, keypoint triangulation, AGM, and OpenSim-based musculoskeletal modelling [41].

In the standard workflow, an OpenCap session begins with calibration of the multi-camera system and acquisition of a static standing pose of the subject. The static pose is used to scale the musculoskeletal model to the participant's anthropometry. Dynamic trials are then acquired; in the present study, these consisted of walking trials. The videos recorded by the two smartphones are processed to estimate two-dimensional body keypoints in each frame; these keypoints are then triangulated to obtain a three-dimensional representation of the movement.

Starting from the three-dimensional keypoints, OpenCap applies a AGM step, whose purpose is to estimate a set of virtual markers compatible with the subsequent musculoskeletal processing. The resulting ML trajectories are then used in OpenSim to perform model scaling and IK. The standard pipeline output includes the file containing the joint angles; however, it is also possible to download the file containing both the keypoint trajectories estimated using HRNet before AGM and the post-augmentation virtual marker trajectories.

An example of the OpenCap graphical interface after processing a dynamic walking trial is shown in Figure 3.5.

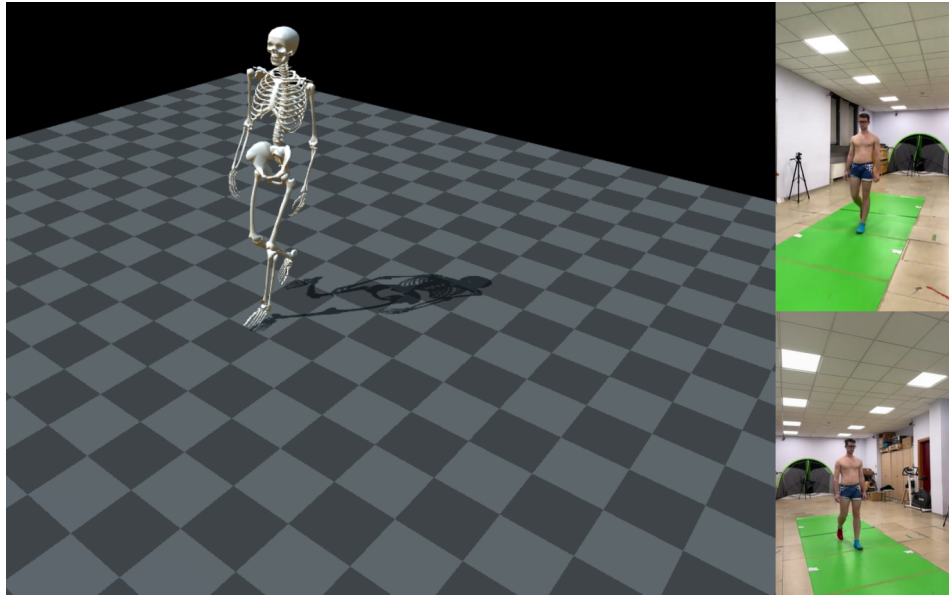


Figure 3.5: The image shows a single frame of the walking trial, with the reconstructed musculoskeletal model displayed on the left and the synchronized smartphone videos shown on the right.

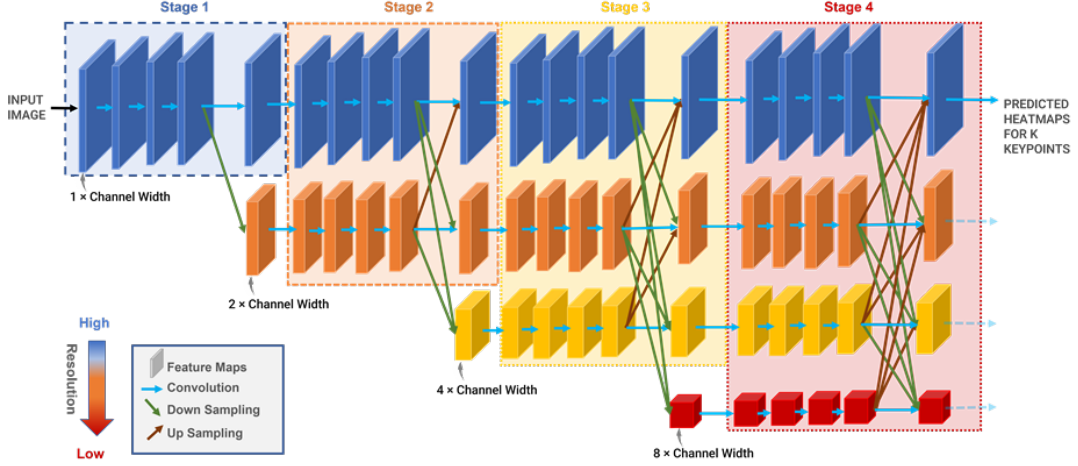


Figure 3.6: General architecture of an HRNet model. Reproduced from MathWorks documentation [43].

3.5.2 Two-dimensional pose estimation with HRNet

The first computational step of the OpenCap pipeline consists of estimating the two-dimensional pose of the subject in the acquired videos. For each frame and for each camera view, a set of body keypoints is identified in the image plane. These points represent the two-dimensional position of anatomical or joint-related regions, such as the pelvis, knees, ankles, and feet, and provide the basic information from which the three-dimensional pose is subsequently reconstructed.

OpenCap uses a deep learning-based approach for human pose estimation. In the original paper, HRNet is reported as the network used for 2D keypoint estimation. HRNet, which stands for High-Resolution Network, is a convolutional neural network developed to maintain high-resolution representations throughout the entire feature extraction process, rather than recovering spatial resolution only in the final stages of the network (Figure 3.6). This characteristic is particularly useful for pose estimation, since accurate keypoint localization requires detailed spatial information to be preserved [42].

Within the OpenCap pipeline, 2D pose estimation is performed separately on the different video views. Since the cameras are calibrated, the keypoints detected in the synchronized videos can be geometrically combined to estimate their position in three-dimensional space. The quality of this step is crucial because errors in 2D keypoint localization can propagate to triangulation, AGM, and finally to joint angle calculation.

3.5.3 LSTM-based marker enhancement

The three-dimensional keypoints obtained from triangulation still represent a sparse description of the human body. They mainly correspond to joint locations or body regions that can be identified in the videos, but they do not directly match the anatomical markers used in traditional biomechanical models. This limitation is relevant because an overly sparse set of points may not sufficiently constrain the musculoskeletal model during IK, especially for segments such as the pelvis and femur. In addition, keypoints estimated from videos may be affected by noise or occlusions, with errors potentially propagating to three-dimensional reconstruction and joint kinematic calculation.

To reduce these issues, OpenCap includes a marker enhancement step, in which a deep learning model, referred to as the marker enhancer, transforms a sparse set of video-based keypoints into a denser set of virtual anatomical markers. In the work by Falisse et al., the marker enhancer is described as a model capable of predicting the three-dimensional position of 43 anatomical markers from the three-dimensional position of 20 video keypoints. The aim is therefore to make the information obtained from videos more similar to that provided by a MB system, thereby improving the subsequent musculoskeletal processing [44].

As shown in Figure 3.7, the marker enhancer consists of two distinct models: an arm model and a body model. The arm model predicts the three-dimensional position of 8 anatomical markers located on the upper limbs from 7 keypoints related to the upper limbs and shoulders. The body model, instead, predicts the three-dimensional position of 35 anatomical markers located on the shoulders, trunk, and lower limbs from 15 keypoints related to the shoulders and lower body. Since the present study focuses on lower-limb kinematics during gait, the body model represents the most relevant component of the marker enhancement procedure. Both models also include the subject's height and body mass as inputs, while all marker positions are expressed with respect to a root marker, defined as the midpoint between the hip keypoints [44].

The use of a LSTM network is motivated by the temporal nature of human movement. Marker trajectories are not independent across consecutive frames, but follow a continuous and biomechanically coherent pattern. An LSTM network can exploit this temporal dependency, using the information contained in the sequence of keypoints to estimate virtual markers that are smoother and less sensitive to localized errors in individual frames. In the marker enhancer integrated into OpenCap, this property is used to transform three-dimensional keypoint trajectories into anatomical marker trajectories that are more suitable for joint kinematic calculation.

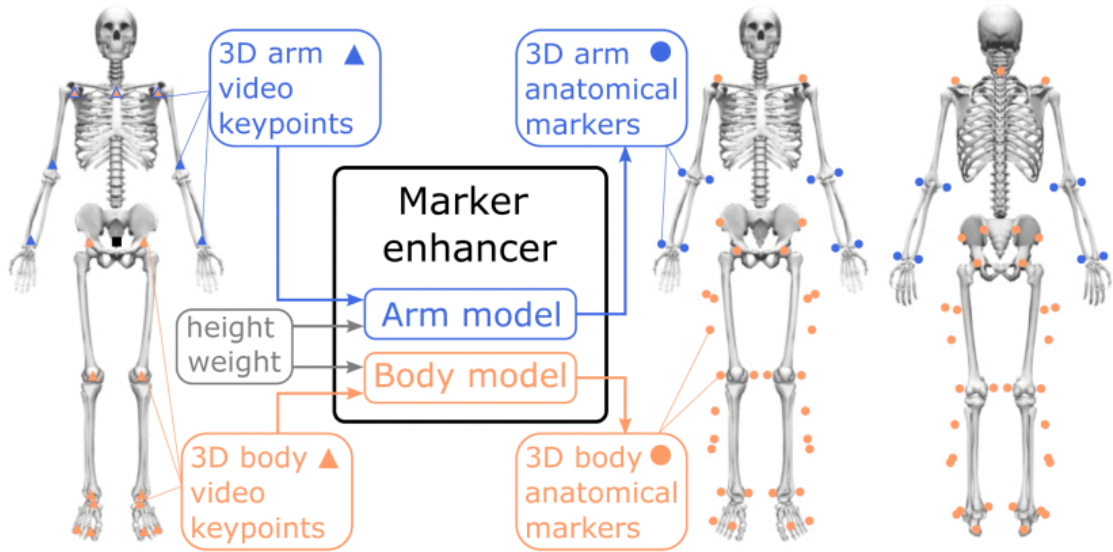


Figure 3.7: Schematic representation of the marker enhancer model. The body model predicts a set of 35 anatomical markers from 15 sparse video keypoints. Reproduced from [44].

3.5.4 OpenSim processing

The final stage of the standard OpenCap pipeline consists of the musculoskeletal processing of the virtual markers estimated by the AGM network. OpenCap uses OpenSim, an open-source platform for musculoskeletal modelling and simulation, to scale a generic model to the subject and compute joint kinematics through IK. In OpenSim, IK determines, for each frame, the model configuration that minimizes the distance between the marker trajectories provided as input and the corresponding virtual markers defined on the model [45].

In the OpenCap workflow, the static pose acquired at the beginning of the session is used to adapt the musculoskeletal model to the subject’s anthropometry. After this scaling step, the virtual marker trajectories obtained from the videos are provided to OpenSim as input for IK. The joint angles calculated through IK are exported by OpenCap in a *.mot* file.

Since the present analysis focused on lower-limb kinematics, six joint angle components were considered: hip flexion/extension, hip abduction/adduction, knee flexion/extension, knee varus/valgus, ankle dorsi/plantar flexion, and ankle internal/external rotation. However, only a subset of these variables is directly provided by OpenCap through the standard OpenSim-IK output, namely sagittal-plane kinematics together with hip ab/adduction. These native OpenCap outputs are reported in Table 3.5.

Table 3.5: Native OpenCap joint angle outputs selected for the analysis. Each output is provided for both the right and left side.

Joint	OpenCap output	Anatomical interpretation
Hip	hip_flexion	Hip Flexion/Extension
Hip	hip_adduction	Hip Abduction/Adduction
Knee	knee_angle	Knee Flexion/Extension
Ankle	ankle_angle	Ankle Dorsi/Plantarflexion

It should be noted that the analysis was not limited to the native joint angle outputs exported by OpenCap. In parallel, the post-augmentation virtual marker trajectories were processed through a separate *PiG-like* workflow. In this case, segment reference frames were reconstructed from the virtual marker trajectories, and all six joint angle components considered in this thesis were computed using conventions consistent with the MB PiG workflow.

3.6 BRIDGE and BRIDGE+AGM pipelines

The BRIDGE pipeline was used in this thesis as the single-camera RGB-D approach for estimating lower-limb kinematics during gait. Unlike OpenCap, which relies on videos acquired with smartphones, BRIDGE uses RGB-D data obtained with Azure Kinect. The first step consists in obtaining a subject-specific three-dimensional representation of the lower limbs, from which anatomical points and kinematic trajectories can be extracted during dynamic walking trials.

The adopted workflow was based on the pipeline proposed by Balta, in which a subject-specific SMPL lower-limb model is calibrated from static acquisitions and subsequently fitted to dynamic data. In BRIDGE, the joint center trajectories estimated from the fitted model are used to compute sagittal joint angles through trigonometric calculations, together with color-based foot orientation for the ankle. This approach allows lower-limb kinematics to be estimated from ML RGB-D data, although it is limited to sagittal-plane kinematics.

The BRIDGE pipeline was also evaluated in an AGM-based configuration, referred to as BRIDGE+AGM. In this workflow, the joint centers obtained from BRIDGE were used as input to an adapted pretrained LSTM neural network to predict virtual anatomical markers compatible with the PiG marker set. These markers were then used to reconstruct segment reference frames and compute lower-limb joint angles through a *PiG-like* workflow, enabling the estimation of both sagittal- and frontal-plane kinematics.

Overall, the two processing paths considered in this thesis can therefore be interpreted as complementary: BRIDGE directly estimates sagittal kinematics

from SMPL-derived joint centers and foot orientation, whereas BRIDGE+AGM uses these keypoints to predict anatomical marker trajectories suitable for *PiG-like* three-dimensional joint kinematic calculation.

The overall workflow adopted for the BRIDGE-based analysis is summarized in Figure 3.8. The diagram highlights the two processing paths considered in this thesis: BRIDGE and BRIDGE+AGM.

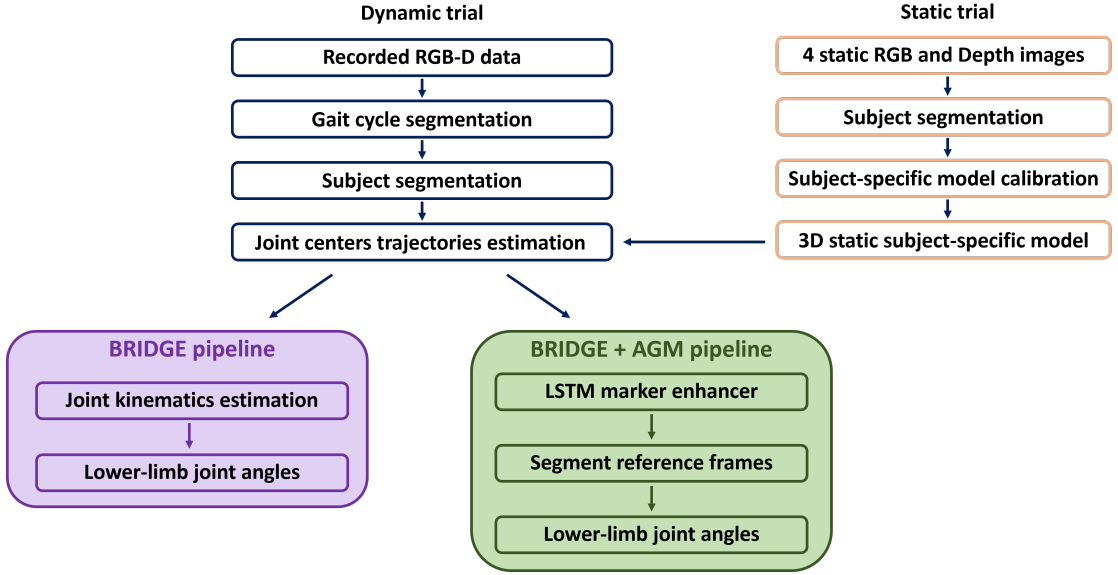


Figure 3.8: Schematic representation of the BRIDGE and BRIDGE+AGM workflows adopted in this thesis.

3.6.1 SMPL body model

In computer graphics and animation, realistic body movements are commonly reproduced by transferring the motion of an articulated skeleton to the external surface of a 3D mesh. Within this framework, the skeleton defines the hierarchical structure and joint transformations, while the mesh represents the visible geometry of the body. One of the main challenges of these approaches is achieving realistic skin deformations during articulated movements, particularly around highly deformable regions such as shoulders, elbows, hips, and knees. Among the available techniques, the most widely adopted approach is Linear Blend Skinning (LBS), mainly because of its simplicity and computational efficiency, which make it suitable for real-time animation and standard rendering pipelines. In the LBS framework, mesh deformation is computed through a weighted linear combination of rigid bone transformations, where each vertex is influenced by one or more

joints of the underlying skeleton through scalar coefficients known as blend weights. Despite its widespread use, LBS presents important limitations when modelling realistic anatomical deformations. In particular, the linear interpolation of rigid transformations often produces unrealistic artifacts in highly articulated regions, commonly referred to in the literature as “taffy” and “bowtie” effects (Figure 3.9). These artifacts arise from the inability of the model to accurately represent non-rigid soft-tissue deformations and volume preservation during joint rotations.

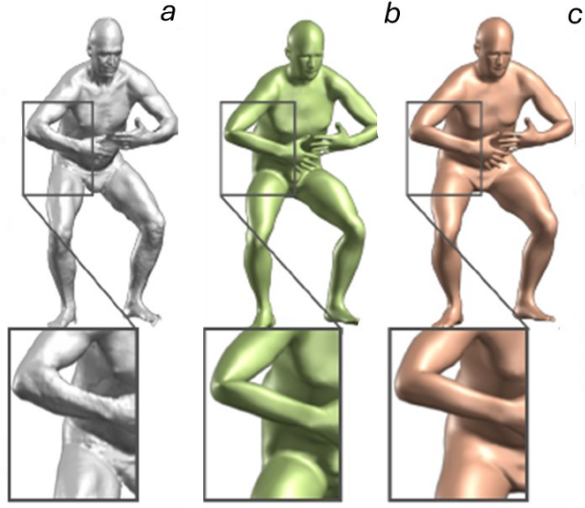


Figure 3.9: a) 3D scan of the human body; b) Linear Blend Skinning result, showing visible deformation artifacts around the right elbow; c) SMPL reconstruction, providing a more realistic representation of pose-dependent body deformation. Reproduced from [4].

Loper et al. introduced the SMPL model in 2015 to address some of the limitations of previous human body representations. SMPL is a realistic statistical 3D human body model learned from thousands of 3D scans, and it was designed to combine the efficiency and compatibility of traditional skinning approaches with the realism of data-driven human body models. Specifically, SMPL combines standard skinning with learned blend shapes, enabling the representation of different body shapes as well as realistic pose-dependent deformations. Unlike conventional skinning techniques, it incorporates learned shape-dependent and pose-dependent blend shapes that correct skinning artifacts while remaining fully compatible with standard graphics and animation pipelines. The following discussion of the SMPL formulation and its components follows the original work *SMPL: A Skinned Multi-Person Linear Model* by Loper et al. (2015) [46].

The SMPL model adopts a vertex-based skinning formulation in which the human

body is represented by an articulated mesh driven by a skeletal structure. Similarly to SCAPE, SMPL separates body shape into an identity-dependent component and a non-rigid pose-dependent component. However, unlike triangle-deformation-based models, it represents corrective deformations directly in vertex space through blend shapes, each encoded as a vector of concatenated vertex offsets.

The model starts from an artist-created template mesh composed of $N = 6890$ vertices and $K = 23$ joints. The same mesh topology is used for male and female bodies, and the template is designed with a clean quad structure, a segmentation into body parts, initial blend weights, and an associated skeletal rig. Following the notation introduced by Loper et al., meshes and blend shapes are represented as vectors of vertices denoted by bold capital letters, such as $\mathbf{X}$, while individual vertices are denoted by bold lowercase letters, such as $\mathbf{x}_i$.

The model is defined starting from a mean template shape $\bar{\mathbf{T}} \in \mathbb{R}^{3N}$, obtained by concatenating the coordinates of the N vertices in the zero pose $\boldsymbol{\theta}^*$. The pose of the body is described by a standard skeletal rig. For each part k , the relative rotation with respect to its parent in the kinematic tree is represented by an axis-angle vector $\vec{\omega}_k \in \mathbb{R}^3$. Since the rig contains $K = 23$ joints, the complete pose vector is defined as

$$\boldsymbol{\theta}^* = [\vec{\omega}_0^T, \dots, \vec{\omega}_K^T]^T$$

and contains 72 parameters, corresponding to three parameters for each articulated part plus three additional parameters for the global root orientation. Each axis-angle vector is then converted into a rotation matrix through the Rodrigues formula.

The statistical nature of SMPL derives from the fact that its shape and deformation are learned from registered 3D body scans. The model is mainly controlled by two sets of parameters:

- $\boldsymbol{\beta}$, the shape parameter vector, whose scalar coefficients weight the learned shape blend shapes. These parameters encode inter-subject body shape variability, allowing the model to represent different body proportions and body sizes.
- $\boldsymbol{\theta}$, the pose parameter vector, which controls the articulated configuration of the body by specifying the relative joint rotations in the kinematic tree. The pose also drives pose-dependent blend shapes, which correct skinning artifacts and model non-rigid deformations induced by skinning.

Before applying skinning, SMPL adapts the template mesh through two blend-shape functions:

- **Shape blend shapes.** The body shape of an individual is represented by a

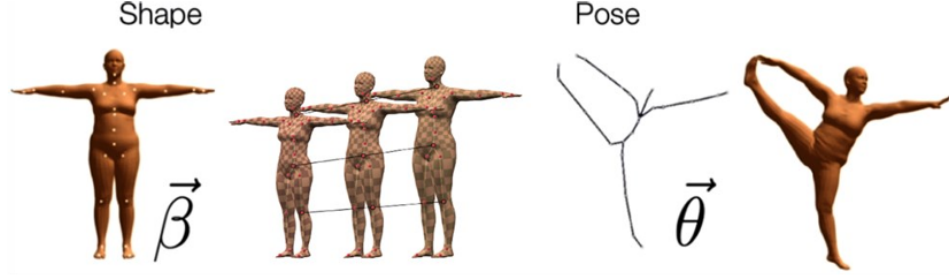


Figure 3.10: Parameters of the SMPL model. Reproduced from [4].

linear combination of learned shape displacements:

$$B_S(\beta; \mathcal{S}) = \sum_{n=1}^{|\beta|} \beta_n \mathbf{S}_n.$$

Here, $|\beta|$ is the number of shape coefficients, while each $\mathbf{S}_n \in \mathbb{R}^{3N}$ represents an orthonormal principal component of shape displacement learned from registered training meshes.

- **Pose blend shapes.** Pose-dependent deformations are modeled as vertex displacements from the rest template:

$$B_P(\theta; \mathcal{P}) = \sum_{n=1}^{9K} (\mathbf{R}_n(\theta) - \mathbf{R}_n(\theta^*)) \mathbf{P}_n.$$

In this expression, $\mathbf{R}(\theta) \in \mathbb{R}^{9K}$ is the vector obtained by concatenating the elements of the relative rotation matrices of the $K = 23$ joints, resulting in $9K = 207$ scalar values. The term $\mathbf{R}_n(\theta)$ denotes the n -th element of this vector, while θ^* is the rest pose. Each $\mathbf{P}_n \in \mathbb{R}^{3N}$ is a pose blend shape. Subtracting $\mathbf{R}_n(\theta^*)$ ensures that the contribution of the pose blend shapes is zero in the rest pose.

In SMPL, the skeletal structure is adapted to the specific body shape rather than being fixed across all subjects. This is necessary because variations in body proportions also affect the location of the joints. Therefore, the joint centers are predicted from the body mesh after the shape-dependent deformation has been applied to the mean template. This relationship is defined as:

$$J(\beta; \mathcal{J}, \bar{\mathbf{T}}, \mathcal{S}) = \mathcal{J}(\bar{\mathbf{T}} + B_S(\beta; \mathcal{S})).$$

$\mathcal{J}$ is the learned joint regressor. Since all SMPL meshes share the same topology, the regressor can estimate each joint location as a weighted combination of mesh vertices.

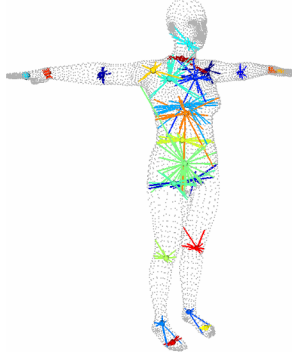


Figure 3.11: Set of vertices and corresponding weights used to estimate each joint location. Reproduced from [46].

The learned weights are sparse, meaning that each joint is mainly determined by a limited set of relevant surface vertices (Figure 3.11).

Before applying skinning, SMPL constructs a pose- and shape-corrected template in the rest pose space. At the vertex level, the corrected position of the i -th vertex is obtained by adding to the corresponding mean-template vertex both the shape-dependent displacement and the pose-dependent corrective displacement:

$$\mathbf{t}_{P,i}(\boldsymbol{\beta}, \boldsymbol{\theta}; \bar{\mathbf{T}}, \mathcal{S}, \mathcal{P}) = \bar{\mathbf{t}}_i + \sum_{m=1}^{|\beta|} \beta_m \mathbf{s}_{m,i} + \sum_{n=1}^{9K} (\mathbf{R}_n(\boldsymbol{\theta}) - \mathbf{R}_n(\boldsymbol{\theta}^*)) \mathbf{p}_{n,i}.$$

Here, $\bar{\mathbf{t}}_i$ is the i -th vertex of the mean template, $\mathbf{s}_{m,i}$ is the displacement associated with the m -th shape blend shape, and $\mathbf{p}_{n,i}$ is the displacement associated with the n -th pose blend shape. The scalar coefficients β_m determine the contribution of each shape component, while the term $(\mathbf{R}_n(\boldsymbol{\theta}) - \mathbf{R}_n(\boldsymbol{\theta}^*))$ activates the pose-dependent correction with respect to the rest pose.

Once the subject-specific joint locations have been estimated and the corrected template has been obtained, the model can be articulated. In this final step, LBS is applied to deform the corrected template according to the pose parameters, and the final position of the i -th vertex is obtained as:

$$\mathbf{t}'_i = \sum_{k=1}^K w_{k,i} \mathbf{G}'_k(\boldsymbol{\theta}, J(\boldsymbol{\beta}; \mathcal{J}, \bar{\mathbf{T}}, \mathcal{S})) \mathbf{t}_{P,i}(\boldsymbol{\beta}, \boldsymbol{\theta}; \bar{\mathbf{T}}, \mathcal{S}, \mathcal{P}).$$

In this expression, $w_{k,i}$ is the blend weight defining the influence of joint k on vertex i , while $\mathbf{G}'_k$ denotes the transformation of joint k relative to the rest pose. Consequently, vertices located near articulated regions can be influenced by

multiple joints at the same time, allowing smooth transitions between adjacent body parts.

For explanatory purposes, this process can be expressed through the following compact notation:

$$\text{SMPL}_{\text{model}} = \mathbf{W} * \mathbf{G}(\boldsymbol{\theta}) * (\mathbf{T} + \mathcal{S} * \boldsymbol{\beta} + \mathbf{R}(\boldsymbol{\theta}) * \mathbf{P})$$

This expression should be interpreted as a schematic representation of the model rather than as the exact vertex-wise skinning equation. In this notation, $\mathbf{T}$ denotes the mean template shape, while $\mathcal{S} * \boldsymbol{\beta}$ represents the shape-dependent displacement of the mesh vertices from the mean body shape. The term $\mathbf{R}(\boldsymbol{\theta}) * \mathbf{P}$ represents the pose-dependent corrective displacement, which accounts for non-rigid deformations associated with articulation and corrects skinning artifacts. After these two blend shape contributions have been added to the template, the resulting corrected mesh is articulated through the skeletal transformations $\mathbf{G}(\boldsymbol{\theta})$. The influence of these transformations on each vertex is controlled by the blend weight matrix $\mathbf{W}$.

The model parameters are learned from registered 3D body scans using two complementary datasets. First, the pose-dependent components are estimated from a multi-pose dataset composed of 1786 registrations of 40 subjects in different body poses. This dataset is used to learn the pose blend shapes $\mathbf{P}$, the blend weights $\mathbf{W}$, and the joint regressor $\mathcal{J}$, which predicts joint locations from the shaped body mesh. These parameters are optimized so that the posed SMPL mesh accurately reconstructs the registered training meshes.

The shape space is learned separately from a multi-shape dataset derived from registrations to the CAESAR dataset, containing 1700 male and 2100 female registrations. Before learning the shape model, the registrations are pose-normalized in order to remove pose-dependent effects and isolate inter-subject body shape variability. Principal Component Analysis is then applied to the pose-normalized meshes to obtain the mean template shape $\bar{\mathbf{T}}$ and the shape blend shapes $\mathcal{S}$.

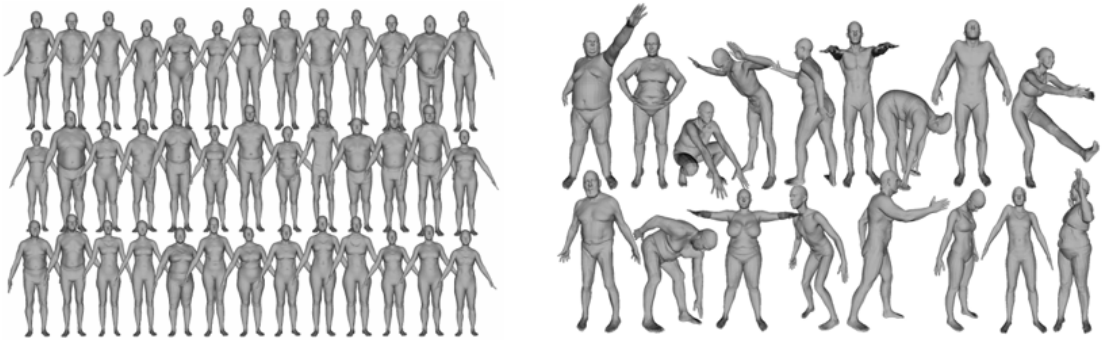


Figure 3.12: Examples from the datasets used for SMPL training: multi-shape dataset on the left and multi-pose dataset on the right. Reproduced from [46].

3.6.2 Subject-specific model calibration

For gait analysis purposes, BRIDGE relies on a generic lower-limb model composed of pelvis, thighs, shanks, and feet, connected by anatomical joints located at the hip, knee, and ankle. The aim of the static calibration procedure is to adapt the mean lower-limb shape of the SMPL model to the morphology and posture of the subject, obtaining a subject-specific three-dimensional representation.

This subject-specific model can be expressed as:

$$\text{SMPL}_{\text{subject-specific}} = \mathbf{W} * \mathbf{G}(\boldsymbol{\theta}_{ss}) * (c_{ss} * \mathbf{T} + \mathbf{S} * \boldsymbol{\beta}_{ss} + \mathbf{R}(\boldsymbol{\theta}_{ss}) * \mathbf{P})$$

where c_{ss} is the scaling factor used to adapt the size of the mean lower-limb template $\mathbf{T}$ to the subject-specific static posture. The model pose is adjusted through the subject-specific pose parameters $\boldsymbol{\theta}_{ss}$, while shape-dependent and pose-dependent displacement terms allow the model surface to better match the observed subject data. In this way, the generic SMPL template is transformed into a static subject-specific model. Since the SMPL mesh has a fixed topology, anatomical joint centers can then be estimated as weighted combinations of the surrounding mesh vertices.

To generate the static three-dimensional representation of the subject, four RGB-D acquisitions were recorded while the participant stood upright: one frontal view, one posterior view, and two lateral views, corresponding to the right and left sides. For each lower limb, three views were used for subject-specific model reconstruction, namely the frontal view, the posterior view, and the corresponding lateral view. For each RGB image, the subject was segmented following the procedure described by Balta et al. [30], and the lower limbs were isolated from the resulting segmentation masks. The depth pixels belonging to the lower limbs were then converted into three-dimensional coordinates, producing point cloud representations of the subject's lower-limb geometry.

Foot identification was performed using a color filter in the RGB images. After identifying the foot regions, the corresponding masks were projected onto the depth images to extract the depth values associated with the feet. These depth data were then converted into three-dimensional coordinates, generating separate foot point clouds.

The static point clouds obtained from the selected views were combined to create a three-dimensional static posture of each lower limb (Figure 3.13). This merging step was performed by aligning a set of common anatomical points automatically identified across the different views. For each side, four reference points were used: toe (T), heel (H), and two proximal points located around the transition between the pelvis and the upper thigh (M1, M2). The right and left lower limbs were processed separately during this step.

After the static posture had been reconstructed, the centroid of the generic SMPL

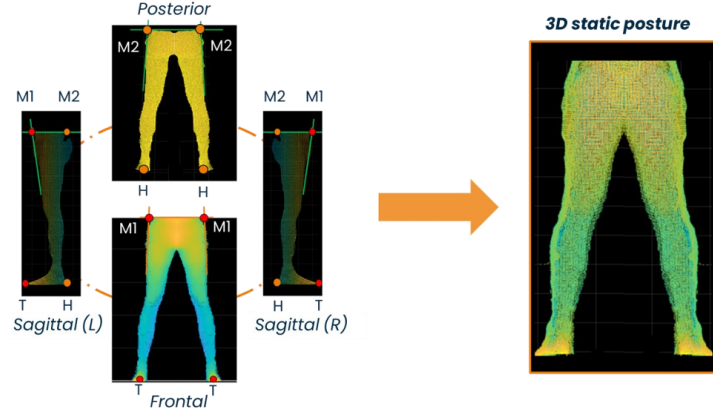


Figure 3.13: Creation of the 3D subject-specific static posture from frontal, posterior, right lateral, and left lateral views by aligning four common anatomical points. Reproduced from [4].

template and the centroid of the subject-specific static point cloud were aligned. The scaling factor c_{ss} was then computed as the ratio between the size of the reconstructed 3D static posture and the size of the generic lower-limb template:

$$c_{ss} = \frac{\text{3D Static Posture Size}}{\text{Template Size}}.$$

This factor was used to scale the mean lower-limb template to the subject's dimensions before fitting the articulated model. Finally, an initial estimate of the lower-limb pose was obtained by fitting the articulated SMPL model to the reconstructed static point cloud through an AICP procedure.

3.6.3 AICP-based fitting procedure

The following description explains the AICP implementation adopted in BRIDGE, following the methodological framework reported by Balta et al. [4]. AICP can be interpreted as an extension of the traditional ICP algorithm designed for the registration of articulated structures. Unlike standard ICP, which treats the object to be aligned as a single rigid body, AICP accounts for the presence of rigid segments connected by joints and therefore allows the pose of the model to be updated while preserving the kinematic relationships between adjacent segments [23].

In BRIDGE, the lower-limb model is considered as an open kinematic chain composed of rigid segments, corresponding to pelvis, thigh, shank, and foot, connected by the hip, knee, and ankle joints. Each joint, modeled as a sphere, is described by a set of pose parameters representing the rotation of the distal

segment with respect to its proximal parent segment. In particular, the pose of the lower-limb model is described by the hip, knee, and ankle pose vectors:

$$\boldsymbol{\theta}_{hip} = \{\alpha_{hip}, \beta_{hip}, \gamma_{hip}\},$$

$$\boldsymbol{\theta}_{knee} = \{\alpha_{knee}, \beta_{knee}, \gamma_{knee}\},$$

$$\boldsymbol{\theta}_{ankle} = \{\alpha_{ankle}, \beta_{ankle}, \gamma_{ankle}\}.$$

For the hip and knee joints, α , β , and γ represent flexion/extension, abduction/adduction, and internal/external rotation, respectively. For the ankle joint, the three components describe dorsi/plantar flexion, internal/external rotation, and prono-supination, respectively. The complete subject-specific pose vector is therefore defined as:

$$\boldsymbol{\theta}_{ss} = \{\boldsymbol{\theta}_{hip}, \boldsymbol{\theta}_{knee}, \boldsymbol{\theta}_{ankle}\}.$$

The absolute orientation of each segment in the world reference frame is obtained by concatenating the transformations along the kinematic chain, starting from the root segment. In the adopted implementation, the root of the model is placed at the hip, and the alignment is performed using a top-down strategy along the articulated chain.

The main principle of AICP is to select, at each iteration, a specific joint and split the articulated model into two branches: a base branch, which contains the root, and an outer branch, which contains the segments distal to the selected joint. Only the outer branch is transformed, whereas the base branch is kept fixed. For each point of the outer branch, the closest point in the subject-specific static point cloud is identified, and the rigid transformation associated with the selected joint is estimated by minimizing the distance between the model and the observed data. In general form, the alignment problem can be expressed as:

$$E_o(\boldsymbol{\theta}_i) = \sum_{p_k \in \mathbf{M}_o^i} \sum_{j=1}^{N_{p_k}} \min \left\| \mathbf{G}_k^W(\boldsymbol{\theta}_i) \mathbf{m}_j^{p_k} - \mathbf{d}_s \right\|^2,$$

where $\mathbf{M}_o^i$ denotes the outer branch obtained by splitting the model at joint j_i , $\mathbf{m}_j^{p_k}$ is a point belonging to segment p_k , $\mathbf{d}_s$ represents the closest point in the subject-specific static point cloud, and $\mathbf{G}_k^W(\boldsymbol{\theta}_i)$ is the transformation that optimally aligns $\mathbf{M}_o^i$ and $\mathbf{d}_s$.

During static calibration, the joints are selected cyclically. First, the hip joint is selected; since it corresponds to the root of the kinematic chain, the resulting transformation is applied to the entire lower-limb model. The knee joint is then

selected, and the model is split so that the transformation is applied only to the distal branch, composed of the shank and foot. This procedure is repeated until convergence is reached, namely when the alignment error between the model and the point cloud is no longer substantially reduced.

In the considered protocol, the hip, knee, and ankle joints are modeled as spherical joints with three degrees of freedom. However, during static calibration, hip and knee internal/external rotations are constrained, since static acquisitions are performed with the subjects in an upright standing posture, with parallel feet and knees extended as much as possible. The optimization search limits are also defined according to physiological constraints, in order to avoid non-admissible model configurations. To reduce the computational cost of the procedure, the foot pose is not estimated through AICP; instead, it is determined separately from the previously identified heel and toe positions. More specifically, ankle dorsi/plantar flexion and foot internal/external rotation are computed using trigonometric relationships.

At the end of the AICP procedure, the pose parameter θ_{ss} describes the initial articulated configuration of the subject-specific SMPL model with respect to the reconstructed static posture. This estimate provides an initial articulated registration of the model to the observed data.

The remaining discrepancy between the mean shape ($\mathbf{T}$) and the 3D static posture ($\mathbf{P}_{static}$) is then reduced by estimating the subject-specific shape parameters β_{ss} :

$$\beta_{ss} = \text{inv}(\mathbf{S}) * [\mathbf{P}_{static} - (\mathbf{W} * \mathbf{G}(\theta_{ss}) * (c_{ss} * \mathbf{T} + \mathbf{R}(\theta_{ss}) * \mathbf{P}))].$$

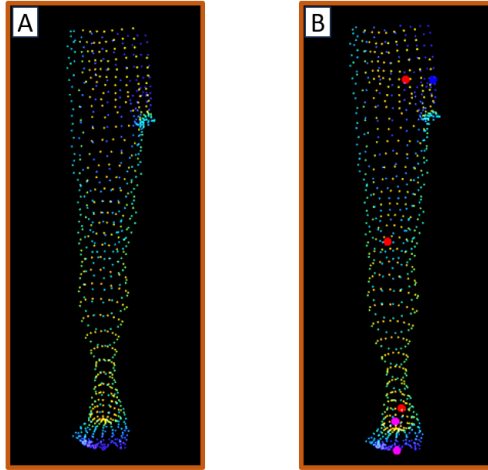


Figure 3.14: Three-dimensional subject-specific lower-limb model obtained during static calibration. (A) Reconstructed lower-limb model. (B) Model with anatomical points used for AGM: estimated joint centers (red), heel and toe (magenta), and the mid-hip reference point (blue).

3.6.4 Estimation of joint center and keypoint trajectories

Once the subject-specific SMPL lower-limb model used in BRIDGE had been obtained from static calibration, joint center trajectories were estimated by fitting this model to the dynamic point clouds reconstructed during the gait cycle. For each frame of the dynamic trial, RGB-D images were processed to isolate the subject's lower limbs and convert the corresponding depth pixels into three-dimensional coordinates. In this way, a dynamic point cloud was generated for each time instant and was subsequently used for fitting the subject-specific SMPL model [4].

Since the two lower limbs may overlap in the sagittal view during gait, the foreground limb was separated from the background limb before model alignment. For this purpose, depth information was used: the depth values of the segmented region were analysed and classified using Otsu's method, which automatically identifies an optimal threshold to separate pixel classes with different distributions [47]. This procedure allowed the limb closer to the camera to be distinguished from the farther one, reducing the risk that the point cloud used for fitting included points belonging to the contralateral limb.

A further critical aspect was the possible overlap between the hand and the thigh during arm swing. When the hand partially interfered with the thigh profile, the corresponding points could introduce local minima in the fitting procedure. To limit this effect, regions not belonging to the lower limb were removed when they could be identified. Conversely, when the hand was completely superimposed on the thigh and could not be reliably separated, the region was retained in the dynamic point cloud.

Before applying AICP, the subject-specific SMPL model was initialized with respect to the dynamic point cloud. In particular, the model size, controlled through the scaling factor c_{ss} , was adapted frame by frame to the size of the observed lower limb, in order to compensate for apparent variations due to soft-tissue artifacts or small changes in the distance between the subject and the camera. The model was also translated to provide an initial configuration consistent with the dynamic point cloud. For the first frame of the gait cycle, the model was initialized in a neutral configuration, whereas for the following frames, the pose estimated in the previous frame was used as the initial condition. This strategy reduced computational time and promoted temporal continuity of the estimate. To avoid non-physiological configurations during optimization, joint parameters were also constrained within limits defined according to biomechanical criteria, as reported in Table 3.6.

Dynamic alignment was then performed using AICP, following a top-down strategy along the kinematic chain. First, the hip joint was selected, and the model was aligned to the dynamic point cloud through a rigid transformation applied to the entire lower limb. The knee joint was then selected, and the transformation was

Table 3.6: Physiological joint limits used during the AICP optimization.

Joint	Movement	Search range
Hip	flexion/extension	$[-\pi/4; \pi/4]$
Hip	Abduction/adduction	0
Hip	Internal/external rotation	0
Knee	Flexion/extension	$[-\pi/3; \pi/3]$
Knee	Abduction/adduction	0
Knee	Internal/external rotation	0

applied to the distal branch composed of the shank and foot. In the original protocol, AICP optimization is mainly limited to hip and knee flexion/extension movements, since the single sagittal view does not provide sufficiently robust information to reliably estimate out-of-plane components, such as abduction/adduction and internal/external rotation [4].

The foot pose, described by the parameter θ_{ankle} , was not directly estimated through AICP. Similarly to the static calibration procedure, it was derived from the positions of heel and toe, which were identified in the RGB images consistently with the PiG marker set definition and subsequently mapped into three-dimensional space using the corresponding depth values. In particular, dorsi/plantar flexion and internal/external rotation of the foot were estimated using trigonometric relationships.

The iterative process continued until convergence was reached, defined either by the reduction of the alignment error between consecutive iterations or by the achievement of a maximum number of iterations. At the end of the fitting procedure, the positions of the hip, knee, and ankle joint centers were estimated by applying the joint regression matrix to the aligned SMPL model. In this way, three-dimensional trajectories of the hip, knee, and ankle joint centers were obtained for each frame of the gait cycle.

These trajectories were integrated with the heel and toe anatomical points, which had already been identified for the estimation of θ_{ankle} . Moreover, since BRIDGE analysed one lower limb at a time, the midpoint between the two hip joint centers, required for the following stages of the pipeline, could not be directly computed. It was therefore defined as the most medial point of the pelvis on the frontal plane of the model, while keeping the other two coordinates equal to those of the already estimated hip joint center.

Overall, for each frame of the gait cycle, six anatomical points were extracted from the aligned SMPL model: HJC, KJC, AJC, heel, toe, and the mid-hip reference point. For simplicity, these points are hereafter collectively referred to as keypoints. In BRIDGE, these trajectories were used to compute sagittal

joint kinematics, whereas in BRIDGE+AGM they constituted the input for the subsequent LSTM-based AGM step. A representative example of the keypoints identified during a left gait cycle is shown in Figure 3.15.

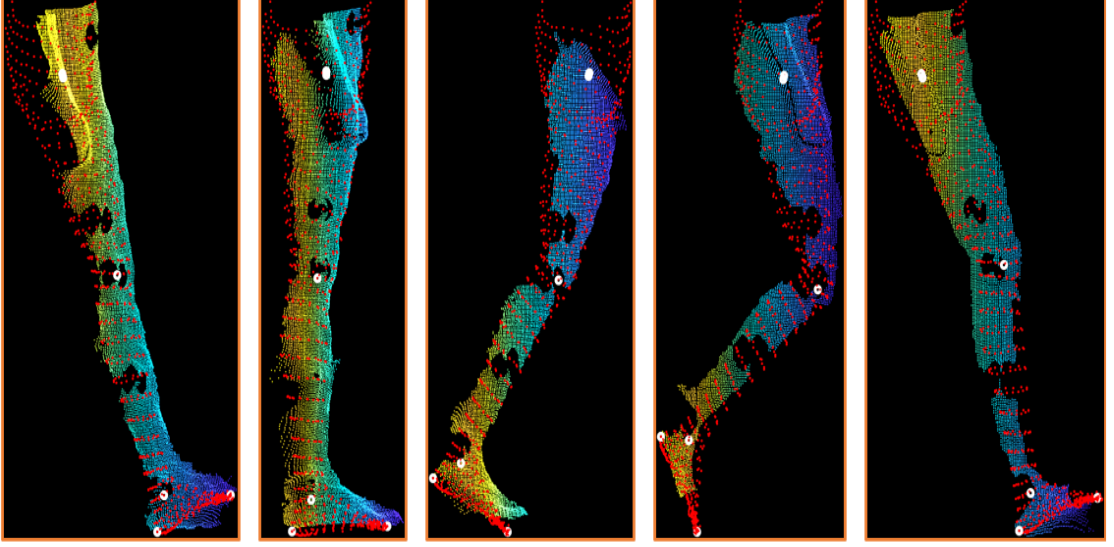


Figure 3.15: Representative example of keypoint identification from the aligned SMPL model during a left gait cycle.

3.6.5 Estimation of joint kinematics

In the BRIDGE pipeline, lower-limb joint kinematics are estimated directly from the joint center trajectories obtained after fitting the subject-specific SMPL model to the dynamic point clouds. In this approach, joint angles are not computed from segment reference frames reconstructed from anatomical markers, but through trigonometric calculations based on anatomical segments defined by the estimated joint centers. Therefore, the analysis is limited to sagittal-plane kinematics.

The ankle dorsi/plantar flexion angle is calculated from the relative orientation between the foot and the shank. The shank direction is defined by the vector connecting the AJC to the KJC, whereas the foot direction is obtained from the color-based foot orientation estimation and is centered at the AJC. The ankle angle is then computed as the angle between these two directions.

The knee flexion/extension angle is obtained from the relative orientation between the shank and thigh segments. The shank segment is defined by the vector connecting AJC and KJC, while the thigh segment is defined by the vector connecting KJC and HJC.

Finally, hip flexion/extension is estimated from the inclination of the thigh segment with respect to the horizontal axis.

3.6.6 LSTM-based marker augmentation

The BRIDGE pipeline provides a limited set of anatomical keypoints, namely the HJC, KJC, AJC, the heel, the toe, and the mid-hip. Although these points are sufficient to estimate sagittal-plane joint angles through trigonometric relationships, they do not provide enough anatomical information to reconstruct segment reference frames according to the PiG convention. For this reason, an AGM-based version of the pipeline, referred to as BRIDGE+AGM, was evaluated.

The AGM approach was developed starting from the training framework made available by Falisse et al. [44]. In particular, the implementation was adapted from the body marker enhancer model, which is the component mainly focused on the lower limbs. Because right and left gait cycles were processed separately, two side-specific LSTM models were trained: a *Body Left* model and a *Body Right* model. The two networks were designed to predict anatomical marker trajectories compatible with the *PiG-like* workflow adopted in this thesis, using as input the keypoints previously extracted from the fitted subject-specific SMPL model.

For each side-specific model, the input included the subject's height, body mass, and the side-specific keypoints. The output consisted of the pelvic markers and the anatomical markers required to define the thigh and shank segments of the analysed limb. The input and output variables used for the two models are summarized in Table 3.7. A schematic representation of the *Body Left* and *Body Right* networks is shown in Figure 3.16.

Table 3.7: Input and output variables of the side-specific LSTM models used in the BRIDGE+AGM pipeline.

Model	Inputs	Outputs
Body Left	Height, mass, mid-hip,	LASI, RASI, LPSI, RPSI,
	LHJC, LKJC, LAJC, LTOE, LHEEL	LTHI cluster, LKNE, LTIB cluster, LANK
Body Right	Height, mass, mid-hip,	LASI, RASI, LPSI, RPSI,
	RHJC, RKJC, RAJC, RTOE, RHEEL	RTHI cluster, RKNE, RTIB cluster, RANK

For clarity, the schematic representation in Figure 3.16 does not show all markers belonging to the thigh and tibia clusters. Instead, only the marker that best represents the corresponding cluster position within the adopted PiG marker set is displayed. Consistently with the marker placement described for the reference MB model, the representative thigh and tibia markers are located at the lower third of the segment for the left side and at the upper third of the segment for the right side.

The training data used in this thesis were obtained from the datasets made available by Falisse et al. [44]. In their work, expert-processed MB motion capture datasets were processed in OpenSim and AddBiomechanics to obtain scaled musculoskeletal models and corresponding kinematic coordinate files. Virtual markers representing both video keypoints and anatomical markers were then added to the scaled models, and their three-dimensional trajectories were extracted to generate the synthetic training data. The original dataset was further augmented by uniformly scaling each OpenSim model, thereby simulating subjects with different body sizes.

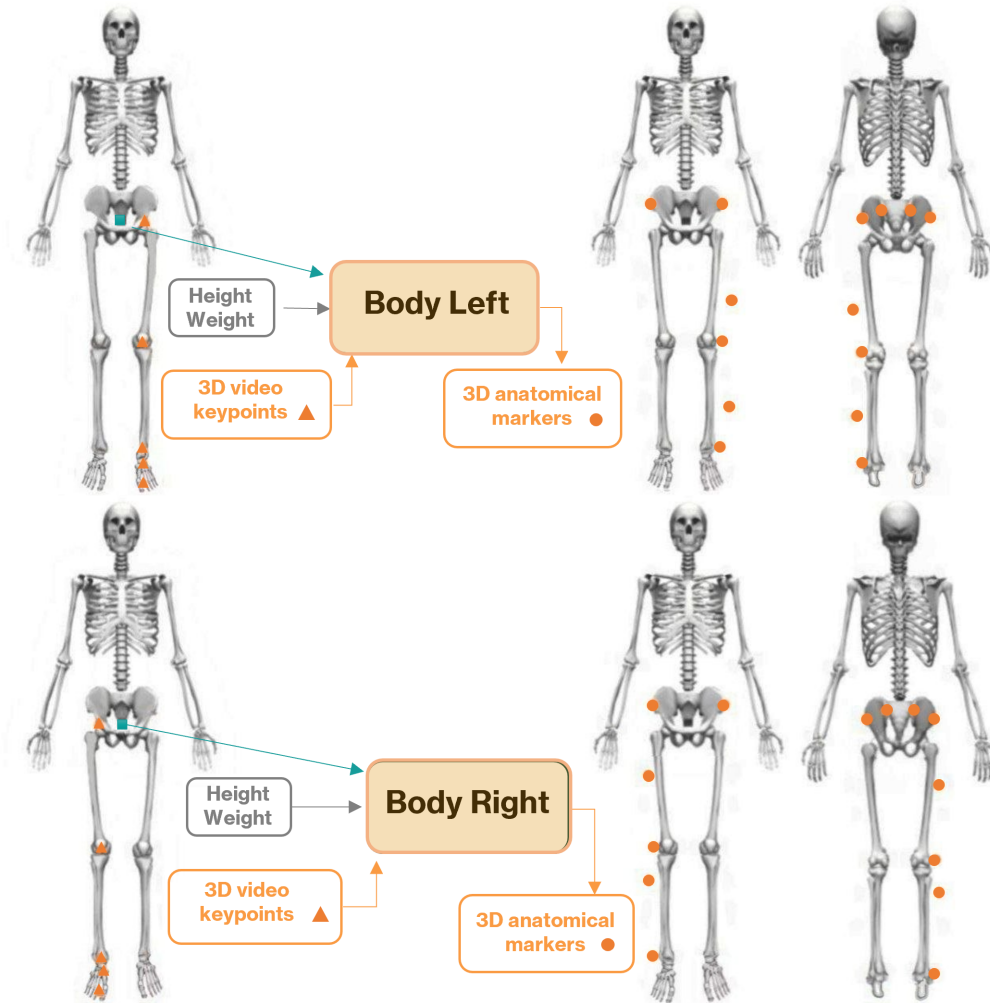


Figure 3.16: Schematic representation of the side-specific LSTM models used in the BRIDGE+AGM pipeline. For graphical clarity, only the representative thigh and tibia markers are shown instead of the complete marker clusters. Adapted from [44].

In the present work, this dataset was not used in its entirety. Only the subsets containing overground walking, treadmill walking and running trials were selected for training. This choice was made to focus the learning process on locomotor tasks that were biomechanically closer to the movements analysed in this thesis, while avoiding the introduction of variability associated with unrelated motor tasks. At the same time, restricting the training data reduced the computational cost of the training procedure.

Each selected dataset was divided into three subsets: 80% of the data were used for training, 10% for validation and 10% for testing. Both the *Body Left* and *Body Right* models were first trained from the selected data and subsequently refined through two additional fine-tuning stages. A weighted mean squared error was used as the loss function, so that the contribution of the different predicted markers could be balanced during optimization. Each training stage was run for a maximum of 10 epochs. Early stopping was applied by monitoring the validation loss: if no improvement in validation loss was observed for seven consecutive epochs, the training was stopped. For the fine-tuning stages, the checkpoint corresponding to the lowest validation loss obtained in the previous stage was used as initialization.

The resulting side-specific LSTM models therefore allowed the sparse BRIDGE keypoint trajectories to be transformed into a richer set of anatomical marker trajectories. Before these trajectories were used for *PiG-like* joint angle calculation, a post-processing refinement was applied to reduce systematic inconsistencies in the predicted marker positions.

For each frame, the vertical coordinates of LASI and RASI were first aligned to a common level, defined as the average of their predicted vertical positions. The posterior pelvic markers were then constrained by assigning LPSI and RPSI to a vertical level 3 cm higher than the corrected ASIS level. After this initial pelvic alignment, an additional refinement was applied to account for the expected vertical oscillation of the pelvis during the gait cycle. For left gait cycles, the vertical coordinates of LASI and LPSI were increased by 1 cm between 0% and 30% of the gait cycle and decreased by 1 cm from 31% to 100%. The same criterion was applied symmetrically to right gait cycles, using the corresponding right pelvic markers.

Additional constraints were imposed on the markers used to define the thigh and shank reference frames. The KNE and ANK markers were vertically aligned with the corresponding estimated joint centers, namely the KJC and AJC. The THI marker was projected onto the line connecting the HJC and KJC, whereas the TIB marker was projected onto the line connecting the KJC and AJC. Finally, a consistency check was performed on the pelvic marker configuration by computing, for each frame, the area enclosed by LASI, RASI, LPSI and RPSI after projection onto the horizontal plane. Frames in which this area was lower than a predefined threshold were considered unreliable, set to NaN and subsequently reconstructed

by interpolation.

The refined marker trajectories were then used as input for the subsequent *PiG-like* workflow. In this step, local segment reference frames were reconstructed, as described in subsection 3.4.3, and used to compute sagittal and frontal lower-limb joint angles.

3.7 Data processing and temporal normalization

For each participant, ten walking trials were acquired, consisting of five right and five left gait trials. For the subsequent analyses, the first three valid trials for each side were selected. A trial was considered valid when the data obtained from all three acquisition systems were available and did not contain damaged files. For each selected trial, only the central gait cycle was considered in order to reduce the influence of gait initiation and termination phases. The identification of the gait cycle differed according to the acquisition system. For the RGB-D-based data used in BRIDGE, the start and end frames of the gait cycle were identified manually through visual inspection of the acquired data. Conversely, for the MB and OpenCap data, gait cycle identification was performed using an automatic criterion based on the knee flexion/extension curve. In healthy subjects, this signal provides a clearly identifiable peak during the swing phase, which can be used to locate consecutive gait cycles (Figure 3.17).

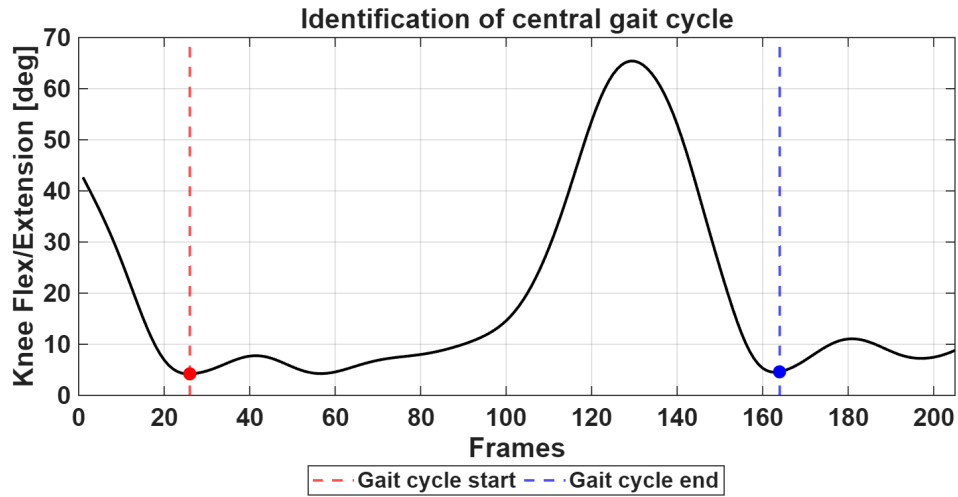


Figure 3.17: Example of gait cycle identification based on the knee flexion/extension angle obtained from the MB system. The selected gait cycle was defined between two consecutive minima following the knee flexion peak observed during the swing phase, indicated by the red and blue markers. In this example, the first swing-phase peak is not shown because the trial had been previously trimmed.

The joint kinematic signals extracted from the MB reference system included hip, knee and ankle flexion/extension, hip abduction/adduction, knee varus/valgus and ankle internal/external rotation. However, not all the ML pipelines considered in this thesis provided the same set of kinematic components. The standard OpenCap output included sagittal-plane kinematics for the hip, knee and ankle, together with hip abduction/adduction. BRIDGE provided only sagittal-plane flexion/extension angles for the hip, knee and ankle. Conversely, the OpenCap *PiG-like* and BRIDGE+AGM pipelines allowed the estimation of all the kinematic components considered in the comparison. Before comparing the joint angle curves obtained from the different pipelines, the sign conventions were harmonized according to the kinematic curve conventions reported by Benedetti et al. [40]. These conventions are summarized in Table 3.8.

Table 3.8: Sign conventions of the kinematic curves used for the comparison of lower-limb joint angles.

Joint	Positive value	Negative value
Hip	Flexion	Extension
	Abduction	Adduction
	Internal rotation	External rotation
Knee	Flexion	Extension
	Valgus / abduction	Varus / adduction
	Internal rotation	External rotation
Ankle	Dorsiflexion	Plantarflexion
	Internal rotation	External rotation

This step was necessary because the native outputs of the different processing pipelines do not necessarily follow the same sign convention or anatomical interpretation for each angular component. After convention harmonization, all joint angle signals were filtered using a second-order zero-lag Butterworth low-pass filter with a cut-off frequency of 7 Hz [48]. The filtered signals were then temporally normalized over the gait cycle. Each gait cycle was resampled from 0% to 100% of the gait cycle, obtaining a normalized waveform composed of 100 points. This normalization allowed the comparison of joint angle curves across trials, subjects and acquisition systems independently of the sampling rate. For each normalized curve, the mean value was also computed and subtracted from the entire waveform to obtain the corresponding mean-removed curve. Subsequent analyses were therefore performed

in parallel on both the original normalized curves and the mean-removed curves.

Discrete kinematic parameters were extracted from the normalized waveforms according to Benedetti et al. [40]. Since no specific parameter is defined for ankle internal/external rotation, only its range of motion was considered. The complete list of selected parameters and their biomechanical meaning is reported in Table 3.9. Representative MB kinematic curves with the extracted parameters are shown in Figure 3.18.

Table 3.9: Discrete kinematic parameters extracted from the time-normalized joint angle curves.

Joint	Parameter	Description
Hip	H3	Maximum hip extension during stance phase
Hip	H6	Total sagittal plane excursion
Hip	H7	Total coronal plane excursion
Hip	H8	Maximum hip adduction during stance phase
Hip	H9	Maximum hip abduction during swing phase
Knee	K1	Knee flexion at heel strike
Knee	K2	Maximum knee flexion during loading response
Knee	K3	Maximum knee extension during stance phase
Knee	K5	Maximum knee flexion during swing phase
Knee	K6	Total sagittal plane excursion
Knee	K7	Total coronal plane excursion
Knee	K8	Maximum knee adduction (varus) during stance phase
Knee	K9	Maximum knee adduction (varus) during swing phase
Ankle	A3	Maximum ankle dorsiflexion during stance phase
Ankle	A5	Maximum ankle plantarflexion during swing phase
Ankle	A6	Total sagittal plane excursion
Ankle	A10	Total transverse plane excursion

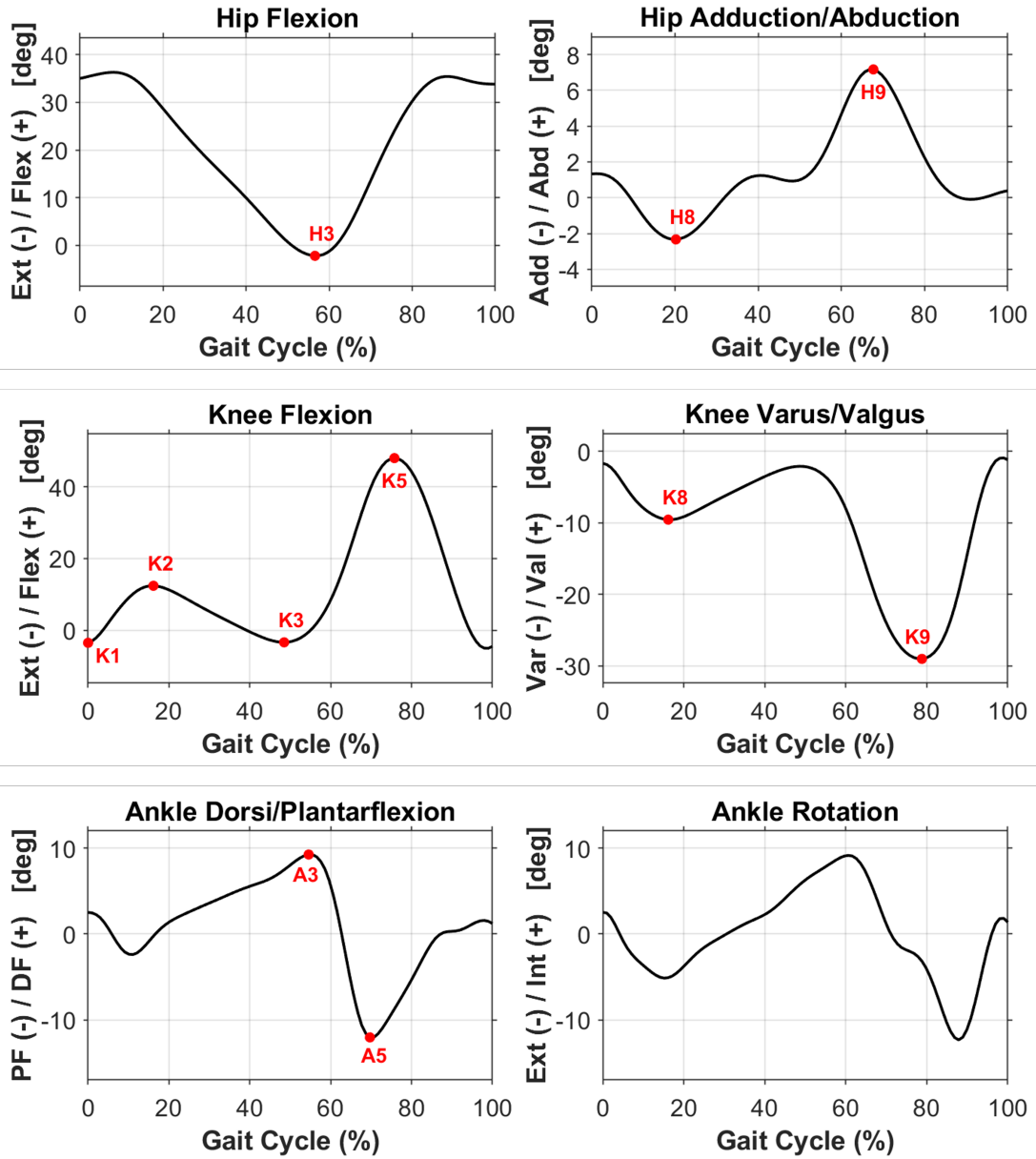


Figure 3.18: Representative MB lower-limb joint angle curves normalized from 0 to 100% of the gait cycle. Red markers indicate the discrete kinematic parameters extracted from the hip, knee and ankle curves.

3.8 Performance evaluation and statistical analysis

The performance of each ML pipeline was evaluated by comparing its outputs with the corresponding MB results, which were used as the reference. The comparison was performed at two levels: waveform-based analysis and discrete-parameter analysis. Both analyses were conducted on the original time-normalized curves and after offset removal. In the offset-removed analysis, the mean value over the gait cycle was subtracted from both the ML and the corresponding MB waveforms before computing waveform errors and extracting discrete parameters.

For the waveform-based comparison, the root mean square error (RMSE) was computed between each ML joint angle curve and the corresponding MB curve. This metric was calculated separately for each trial, subject, and kinematic component, using all points of the time-normalized gait cycle:

$$\text{RMSE}_{c,s,t} = \sqrt{\frac{1}{N} \sum_{i=1}^N [\theta_{\text{ML},c,s,t}(i) - \theta_{\text{MB},c,s,t}(i)]^2}$$

where N is the number of points of the normalized waveform, $\theta_{\text{ML},c,s,t}(i)$ and $\theta_{\text{MB},c,s,t}(i)$ are the joint angle values obtained from the ML pipeline and from the MB reference system, respectively, at the i -th point of the gait cycle. The indices c , s , and t indicate the kinematic component, subject, and trial, respectively.

For each discrete parameter extracted from the normalized kinematic curves, the absolute error (AE) was computed as:

$$\text{AE}_{p,s,t} = |p_{\text{ML},s,t} - p_{\text{MB},s,t}|$$

where $p_{\text{ML},s,t}$ and $p_{\text{MB},s,t}$ denote the value of a given discrete parameter p estimated from the ML and MB data, respectively.

For each ML pipeline, trial-level errors were pooled across all valid trials and subjects, with right and left gait cycles analysed together as repeated valid trials because only healthy participants were included. Separate side-specific analyses would instead be required in pathological populations characterized by asymmetric gait patterns. The pooled RMSE and AE values were used to describe waveform and discrete-parameter errors, respectively, and were represented using boxplots and summarized through the median and interquartile range.

Chapter 4

Results

This chapter presents the comparison between the ML pipelines and the MB reference system. Results are first shown through representative joint angle waveforms, providing a qualitative overview of the kinematics, and then through a quantitative assessment of waveform-based and discrete gait parameter errors.

Figure 4.1, Figure 4.2 and Figure 4.3 show the hip, knee, and ankle kinematics from the right side of one selected subject across three right-side walking trials. Solid lines represent mean waveforms across trials, while shaded areas indicate one standard deviation. Since the pipelines provide different sets of joint angle components, only the available methods are displayed for each movement.

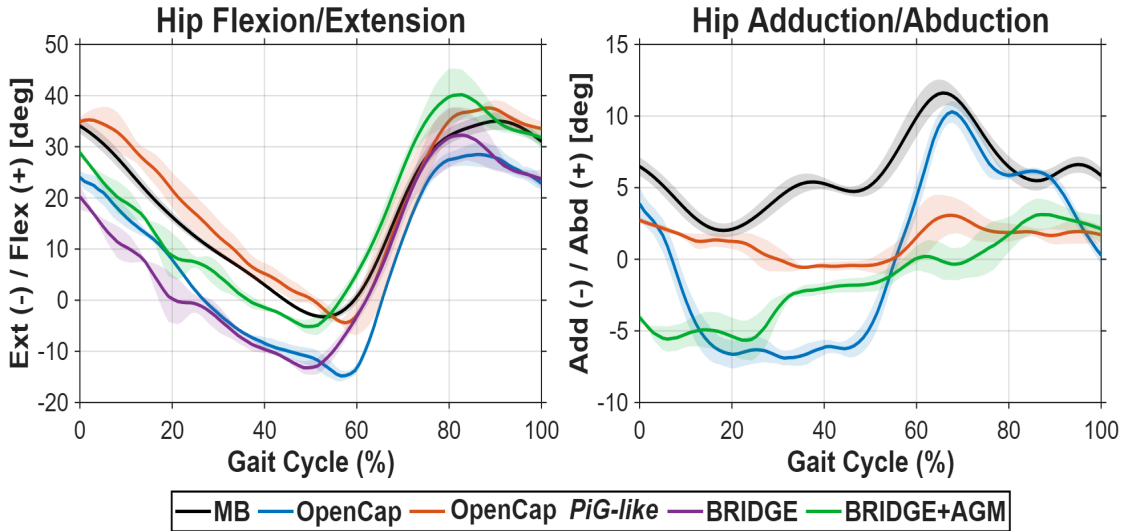


Figure 4.1: Representative hip joint kinematics across the MB reference and ML pipelines.

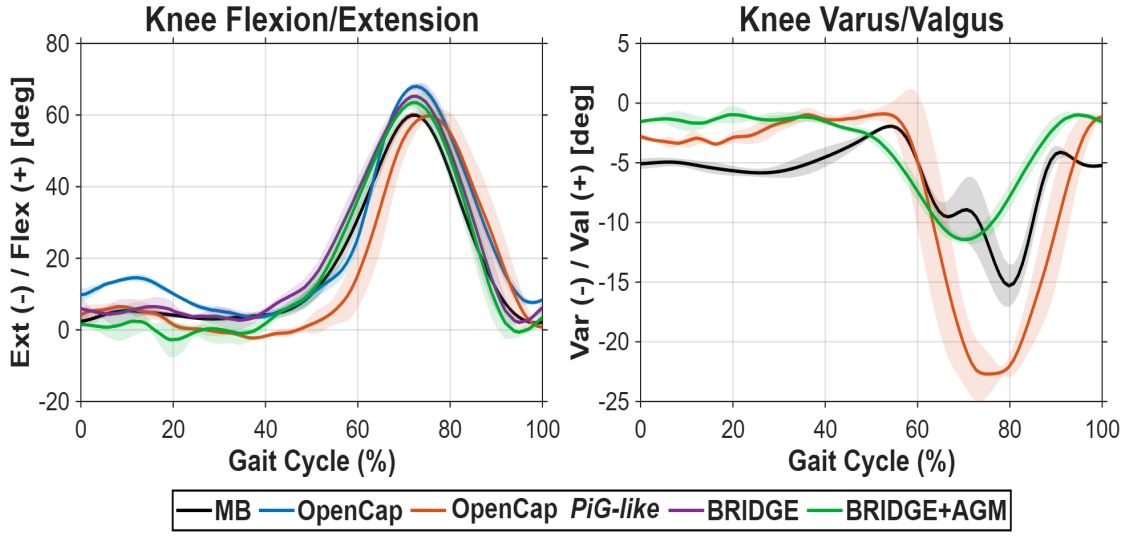


Figure 4.2: Representative knee joint kinematics across the MB reference and ML pipelines.

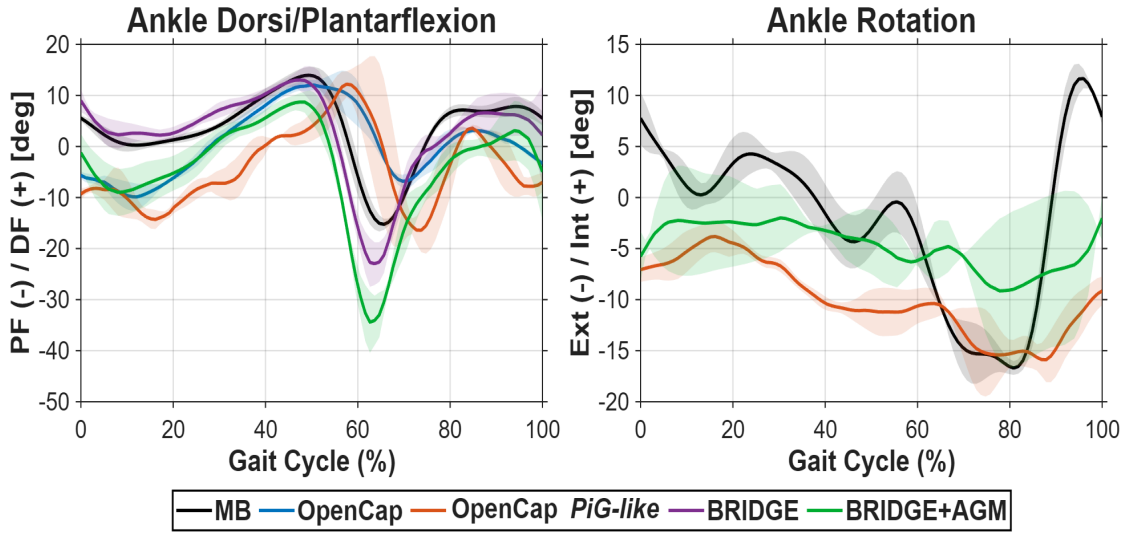


Figure 4.3: Representative ankle joint kinematics across the MB reference and ML pipelines.

The quantitative results are presented in the following sections by separating waveform-based errors from discrete gait parameter errors. For each analysis, boxplots are used to illustrate the distribution of trial-level errors, while the corresponding summary tables report the median, first quartile, and third quartile values.

4.1 Original time-normalized curves

The distribution of RMSE values computed on the original time-normalized joint angle waveforms is shown in Figure 4.4. Since no offset removal was applied, these values include both waveform-shape differences and systematic offsets between the ML pipelines and the MB reference.

Overall, the lowest errors were observed for hip ab/adduction, with median RMSE values below approximately 5° for all available methods. For hip flexion/extension, median values ranged from about 7° to 10° , with slightly lower errors for OpenCap *PiG-like* and BRIDGE. In knee flexion/extension, OpenCap *PiG-like* and BRIDGE+AGM showed lower median errors, around $7-8^\circ$, whereas standard OpenCap and BRIDGE were closer to 10° . For ankle flexion/extension, BRIDGE showed the lowest median RMSE, approximately 7° , while the other pipelines showed median values around $10-12^\circ$. Higher errors were found for ankle rotation, where OpenCap *PiG-like* and BRIDGE+AGM reached median values above 12° and showed the widest distributions.

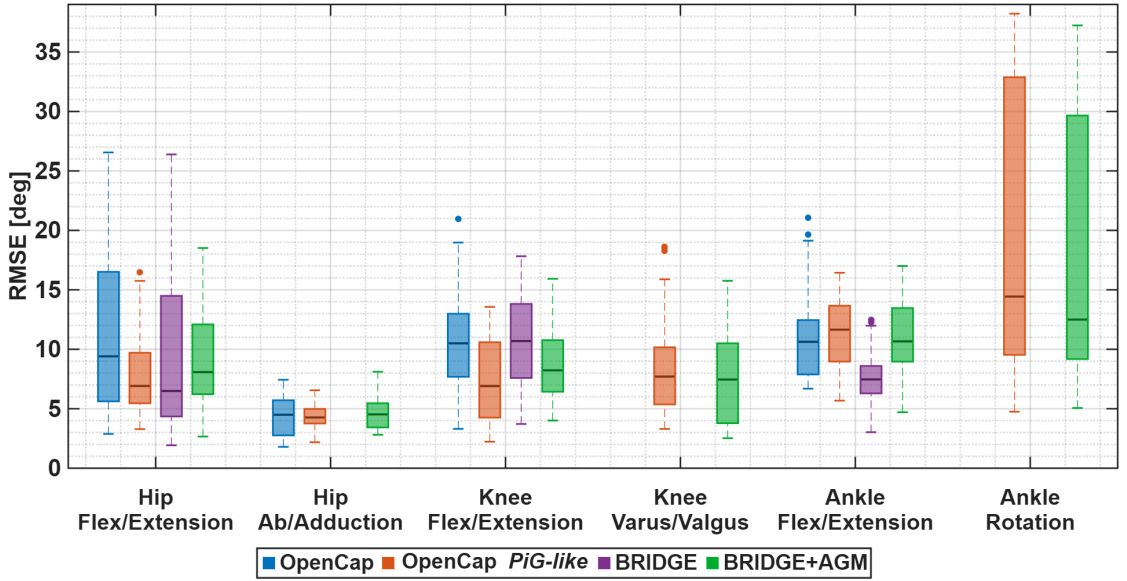


Figure 4.4: Distribution of waveform-based RMSE values.

The distribution of AE values for the hip discrete gait parameters computed from the original time-normalized curves is shown in Figure 4.5. Since BRIDGE provides only sagittal-plane kinematics, it was not available for all hip parameters.

Overall, H6, H8, and H9 showed lower errors than H3. For H3, standard OpenCap showed the highest median AE, around 10° , whereas OpenCap *PiG-like*

and BRIDGE were close to 5° . BRIDGE+AGM showed an intermediate median value, approximately 8° . For H6, all pipelines showed median errors below about 5° , despite isolated high values for BRIDGE and BRIDGE+AGM. For H7–H9, the available methods showed median errors generally between 2° and 8° , with lower values for OpenCap in H9 and for OpenCap *PiG-like* and BRIDGE+AGM in H8.

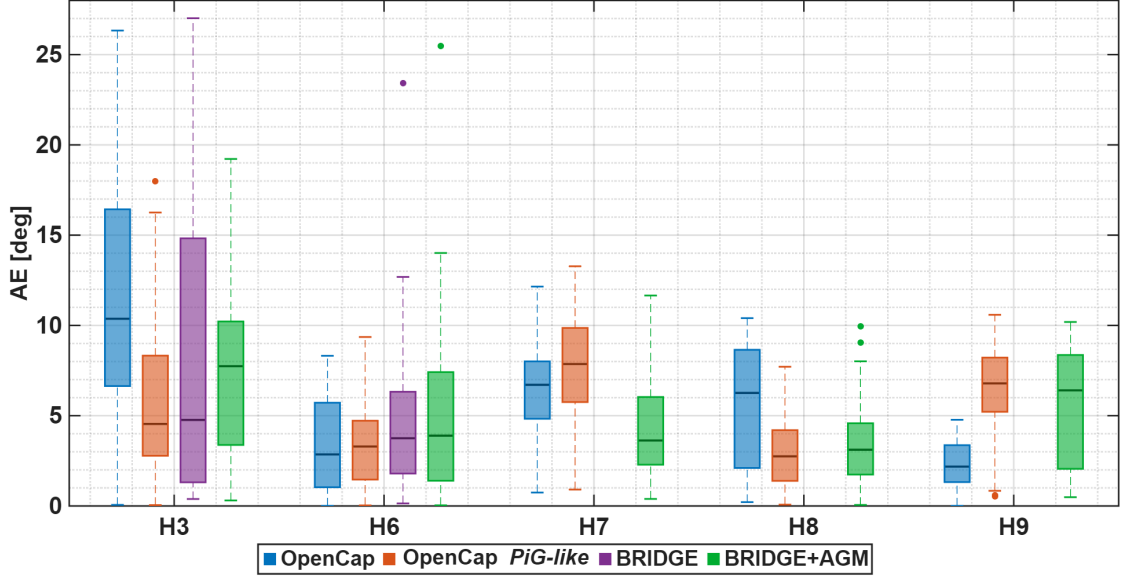


Figure 4.5: Distribution of AE values for hip discrete gait parameters.

The distribution of AE values for the knee discrete gait parameters computed from the original time-normalized curves is shown in Figure 4.6.

Overall, lower errors were observed for K3, K5, K6, and K8, whereas larger errors were found for K1, K2, K7, and K9. For the sagittal-plane parameters, OpenCap *PiG-like* generally showed the lowest median errors, mostly between approximately 3° and 6° . Standard OpenCap showed higher median errors for K1, K2, and K5, ranging from about 10° to 13° , while BRIDGE showed higher errors particularly for K1–K3. BRIDGE+AGM showed intermediate median values, generally around 5° – 7° . Some parameters, particularly K1 and K2, also showed wide interquartile ranges, indicating greater variability across trials.

For frontal-plane knee parameters, only OpenCap *PiG-like* and BRIDGE+AGM were available. Both methods showed low median errors for K8, around 3° – 4° , whereas larger errors were observed for K7 and K9. In particular, OpenCap *PiG-like* showed the highest median AE for K9, above 10° , while BRIDGE+AGM showed a lower median value, around 8° . Isolated high-error values were also observed, most notably for K7 and K9, suggesting that some trials were associated with markedly

larger deviations from the MB reference.

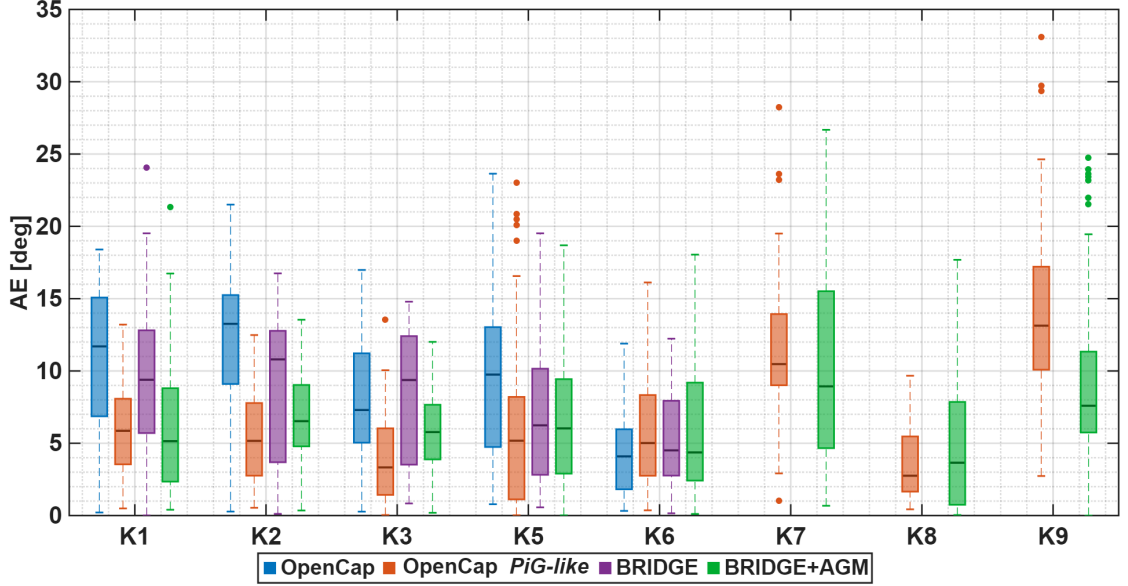


Figure 4.6: Distribution of AE values for knee discrete gait parameters.

The distribution of AE values for the ankle discrete gait parameters computed from the original time-normalized curves is shown in Figure 4.7.

Overall, A3 showed the lowest errors, whereas larger errors and wider distributions were observed for A5, A6, and A10. This indicates that the accuracy of the ankle-related parameters was more variable than that observed for most hip and knee parameters. For A3, all pipelines showed comparable median errors, approximately between 4° and 6° . For A5, OpenCap *PiG-like* and BRIDGE showed lower median errors, around 5° – 6° , while standard OpenCap was closer to 9° . Conversely, BRIDGE+AGM showed the highest median value for this parameter, around 17° , together with marked variability across trials. For A6, standard OpenCap and BRIDGE showed median errors around 6° , whereas OpenCap *PiG-like* and BRIDGE+AGM showed higher median values, approximately 10° – 11° . Isolated high-error values were observed for standard OpenCap. For A10, only OpenCap *PiG-like* and BRIDGE+AGM were available, both showing median errors around 12° and broad distributions, confirming the higher variability associated with this ankle parameter.

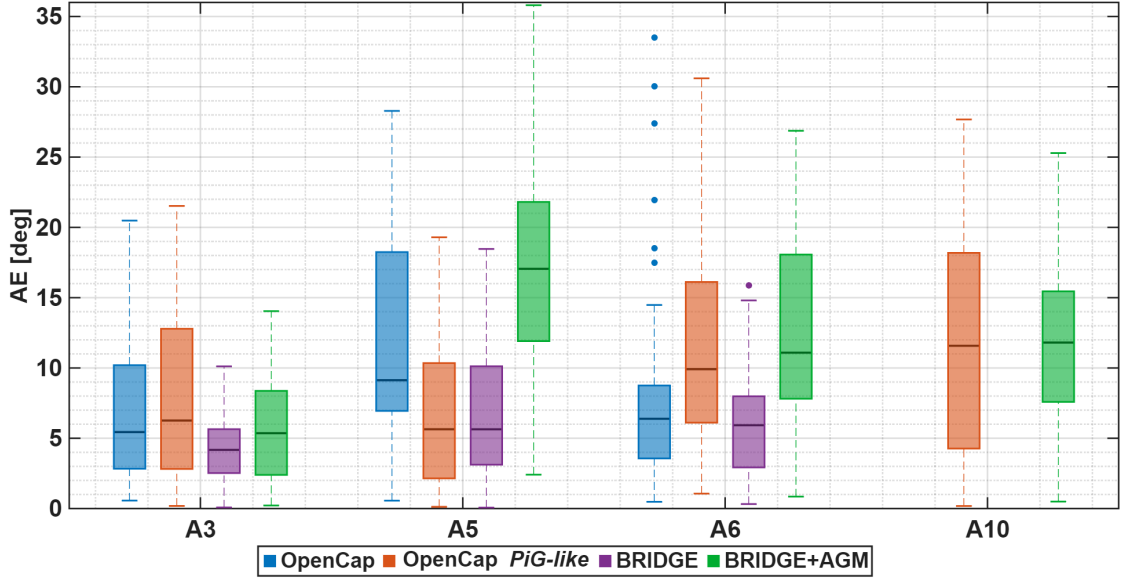


Figure 4.7: Distribution of AE values for ankle discrete gait parameters.

To complement the graphical representation provided by the boxplots, the corresponding descriptive statistics are reported in the following tables. Values are summarized as median, first quartile, and third quartile for each ML pipeline.

Table 4.1: Summary of RMSE values for joint angle waveforms from the original time-normalized curves. Values are reported as median [Q1–Q3].

	OpenCap	OpenCap <i>PiG-like</i>	BRIDGE	BRIDGE+AGM
Hip Flex/Extension	9.4 [5.6–16.5]	6.9 [5.5–9.7]	6.5 [4.3–14.5]	8.1 [6.2–12.1]
Hip Ab/Adduction	4.5 [2.8–5.7]	4.3 [3.8–5.0]	/	4.5 [3.4–5.5]
Knee Flex/Extension	10.5 [7.7–13.0]	6.9 [4.3–10.6]	10.7 [7.6–13.8]	8.2 [6.4–10.8]
Knee Varus/Valgus	/	7.7 [5.4–10.2]	/	7.5 [3.8–10.5]
Ankle Flex/Extension	10.6 [7.9–12.5]	11.7 [9.0–13.7]	7.5 [6.3–8.6]	10.7 [9.0–13.5]
Ankle Rotation	/	14.4 [9.5–32.9]	/	12.5 [9.2–29.7]

Table 4.2: Summary of AE values for hip discrete gait parameters derived from the original time-normalized curves. Values are reported as median [Q1–Q3].

	OpenCap	OpenCap <i>PiG-like</i>	BRIDGE	BRIDGE+AGM
H3	10.4 [6.6–16.4]	4.5 [2.8–8.3]	4.8 [1.3–14.8]	7.7 [3.4–10.2]
H6	2.9 [1.0–5.7]	3.3 [1.5–4.7]	3.8 [1.8–6.3]	3.9 [1.4–7.4]
H7	6.7 [4.8–8.0]	7.9 [5.8–9.9]	/	3.6 [2.3–6.0]
H8	6.3 [2.1–8.6]	2.7 [1.4–4.2]	/	3.1 [1.7–4.6]
H9	2.2 [1.3–3.4]	6.8 [5.2–8.2]	/	6.4 [2.1–8.4]

Table 4.3: Summary of AE values for knee discrete gait parameters derived from the original time-normalized curves. Values are reported as median [Q1–Q3].

	OpenCap	OpenCap <i>PiG-like</i>	BRIDGE	BRIDGE+AGM
K1	11.7 [6.9–15.1]	5.9 [3.5–8.1]	9.4 [5.7–12.8]	5.1 [2.3–8.8]
K2	13.3 [9.1–15.2]	5.2 [2.8–7.8]	10.8 [3.7–12.8]	6.5 [4.8–9.0]
K3	7.3 [5.0–11.2]	3.3 [1.4–6.0]	9.4 [3.5–12.4]	5.8 [3.9–7.7]
K5	9.7 [4.7–13.0]	5.2 [1.1–8.2]	6.2 [2.8–10.2]	6.0 [2.9–9.4]
K6	4.1 [1.8–6.0]	5.0 [2.8–8.3]	4.5 [2.8–7.9]	4.4 [2.4–9.2]
K7	/	10.5 [9.0–13.9]	/	8.9 [4.7–15.5]
K8	/	2.8 [1.7–5.5]	/	3.7 [0.7–7.9]
K9	/	13.1 [10.1–17.2]	/	7.6 [5.7–11.3]

Table 4.4: Summary of AE values for ankle discrete gait parameters derived from the original time-normalized curves. Values are reported as median [Q1–Q3].

	OpenCap	OpenCap <i>PiG-like</i>	BRIDGE	BRIDGE+AGM
A3	5.4 [2.8–10.2]	6.3 [2.8–12.8]	4.2 [2.5–5.6]	5.4 [2.4–8.4]
A5	9.1 [6.9–18.2]	5.6 [2.1–10.3]	5.6 [3.1–10.1]	17.1 [11.9–21.8]
A6	6.4 [3.6–8.8]	9.9 [6.1–16.1]	5.9 [2.9–8.0]	11.1 [7.8–18.1]
A10	/	11.6 [4.3–18.2]	/	11.8 [7.6–15.5]

4.2 Curves after offset removal

The same analysis was then repeated after offset removal. In this case, the reported errors mainly reflect differences in waveform shape and in the corresponding discrete gait parameters.

The distribution of RMSE values computed on the joint angle waveforms after offset removal is shown in Figure 4.8. Compared with the original curves, the overall error distributions were markedly reduced, particularly for hip and knee kinematics.

For hip flexion/extension and hip ab/adduction, all available pipelines showed low median RMSE values, generally below approximately 4° . Knee flexion/extension also showed comparable errors across pipelines, with median values around 4° – 5° . For knee varus/valgus, BRIDGE+AGM showed slightly lower median errors than OpenCap *PiG-like*, although both methods remained within a limited error range.

Ankle kinematics remained the most challenging. For ankle flexion/extension, BRIDGE showed the lowest median RMSE, around 6° , whereas OpenCap *PiG-like* showed the highest median value, close to 10° . For ankle rotation, OpenCap *PiG-like* and BRIDGE+AGM showed comparable median errors, approximately around 6° – 7° , with BRIDGE+AGM displaying a wider distribution.

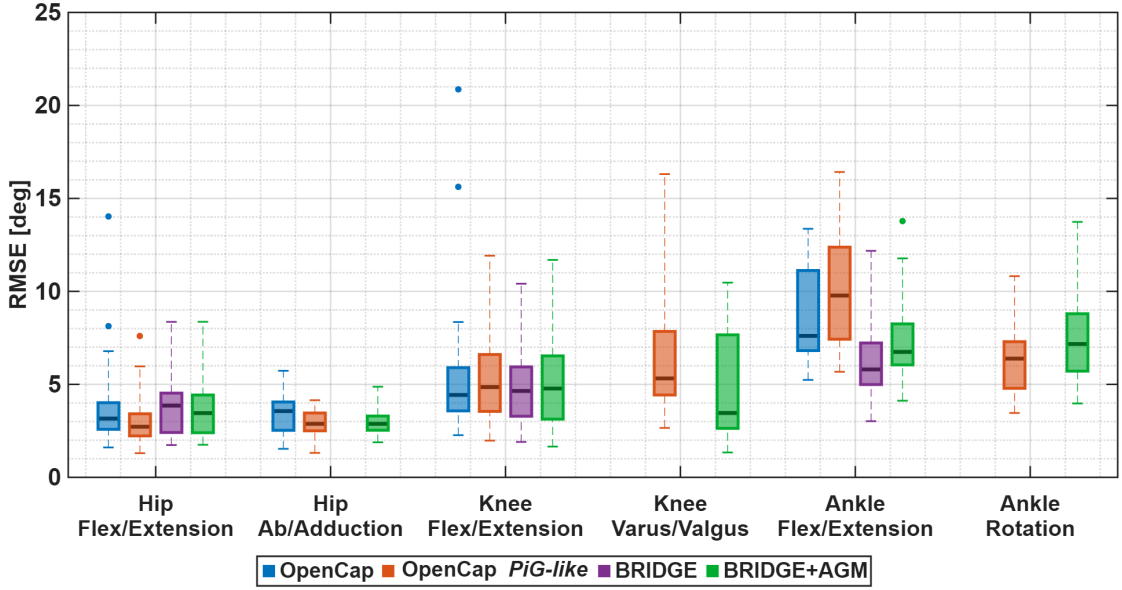


Figure 4.8: Distribution of waveform-based RMSE values.

The distribution of AE values for the hip discrete gait parameters computed after offset removal is shown in Figure 4.9. Compared with the original curves, a general reduction in the error distributions was observed.

The largest reduction was observed for H3, for which all pipelines showed low median errors, generally below 2° , together with relatively narrow distributions. For H6, median errors were slightly higher, around 3° – 4° , and a few isolated high-error values were observed for BRIDGE and BRIDGE+AGM.

For the remaining hip parameters, available only for the pipelines providing frontal-plane kinematics, median errors generally remained below 5° . H8 showed low errors across methods, with BRIDGE+AGM presenting the lowest median value. For H7 and H9, OpenCap *PiG-like* showed higher median errors, whereas standard OpenCap and BRIDGE+AGM generally showed lower values.

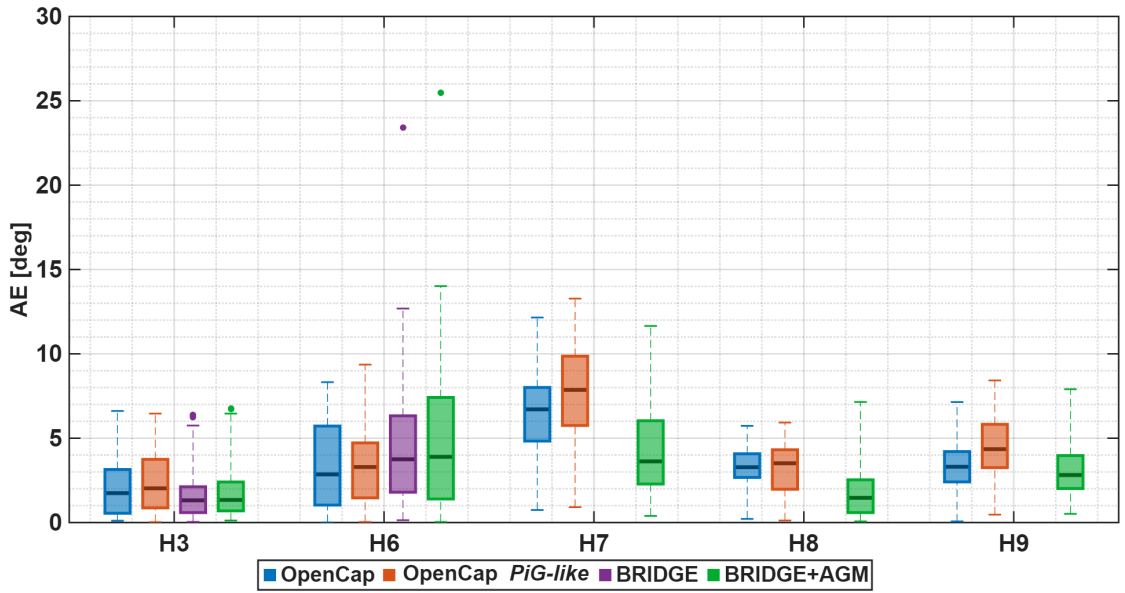


Figure 4.9: Distribution of AE values for hip discrete gait parameters.

The distribution of AE values for the knee discrete gait parameters computed after offset removal is shown in Figure 4.10. Overall, errors were reduced compared with those obtained from the original curves, particularly for the sagittal-plane parameters, with median values generally below 5° .

Among the sagittal parameters, K2, K3, and K5 showed the lowest errors, with median values mostly between 2° and 4° . K1 and K6 exhibited slightly higher errors and wider distributions, especially for BRIDGE and BRIDGE+AGM, indicating greater variability across trials.

For frontal-plane parameters, only OpenCap *PiG-like* and BRIDGE+AGM were available. K8 showed low median errors for both methods (around 2°), although OpenCap *PiG-like* presented some isolated high-error values. In contrast, K7 and K9 remained the most critical parameters after offset removal. K7 showed median errors around 9° – 10° for both methods, while K9 showed higher errors for OpenCap

PiG-like (about 9°) than for BRIDGE+AGM (about 5° – 6°).

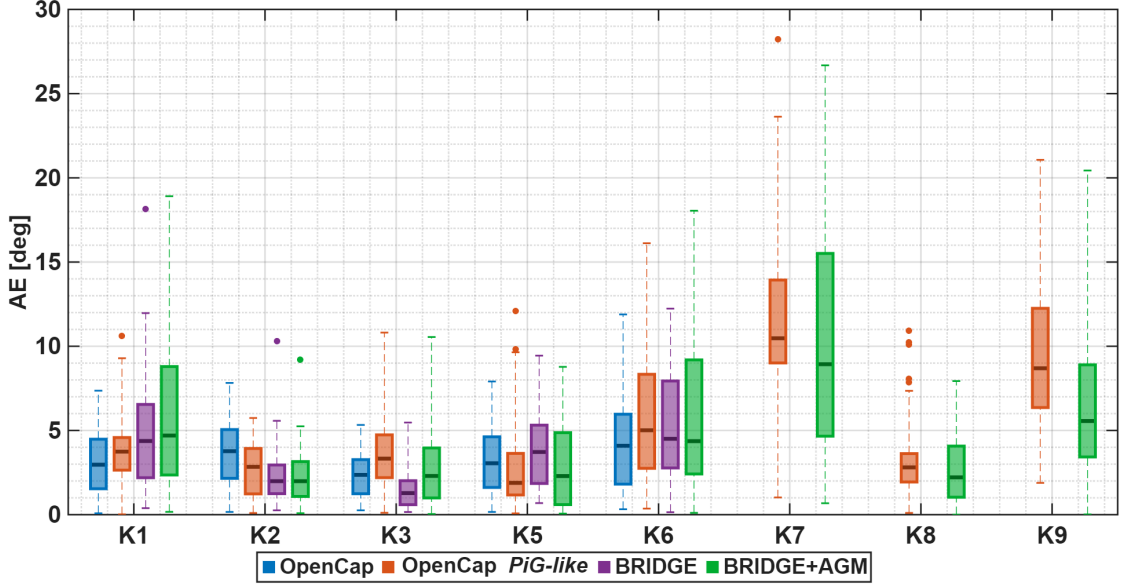


Figure 4.10: Distribution of AE values for knee discrete gait parameters.

The distribution of AE values for the ankle discrete gait parameters computed after offset removal is shown in Figure 4.11. Although a reduction in errors was observed for some parameters compared with the original curves, ankle-related parameters remained more variable than hip and knee parameters.

For A3, the lowest median errors were observed for standard OpenCap and BRIDGE, both remaining below approximately 3° . BRIDGE+AGM also showed relatively low errors, with a median value around 4° , whereas OpenCap *PiG-like* showed a higher median error, around 7° . This indicates that, even after offset removal, differences among pipelines were still evident for this parameter.

For A5, OpenCap *PiG-like* and BRIDGE showed the lowest median errors, around 4° – 5° . In contrast, standard OpenCap and BRIDGE+AGM showed higher median values, close to 8° – 10° , together with wider distributions. A similar pattern was observed for A6, where standard OpenCap and BRIDGE showed median errors around 6° , while OpenCap *PiG-like* and BRIDGE+AGM reached higher median values, approximately 10° – 11° . Finally, A10 remained the most variable ankle parameter: both available methods showed median errors around 12° and broad distributions, indicating that ankle rotation was still associated with considerable residual discrepancies after offset removal.

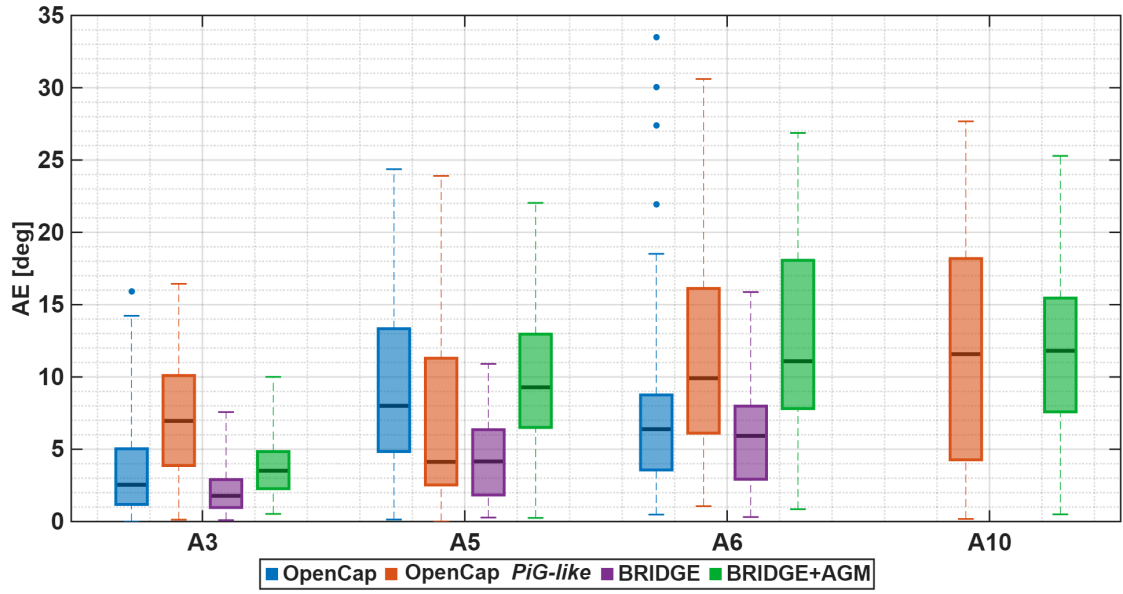


Figure 4.11: Distribution of AE values for ankle discrete gait parameters.

Table 4.5: Summary of RMSE values for joint angle waveforms after offset removal. Values are reported as median [Q1–Q3].

	OpenCap	OpenCap <i>PiG-like</i>	BRIDGE	BRIDGE+AGM
Hip Flex/Extension	3.2 [2.6–4.0]	2.7 [2.2–3.4]	3.9 [2.4–4.5]	3.5 [2.4–4.4]
Hip Ab/Adduction	3.6 [2.5–4.1]	2.9 [2.5–3.5]	/	2.9 [2.5–3.3]
Knee Flex/Extension	4.4 [3.6–5.9]	4.9 [3.5–6.6]	4.6 [3.3–5.9]	4.8 [3.1–6.5]
Knee Varus/Valgus	/	5.3 [4.4–7.8]	/	3.5 [2.6–7.7]
Ankle Flex/Extension	7.6 [6.8–11.1]	9.8 [7.4–12.4]	5.8 [5.0–7.2]	6.7 [6.0–8.3]
Ankle Rotation	/	6.4 [4.8–7.3]	/	7.2 [5.7–8.8]

Table 4.6: Summary of AE values for hip discrete gait parameters derived from the curves after offset removal. Values are reported as median [Q1–Q3].

	OpenCap	OpenCap <i>PiG-like</i>	BRIDGE	BRIDGE+AGM
H3	1.7 [0.5–3.1]	2.0 [0.9–3.7]	1.3 [0.6–2.1]	1.3 [0.7–2.4]
H6	2.9 [1.0–5.7]	3.3 [1.5–4.7]	3.8 [1.8–6.3]	3.9 [1.4–7.4]
H7	6.7 [4.8–8.0]	7.9 [5.8–9.9]	/	3.6 [2.3–6.0]
H8	3.3 [2.7–4.1]	3.5 [2.0–4.3]	/	1.5 [0.6–2.5]
H9	3.3 [2.4–4.2]	4.4 [3.3–5.8]	/	2.8 [2.0–4.0]

Table 4.7: Summary of AE values for knee discrete gait parameters derived from the curves after offset removal. Values are reported as median [Q1–Q3].

	OpenCap	OpenCap <i>PiG-like</i>	BRIDGE	BRIDGE+AGM
K1	3.0 [1.5–4.5]	3.7 [2.6–4.6]	4.4 [2.2–6.5]	4.7 [2.4–8.8]
K2	3.8 [2.2–5.1]	2.8 [1.2–3.9]	2.0 [1.2–3.0]	2.0 [1.1–3.2]
K3	2.4 [1.2–3.3]	3.3 [2.2–4.7]	1.3 [0.6–2.0]	2.3 [1.0–4.0]
K5	3.1 [1.6–4.6]	1.9 [1.2–3.6]	3.7 [1.9–5.3]	2.3 [0.6–4.9]
K6	4.1 [1.8–6.0]	5.0 [2.8–8.3]	4.5 [2.8–7.9]	4.4 [2.4–9.2]
K7	/	10.5 [9.0–13.9]	/	8.9 [4.7–15.5]
K8	/	2.8 [1.9–3.6]	/	2.2 [1.0–4.1]
K9	/	8.7 [6.4–12.3]	/	5.6 [3.4–8.9]

Table 4.8: Summary of AE values for ankle discrete gait parameters derived from the curves after offset removal. Values are reported as median [Q1–Q3].

	OpenCap	OpenCap <i>PiG-like</i>	BRIDGE	BRIDGE+AGM
A3	2.5 [1.2–5.0]	7.0 [3.9–10.1]	1.8 [1.0–2.9]	3.5 [2.3–4.8]
A5	8.0 [4.8–13.3]	4.1 [2.5–11.3]	4.2 [1.8–6.4]	9.3 [6.5–13.0]
A6	6.4 [3.6–8.8]	9.9 [6.1–16.1]	5.9 [2.9–8.0]	11.1 [7.8–18.1]
A10	/	11.6 [4.3–18.2]	/	11.8 [7.6–15.5]

Chapter 5

Discussion

The aim of this thesis was to evaluate different ML gait analysis approaches against a MB reference system, with particular attention to the role of OpenSim IK in OpenCap and of AGM in the BRIDGE pipeline. The following discussion is organized according to the main objectives of the study. The interpretation is mainly based on the results obtained after offset removal, since this analysis reduces the influence of systematic angular offsets and allows a more focused comparison of waveform morphology and discrete gait parameters. However, the results obtained from the original time-normalized curves are also considered where relevant, as they provide complementary information on the absolute agreement between methods. When interpreting these results, it should also be considered that kinematic errors in the range of approximately 2° – 5° are often regarded as acceptable in clinical gait analysis, although their relevance depends on the joint, anatomical plane, gait parameter, and intended application [49]. In this perspective, the observed differences should not be attributed only to ML reconstruction errors, but also to the different modelling assumptions and joint-angle computation strategies adopted by each pipeline.

5.1 Validation of OpenCap and effect of OpenSim inverse kinematics

One of the objectives of this thesis was to validate OpenCap against the MB reference and to assess whether OpenSim IK provided an advantage over a direct *PiG-like* computation based on OpenCap virtual markers. After offset removal, standard OpenCap provided low waveform errors for hip and knee flexion/extension and for hip ab/adduction, with values generally within or close to the range commonly considered acceptable for clinical gait analysis. This supports the ability of OpenCap to reproduce the main lower-limb kinematic patterns during walking

in healthy subjects, although the agreement with the MB reference remained dependent on the joint and anatomical plane considered.

The comparison with OpenCap *PiG-like* highlighted the main contribution of OpenSim IK at the ankle in the sagittal plane. Standard OpenCap showed lower ankle flexion/extension waveform errors than the direct *PiG-like* computation, suggesting that the model-based optimization performed by OpenSim may reduce the sensitivity to inaccuracies in virtual marker reconstruction and foot segment orientation. However, this improvement in global waveform agreement was not fully reflected in the discrete parameter A5, corresponding to maximum ankle plantarflexion during swing.

OpenCap *PiG-like* remained useful because it allowed the estimation of additional *PiG-like* quantities that are not provided by the standard OpenCap output considered in this thesis, including knee varus/valgus and ankle rotation. Nevertheless, ankle rotation showed larger errors and remained one of the most critical outputs. This result should be interpreted with caution, since transverse-plane foot kinematics are highly sensitive to small errors in foot segment orientation and anatomical landmark reconstruction.

5.2 BRIDGE performance and the role of marker augmentation

The second objective of this thesis was to evaluate BRIDGE, a subject-specific RGB-D fitting approach based on a lower-limb statistical body model. In this comparison, BRIDGE and the MB reference were based on different biomechanical formulations for joint angle estimation: BRIDGE estimated sagittal angles from joint centers, whereas the MB reference followed the *PiG* convention based on segment coordinate systems and angle decomposition. After offset removal, BRIDGE showed good accuracy for hip and knee flexion/extension, with errors generally within or close to the range commonly considered acceptable for clinical gait analysis. These results suggest that a single RGB-D camera combined with subject-specific model fitting was able to reproduce the main sagittal-plane features of healthy gait.

A relevant finding was observed for ankle flexion/extension. BRIDGE showed the lowest waveform errors for this component and also performed well for ankle discrete parameters. This behaviour is consistent with the structure of the pipeline, in which foot orientation was explicitly estimated using color-based information from the socks. Since sagittal ankle kinematics are strongly influenced by the orientation of the foot segment, this additional information likely contributed to the better performance of BRIDGE compared with methods relying only on virtual marker reconstruction.

The third objective was to investigate whether AGM could improve the sagittal

kinematics obtained with BRIDGE. In the present implementation, this improvement was not observed. Compared with BRIDGE alone, BRIDGE+AGM did not reduce sagittal-plane waveform errors and showed larger discrepancies for some ankle-related parameters, especially those depending on local peaks or foot-segment orientation. This suggests that, for sagittal kinematics, the augmented virtual markers did not provide additional information beyond the joint centers and foot orientation already estimated by BRIDGE.

A possible explanation is that the adapted pretrained AGM was not specifically optimized for the input configuration used in BRIDGE. The model was originally designed to infer anatomical markers by exploiting information from both lower limbs, whereas in this thesis it was applied to joint centers derived from a single-camera RGB-D fitting procedure. In this context, small errors in the estimated joint centers may propagate to the predicted virtual markers and subsequently affect the *PiG-like* angle computation. Therefore, the lack of improvement in sagittal kinematics should not be interpreted as a general limitation of AGM, but rather as an indication that the AGM should be specifically retrained for the characteristics of the BRIDGE pipeline.

Nevertheless, the use of AGM provided an important advantage by extending BRIDGE beyond sagittal-plane kinematics. While BRIDGE alone was limited to flexion/extension angles, BRIDGE+AGM enabled the estimation of additional frontal-plane components through a *PiG-like* workflow. After offset removal, hip ab/adduction and knee varus/valgus waveform errors remained within a limited range, suggesting that AGM can provide useful information for extending a single-camera sagittal pipeline toward a more complete three-dimensional gait analysis.

However, the analysis of discrete gait parameters showed that frontal-plane performance was not uniform. Hip parameters were generally well reproduced, whereas knee coronal-plane parameters, especially total coronal excursion (K7) and maximum adduction during swing (K9), remained more critical. This finding is consistent with previous observations by Benedetti et al., who reported that K9 may show high variability and that the reliability of knee measurements in the coronal plane can be particularly sensitive to experimental and analytical inaccuracies, including uncertainties in the anatomical calibration of the femoral epicondyles [40]. The larger errors observed for K7 may therefore be partly related to the difficulty of accurately estimating frontal-plane extrema. Since K7 represents the total coronal-plane excursion, inaccuracies in the identification of swing-phase maximum (K9) may directly affect the estimated range of motion. A similar caution is required for the ankle, which remained the most challenging joint; in particular, ankle rotation was computed from augmented virtual markers, and small errors in their position can have a larger effect on the estimated transverse-plane motion.

5.3 Limitations

This study has some limitations that should be considered when interpreting the results. First, the analysis was performed on a relatively small sample of healthy participants walking at self-selected speed. Therefore, the findings cannot be directly generalized to pathological gait, where larger asymmetries, altered movement patterns, and compensatory strategies may affect the performance of markerless pipelines differently.

Another aspect concerns the interpretation of the results after offset removal. This analysis was useful to reduce the influence of systematic angular offsets and to focus more directly on waveform morphology and discrete gait parameters. However, it also limits the assessment of absolute joint angle agreement. For applications in which absolute angular values are clinically relevant, the results obtained from the original time-normalized curves should therefore also be considered.

The acquisition protocol was also performed under controlled laboratory conditions. In particular, the BRIDGE pipeline relied on a lateral RGB-D view, on color-based information from the socks, and on the presence of a green carpet to estimate foot orientation. Although this strategy contributed to the good performance observed for sagittal ankle kinematics, it represents a pipeline-specific condition that may limit direct generalization to less controlled outpatient environments.

Another limitation concerns the quality of the RGB-D data during dynamic acquisitions. Some errors related to swing-phase parameters such as K5, K9, and A5 may be partly explained by technological limitations of the depth sensor. During swing, the shank and foot move at higher velocity, and the limited exposure time of RGB-D cameras can lead to motion blur and depth artifacts [50]. These effects may reduce the accuracy of the reconstructed lower-limb geometry and affect the estimation of joint centers and foot orientation, especially in distal segments.

A further limitation is related to the validation setup itself. Since ML and MB data were acquired simultaneously, reflective markers were present on the lower limbs during the RGB-D recordings. As shown in Figure 5.1, black or non-informative pixels were observed in correspondence with reflective markers, particularly on the foreground leg and foot. This loss of depth information may have adversely affected the fitting procedure and the precision of joint center estimation. Therefore, the reported performance of the RGB-D-based pipeline may have been influenced not only by the intrinsic limitations of the markerless method, but also by marker-related artifacts introduced by the validation protocol. In a purely markerless acquisition scenario, such artifacts would not be present, and the fitting quality, especially at the ankle and foot, could potentially improve.



Figure 5.1: Example of depth artifacts observed during the simultaneous ML and MB acquisition. Black pixels are visible in correspondence with reflective markers on the foreground leg and foot, reducing the available depth information.

5.4 Future developments

Future developments should mainly focus on improving the integration between BRIDGE and AGM. In the present study, BRIDGE+AGM enabled the estimation of additional frontal-plane components, but did not improve sagittal-plane kinematics compared with BRIDGE alone. A dedicated retraining of the AGM on inputs consistent with the BRIDGE pipeline could therefore improve the prediction of virtual anatomical markers and reduce errors to the subsequent *PiG-like* angle computation.

A second promising development would be the application of OpenSim IK to the virtual markers estimated by BRIDGE+AGM. Since OpenSim IK improved ankle flexion/extension estimates in the OpenCap pipeline, a similar model-based optimization could help improve the consistency of the full three-dimensional kinematics.

Bibliography

- [1] Richard Baker. «Gait analysis methods in rehabilitation». In: *Journal of NeuroEngineering and Rehabilitation* 3.1 (2006), p. 4. DOI: 10.1186/1743-0003-3-4 (cit. on p. 1).
- [2] Logan Wade, Laurie Needham, Polly McGuigan, and James Bilzon. «Applications and limitations of current markerless motion capture methods for clinical gait biomechanics». In: *PeerJ* 10 (2022), e12995. DOI: 10.7717/peerj.12995 (cit. on p. 2).
- [3] Steffi L Colyer, Murray Evans, Darren P Cosker, and Aki IT Salo. «A review of the evolution of vision-based motion analysis and the integration of advanced computer vision methods towards developing a markerless system». In: *Sports Medicine - Open* 4.1 (2018), p. 24. DOI: 10.1186/s40798-018-0139-y (cit. on pp. 2, 7).
- [4] Diletta Balta. «Markerless Movement Analysis Based on RGB and Depth Sensor Technology for Clinical Applications». PhD thesis. Politecnico di Torino, 2024 (cit. on pp. 3, 18, 43, 45, 49, 52, 53).
- [5] Jacquelin Perry and Judith Burnfield. *Gait analysis: normal and pathological function*. CRC Press, 2024 (cit. on p. 4).
- [6] Walter Pirker and Regina Katzenschlager. «Gait disorders in adults and the elderly: A clinical guide». In: *Wiener Klinische Wochenschrift* 129.3 (2017), pp. 81–95. DOI: <https://doi.org/10.1007/s00508-016-1096-4> (cit. on p. 5).
- [7] David A Winter. *Biomechanics and motor control of human movement*. John Wiley & sons, 2009 (cit. on pp. 6, 7).
- [8] Mrn P Kadaba, HK Ramakrishnan, and ME Wootten. «Measurement of lower extremity kinematics during level walking». In: *Journal of orthopaedic research* 8.3 (1990), pp. 383–392. DOI: <https://doi.org/10.1002/jor.1100080310> (cit. on pp. 6, 7, 33).

- [9] Cai Jiang, Xiaoling Liu, Qingping Su, Dunbing Huang, Xueling Tu, Xiaohua Ke, and Zhonghua Lin. «Gait kinematic and kinetic characteristics among older adults with varying degrees of frailty: a cross-sectional study». In: *Scientific Reports* 15.1 (2025), p. 10915. DOI: <https://doi.org/10.1038/s41598-025-95101-y> (cit. on p. 6).
- [10] Eline Van der Kruk and Marco M Reijne. «Accuracy of human motion capture systems for sport applications; state-of-the-art review». In: *European journal of sport science* 18.6 (2018), pp. 806–819. DOI: <https://doi.org/10.1080/17461391.2018.1463397> (cit. on p. 7).
- [11] Aurelio Cappozzo, Ugo Della Croce, Alberto Leardini, and Lorenzo Chiari. «Human movement analysis using stereophotogrammetry: Part 1: theoretical background». In: *Gait & posture* 21.2 (2005), pp. 186–196. DOI: <https://doi.org/10.1016/j.gaitpost.2004.01.010> (cit. on p. 8).
- [12] Vicon Motion Systems Limited. *Plug-in Gait Reference Guide*. Revision 2, for use with Nexus 2.15. Vicon Motion Systems Limited. 2023 (cit. on pp. 9, 27, 29, 33, 34).
- [13] Alberto Leardini, Lorenzo Chiari, Ugo Della Croce, and Aurelio Cappozzo. «Human movement analysis using stereophotogrammetry: Part 3. Soft tissue artifact assessment and compensation». In: *Gait & posture* 21.2 (2005), pp. 212–225. DOI: <https://doi.org/10.1016/j.gaitpost.2004.05.002> (cit. on p. 9).
- [14] Lorenzo Chiari, Ugo Della Croce, Alberto Leardini, and Aurelio Cappozzo. «Human movement analysis using stereophotogrammetry: Part 2: Instrumental errors». In: *Gait & posture* 21.2 (2005), pp. 197–211. DOI: <https://doi.org/10.1016/j.gaitpost.2004.04.004> (cit. on p. 10).
- [15] Jamie Shotton, Toby Sharp, Alex Kipman, Andrew Fitzgibbon, Mark Finocchio, Andrew Blake, Mat Cook, and Richard Moore. «Real-time human pose recognition in parts from single depth images». In: *Communications of the ACM* 56.1 (2013), pp. 116–124. DOI: <https://doi.org/10.1145/2398356.2398381> (cit. on p. 10).
- [16] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. «Openpose: Realtime multi-person 2d pose estimation using part affinity fields». In: *IEEE transactions on pattern analysis and machine intelligence* 43.1 (2019), pp. 172–186. DOI: 10.1109/TPAMI.2019.2929257 (cit. on pp. 10, 19).
- [17] Michael Zollhöfer. «Commodity RGB-D sensors: Data acquisition». In: *RGB-D image analysis and processing*. Springer, 2019, pp. 3–13. DOI: https://doi.org/10.1007/978-3-030-28603-3_1 (cit. on p. 11).

- [18] Hamed Sarbolandi, Damien Lefloch, and Andreas Kolb. «Kinect range sensing: Structured-light versus Time-of-Flight Kinect». In: *Computer vision and image understanding* 139 (2015), pp. 1–20. DOI: <https://doi.org/10.1016/j.cviu.2015.05.006> (cit. on pp. 11, 13).
- [19] Andrea Fusiello, Emanuele Trucco, and Alessandro Verri. «A compact algorithm for rectification of stereo pairs». In: *Machine vision and applications* 12.1 (2000), pp. 16–22. DOI: <https://doi.org/10.1007/s001380050120> (cit. on p. 11).
- [20] Gregorij Kurillo, Evan Hemingway, Mu-Lin Cheng, and Louis Cheng. «Evaluating the accuracy of the azure kinect and kinect v2». In: *Sensors* 22.7 (2022), p. 2469. DOI: <https://doi.org/10.3390/s22072469> (cit. on p. 13).
- [21] Lars Mündermann, Stefano Corazza, and Thomas P Andriacchi. «The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications». In: *Journal of neuroengineering and rehabilitation* 3.1 (2006), p. 6. DOI: <https://doi.org/10.1186/1743-0003-3-6> (cit. on pp. 14, 16, 17).
- [22] Paul J Besl and Neil D McKay. «Method for registration of 3-D shapes». In: *Sensor fusion IV: control paradigms and data structures*. Vol. 1611. Spie. 1992, pp. 586–606. DOI: <https://doi.org/10.1117/12.57955> (cit. on p. 16).
- [23] Stefano Pellegrini, Konrad Schindler, Daniele Nardi, et al. «A Generalisation of the ICP Algorithm for Articulated Bodies.» In: *BMVC*. Vol. 3. 2008, p. 4. URL: <https://api.semanticscholar.org/CorpusID:12104382> (cit. on pp. 16, 49).
- [24] Elena Ceseracciu, Zimi Sawacha, and Claudio Cobelli. «Comparison of markerless and marker-based motion capture technologies through simultaneous data collection during gait: proof of concept». In: *PloS one* 9.3 (2014), e87640. DOI: <https://doi.org/10.1371/journal.pone.0087640> (cit. on p. 16).
- [25] Michela Goffredo, John N Carter, and Mark S Nixon. «2D markerless gait analysis». In: *4th European Conference of the International Federation for Medical and Biological Engineering: ECIFMBE 2008 23–27 November 2008 Antwerp, Belgium*. Springer. 2009, pp. 67–71. DOI: https://doi.org/10.1007/978-3-540-89208-3_18 (cit. on p. 16).
- [26] Adrian Leu, D Ristić-Durrant, and Axel Gräser. «A robust markerless vision-based human gait analysis system». In: *2011 6th IEEE international symposium on applied computational intelligence and informatics (SACI)*. IEEE. 2011, pp. 415–420. DOI: [10.1109/SACI.2011.5873039](https://doi.org/10.1109/SACI.2011.5873039) (cit. on p. 16).

- [27] Elif Surer, Andrea Cereatti, Enrico Grosso, and Ugo Della Croce. «A markerless estimation of the ankle-foot complex 2D kinematics during stance». In: *Gait & Posture* 33.4 (2011), pp. 532–537. DOI: 10.1016/j.gaitpost.2011.01.003 (cit. on p. 16).
- [28] Andrea Castelli, Gabriele Paolini, Andrea Cereatti, and Ugo Della Croce. «A 2D markerless gait analysis methodology: Validation on healthy subjects». In: *Computational and mathematical methods in medicine* 2015.1 (2015), p. 186780. DOI: <https://doi.org/10.1155/2015/186780> (cit. on p. 17).
- [29] Evelina Pantzar-Castilla, Andrea Cereatti, Giulio Figari, Nicolò Valeri, Gabriele Paolini, Ugo Della Croce, Anders Magnuson, and Jacques Riad. «Knee joint sagittal plane movement in cerebral palsy: a comparative study of 2-dimensional markerless video and 3-dimensional gait analysis». In: *Acta Orthopaedica* 89.6 (2018), pp. 656–661. DOI: <https://doi.org/10.1080/17453674.2018.1525195> (cit. on p. 17).
- [30] Diletta Balta, Giulio Figari, Gabriele Paolini, Evelina Pantzar-Castilla, Jacques Riad, Ugo Della Croce, and Andrea Cereatti. «A model-based markerless protocol for clinical gait analysis based on a single RGB-depth camera: concurrent validation on patients with cerebral palsy». In: *IEEE Access* 11 (2023), pp. 144377–144393. DOI: 10.1109/ACCESS.2023.3340622 (cit. on pp. 17, 48).
- [31] Aditya Viswakumar, Venkateswaran Rajagopalan, Tathagata Ray, and Chandu Parimi. «Human gait analysis using OpenPose». In: *2019 fifth international conference on image information processing (ICIIP)*. IEEE. 2019, pp. 310–314. DOI: 10.1109/ICIIP47207.2019.8985781 (cit. on p. 19).
- [32] Jan Stenum, Cristina Rossi, and Ryan T Roemmich. «Two-dimensional video-based analysis of human gait using pose estimation». In: *PLoS computational biology* 17.4 (2021), e1008935. DOI: <https://doi.org/10.1371/journal.pcbi.1008935> (cit. on p. 19).
- [33] Alexander Mathis, Pranav Mamidanna, Kevin M Cury, Taiga Abe, Venkatesh N Murthy, Mackenzie Weygandt Mathis, and Matthias Bethge. «DeepLabCut: markerless pose estimation of user-defined body parts with deep learning». In: *Nature neuroscience* 21.9 (2018), pp. 1281–1289. DOI: <https://doi.org/10.1038/s41593-018-0209-y> (cit. on p. 19).
- [34] Giulia Panconi, Stefano Grasso, Sara Guarducci, Lorenzo Mucchi, Diego Minciocchi, and Riccardo Bravi. «DeepLabCut custom-trained model and the refinement function for gait analysis». In: *Scientific Reports* 15.1 (2025), p. 2364. DOI: <https://doi.org/10.1038/s41598-025-85591-1> (cit. on p. 20).

- [35] Microsoft. *Microsoft.Azure.Kinect.BodyTracking.JointConfidenceLevel*. https://microsoft.github.io/Azure-Kinect-Body-Tracking/release/1.1.x/namespace_microsoft_1_1_azure_1_1_kinect_1_1_body_tracking_adfff503ebc1491373c89e96887cad226.html. Accessed: 2026-05-15 (cit. on p. 20).
- [36] Microsoft. *Azure Kinect DK Documentation*. <https://docs.microsoft.com/en-us/azure/kinect-dk/>. Accessed: 2026-05-15 (cit. on p. 21).
- [37] Microsoft. *Synchronize multiple Azure Kinect DK devices*. <https://learn.microsoft.com/en-us/previous-versions/azure/kinect-dk/multi-camera-sync>. 2022 (cit. on p. 27).
- [38] Vicon Motion Systems Ltd. *Vicon Nexus Reference Guide*. Vicon Motion Systems Ltd. 2023 (cit. on p. 27).
- [39] Vicon Motion Systems Ltd. *Vicon Systems Setup Guide*. Vicon Motion Systems Ltd. 2023 (cit. on p. 27).
- [40] MG Benedetti, Fabio Catani, A Leardini, E Pignotti, and S Giannini. «Data management in gait analysis for clinical applications». In: *Clinical biomechanics* 13.3 (1998), pp. 204–215. DOI: [https://doi.org/10.1016/S0268-0033\(97\)00041-7](https://doi.org/10.1016/S0268-0033(97)00041-7) (cit. on pp. 35, 59, 60, 79).
- [41] Scott D Uhlich, Antoine Falisse, Łukasz Kidziński, Julie Muccini, Michael Ko, Akshay S Chaudhari, Jennifer L Hicks, and Scott L Delp. «OpenCap: Human movement dynamics from smartphone videos». In: *PLoS computational biology* 19.10 (2023), e1011462. DOI: <https://doi.org/10.1371/journal.pcbi.1011462> (cit. on pp. 35, 37).
- [42] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. «Deep high-resolution representation learning for human pose estimation». In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 5693–5703 (cit. on p. 38).
- [43] The MathWorks, Inc. *Getting Started with HRNet*. <https://it.mathworks.com/help/vision/ug/getting-started-with-hrnet.html>. Accessed: 2026-05-22. 2025 (cit. on p. 38).
- [44] Antoine Falisse, Scott D Uhlich, Akshay S Chaudhari, Jennifer L Hicks, and Scott L Delp. «Marker data enhancement for markerless motion capture». In: *IEEE Transactions on Biomedical Engineering* 72.6 (2025), pp. 2013–2022. DOI: 10.1109/TBME.2025.3530848 (cit. on pp. 39, 40, 55, 56).

- [45] Scott L Delp, Frank C Anderson, Allison S Arnold, Peter Loan, Ayman Habib, Chand T John, Eran Guendelman, and Darryl G Thelen. «OpenSim: open-source software to create and analyze dynamic simulations of movement». In: *IEEE transactions on biomedical engineering* 54.11 (2007), pp. 1940–1950. DOI: 10.1109/TBME.2007.901024 (cit. on p. 40).
- [46] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. «SMPL: a skinned multi-person linear model». In: *ACM TOG* 34.6 (2015). DOI: 10.1145/2816795.2818013 (cit. on pp. 43, 46, 47).
- [47] Nobuyuki Otsu. «A Threshold Selection Method from Gray-Level Histograms». In: *IEEE Transactions on Systems, Man, and Cybernetics* 9.1 (1979), pp. 62–66. DOI: 10.1109/TSMC.1979.4310076 (cit. on p. 52).
- [48] Elena Bergamini, Gabriele Ligorio, Aurora Summa, Giuseppe Vannozzi, Aurelio Cappozzo, and Angelo Maria Sabatini. «Estimating orientation using magnetic and inertial sensors and different sensor fusion approaches: Accuracy assessment in manual and locomotion tasks». In: *Sensors* 14.10 (2014), pp. 18625–18649. DOI: <https://doi.org/10.3390/s141018625> (cit. on p. 59).
- [49] Jennifer L McGinley, Richard Baker, Rory Wolfe, and Meg E Morris. «The reliability of three-dimensional kinematic gait measurements: a systematic review». In: *Gait & posture* 29.3 (2009), pp. 360–369. DOI: 10.1016/j.gaitpost.2008.09.003 (cit. on p. 77).
- [50] Yue Gao, You Yang, Yi Zhen, and Qionghai Dai. «Depth error elimination for RGB-D cameras». In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 6.2 (2015), pp. 1–16. DOI: <https://doi.org/10.1145/2735959> (cit. on p. 80).