A Multi-view Video Based Deep Learning Approach for Human Movement Analysis

Connor McGuirk

Thesis submitted to the Faculty of Engineering
in partial fulfillment of requirements for the degree of

Master of Applied Science

in

Biomedical Engineering

Ottawa Carleton Institute for Biomedical Engineering
University of Ottawa
Ottawa, Ontario

Abstract

Human motion analysis is an important tool for assessing movement, rehabilitation progress, fall risk, progression of neurodegenerative diseases, and classifying gait patterns. Advancements in artificial intelligence models and high-performance computing technologies have given rise to marker-less human motion analysis that determine point correspondences between an array of cameras and estimate 3D joint coordinates using triangulation. However, existing methods have not considered the physical setup and design of a marker-less human motion analysis tool that could be deployed in an institutional environment for active use, such as an institutional hallway where individuals pass regularly on a daily basis (i.e., Smart Hallway). In this thesis, camera locations were modelled, four machine vision grade cameras connected to an NVIDIA Jetson AGX were set up in a simulated institutional hallway environment, and custom software was written to capture synchronized 60 frame per second video of a participant walking through the Smart Hallway capture volume. Software was also written to calculate 3D joint coordinates and extract outcome measures for various test conditions. These outcome measures were compared to ground truth body segment length measurements obtained from direct measurement and ground truth foot event timings obtained from direct observation.

Body segment length measurements were within 1.56 (SD=2.77) cm of ground truth values. AI-based stride parameters were comparable with ground truth foot event timings and the implemented foot event detection algorithm was within 4 frames (67 ms), with an absolute error of 3 frames (50 ms) on the ground truth foot event labels. The Smart Hallway can be deployed in an unobtrusive manner and be temporally and spatially calibrated with ease. This multi-camera marker-less approach is viable for calculating useful outcome measures for clinical decision making. With these findings, marker-less motion capture is viable for non-invasive human motion analysis and compares well with marker-based systems. With future research and innovations, marker-less motion analysis will be the ideal approach for human motion analysis that requires minimal human resource to collect meaningful information.

Table of Contents

List of Figures

List of Tables

Abbreviations and Definitions

2D	Two-Dimensional
3D	Three-Dimensional
AI	Artificial Intelligence
BA	Bundle Adjustment
ChArUCo	Chessboard Augmented Reality University of Cordoba
CNN	Convolutional Neural Network
CPU	Central Processing Unit
DOF	Degree(s) of Freedom
GPIO	General Purpose Input/Output
GPU	Graphics Processing Unit
FOV	Field of View
FO	Foot Off
FPS	Frames per Second
FS	Foot Strike
FSR	Force Sensing Resistor
HPC	High Performance Computing
IMU	Inertial Measurement Units
LED	Light Emitting Diode
NAN	Not a Number

NVMe	Non-Volatile Memory Express
PCIe	Peripheral Component Interconnect express
PCLS	Physical Contact-less Systems
PCS	Physical Contact-based Systems
PnP	Point n-Perspective
POE	Power Over Ethernet
RANSAC	Random Sample Consensus
RGB	Red Green Blue
SD	Standard Deviation
SHUEE	Shriners Hospital Upper Extremity Evaluation
SSD	Solid State Drive
USB	Universal Serial Bus

Acknowledgments

I dedicate this work to my late sister, Dr. Ashlee Grace McGuirk, whose passing during my research forced me to reflect on my life, passions, and goals. Her pursuit for knowledge, and how she always strived for greatness in her work, was a driving force in helping me to achieve my best throughout my thesis. Her love is missed, but not forgotten.

I am forever grateful to my father Kenneth McGuirk, mother Cynthia McGuirk, and fiancée Amanda Danis, for their love and support throughout the course of my thesis work. Without their help and understanding I would not be where I am today.

I would like to thank Dr. Edward Lemaire, and Dr. Natalie Baddour for their advice and encouragement throughout my research. On multiple occasions they supplied me with insights, resources, and belief in my skills that I have developed throughout my thesis work. They were with me through each stage of my research and did not waiver in their weekly support, even on the weeks where progress was slow. They have provided me with invaluable opportunities, skills, and connections to help further my knowledge, and career.

This thesis is part of the CREATE-READi program funded by Natural Sciences and Engineering Research Council of Canada (NSERC), my sincere acknowledgments for financing this research.

I want to acknowledge the Ottawa Hospital Rehabilitation Centre for providing the infrastructure to develop and perform preliminary testing in the Rehabilitation Technology laboratory. As well as the University of Ottawa for providing me with the space to perform the final validation on my thesis work.

1 Introduction

Understanding human movement is important in healthcare, sports, education, and entertainment. Motion analysis provides information and insights into the quality of movement for rehabilitation, performance analysis of professional athletes, and animation for video games or computer-generated imagery in movies. Of particular interest is how motion capture technology can improve the quality of information provided to healthcare professionals. Stride analysis and various other types of gait tests are used in clinical decision making to optimally care for patients. Human motion analyses currently performed in a healthcare environment can aid in understanding rehabilitation progress [1], [2], fall risk [3], [4], progression of neurodegenerative diseases [5], and classifying gait patterns [6]. Popular human motion analysis methods include wearable inertial measurement units, force sensing techniques, or marker-based 3D reconstruction methods. However, with changes in technology that are available to the average consumer, various new techniques and tools for motion analysis can be harnessed and implemented in a way such that a completely non-invasive approach to human motion analysis is possible. The evolving field of motion analysis is headed to marker-less human motion analysis due feasibility and practicality of these types of tools compared to marker-based or sensor-based approaches.

A marker-less approach to human movement analysis can utilize an array of cameras paired with an artificial intelligence (AI) model to determine two-dimensional (2D) coordinates of a person's joints. The set of joint coordinates can then be used as point correspondences to create a three-dimensional (3D) skeleton reconstruction of the person. With this data, stride parameters and other biomechanical measurements can be extracted and used for clinical decision making [7], [8], [9]. Research is needed to validate outcome measures extracted using this type of marker-less approach.

The goal of this research is to design, develop, and implement a marker-less motion analysis system for institutional hallway settings. A non-invasive movement measurement tool that could be implemented in an institutional environment is needed for the advancement of clinical gait assessment and improving quality of care. This system could be developed and deployed specifically in an institutional hallway to make use of previously unused space in an intelligent manner. The ultimate goal for a "Smart Hallway" is to accurately and non-invasively assess human movement in an institutional setting with minimal human intervention.

1.1 Rationale

Understanding how a person moves is essential when assessing recovery from surgery, progress in a physical rehabilitation program, or decline in elderly care environments. Currently, this understanding comes from visual movement analysis or clinical tests that are based on simple measures such as the time to complete the test and Likert scale performance ratings [10]. If quantitative outcome measurement could be integrated into the care facilities, many opportunities can be presented to provide additional and better information for decision-making.

Currently, human motion analysis involves tracking markers on the body. These reflective or active marker systems can be difficult to implement at hospitals or other facilities due to hardware and software costs, human resource requirements, and space requirements. In addition to the need for dedicated space, the time to attach dozens of markers, capture the movement trials, process the 3D marker data, and generate outcome measures is beyond the capacity of most organizations. To potentially address some of these difficulties, marker-less motion capture systems are emerging that use pose inference AI for contactless joint tracking. Such a system could use lower-cost consumer cameras and open-source software, which make system costs more accessible.

With recent advancements in high-performance computing and AI models, marker-less motion analysis systems have the potential to provide accurate and precise human movement analysis. Combined with video capture devices that provide megapixel image streams at or above 60 frames per second (fps), this type of human motion analysis system could be implemented in institutional areas with available space for cameras and where motion analysis is desired; for example, acute healthcare, rehabilitation, and eldercare environments [11], [12]. This type of marker-less system could form a more general tool for automatic, and non-invasive motion data acquisition, such as a Smart Hallway where an automated system is installed in a patient accessible institutional hallway.

A Smart Hallway implementation could automatically record movement as a person walks into a clinic, so that therapist or physician can review walking parameters before their appointment. In an elderly care residence, movement data could be collected multiple times a day, as the person walks through the Smart Hallway, thereby providing data to track changes in movement quality. Big data models could also be implemented to provide indicators for fall risk or dementia progression. These new capabilities rely on automated movement data acquisition without human intervention.

1.2 Objectives

The objective of this thesis was to develop and evaluate a camera-based system for marker-less human movement measurement within institutional hallways (Smart Hallway). Specific objectives are:

1. Determine an appropriate artificial intelligence model for pose inference within an institutional hallway.
2. Design an appropriate hardware and software system for multi-camera image collection and processing for a hallway setup.
3. Validate the Smart Hallway performance through analysis of participant stride parameters and limb lengths.

1.3 Thesis Contributions

This thesis produced several important contributions for knowledge and marker-less motion analysis technology:

- Created a viable Smart Hallway system for 3D pose inference in an institutional hallway. Through design, development, and analysis, a basis was provided for integrating 3D motion analysis in eldercare and clinical environments.
- Determined that OpenPose BODY25 is superior to HyperPose COCOv2 for clinically relevant human motion assessment of stride parameters and body segment locations.
- Determined that, for a multi-camera system, keypoints connected to multiple segments (i.e., keypoints that are not endpoints) are more accurate. Therefore, a model with multiple connected segments that are constrained relative to each other is recommended for any application that reports stride parameters.
- Established a viable method to implement synchronization and calibration that can be easily implemented in institutional environments.
- Determined that the Zeni et al. [13] foot event detection algorithm was appropriate for use with the Smart Hallway's 3D marker-less foot tracking keypoints.
- Determined that the Smart Hallway outcome measures (body segment lengths, stride parameters, joint and limb angles) were viable for use in clinical decision-making.

1.4 Thesis Outline

Chapter 2 is a literature review that discusses current methods of motion analysis, covering contact-based and contact-less systems, artificial intelligence approaches to human motion analysis, hardware options for motion capture systems, and current computer vision techniques used for 3D reconstruction.

Chapter 3 compares OpenPose BODY25 and HyperPose COCOv2 AI models to determine the quality of joint keypoints when performing clinically relevant movements and to specify criteria for capturing "AI friendly" videos.

Chapter 4 discusses the component selection process, geometric placement of cameras and physical design, multi-camera synchronization solution, and optimized camera calibration routine that are a part of the overall Smart Hallway design.

Chapter 5 presents stride parameter and limb segment length analyses to validate the system. The data collection protocol and data analysis are detailed.

Chapter 6 presents a summary of the thesis work and contributions, thesis conclusions, and suggestions for future work in the context of marker-less motion analysis in an institutional setting.

Appendix A contains the University of Ottawa Ethics approval certificate.

Appendix B contains supplementary tests performed on the hardware synchronization cable using an oscilloscope to analyse the signal characteristics.

Appendix C provides the outcome measures gathered from another participant.

2 Literature Review

Human motion analysis is important for understanding sports biomechanics, evaluating movement-aid devices, and providing information on a person's overall physical health [14]. Motion analysis can be used for gait, upper extremity movements, head and neck movements, fine movements in hands or feet, and posture or trunk movement [15]. Quantitative and qualitative movement data can provide important information for clinical decision making [14].

Gait analysis is used in clinical settings to provide information on patient mobility, rehabilitation, and progression of neurological disorders [14]. Frequently monitoring patient gait can provide useful assessment information. Gait analysis can be performed qualitatively and quantitatively, with each method allowing a clinician to interpret different aspects of a patient's movement. Qualitative gait analysis can be used to study walking asymmetries, pronation type, or perform pain assessment. This type of analysis relies heavily on patient feedback and can vary from clinician to clinician. Quantitative gait analysis techniques provide data relating to limb kinematics and kinetics. Quantitative data can be acquired using accelerometers, gyroscopes, force plates, and motion capture systems like Vicon. These data acquisition methods can provide temporospatial features like position, joint angles, velocities, and accelerations for assessing fall risk [16], balance [17], and mental capacity [18].

2.1 Analyzing Human Movement

Step timing and distance features from quantitative gait analysis are referred to as stride parameters and describe a person's gait cycle. A walking cycle is defined by a series of events and occurs when the body moves forward with one leg acting as a support while the other leg advancing [19] (Figure 2.1).

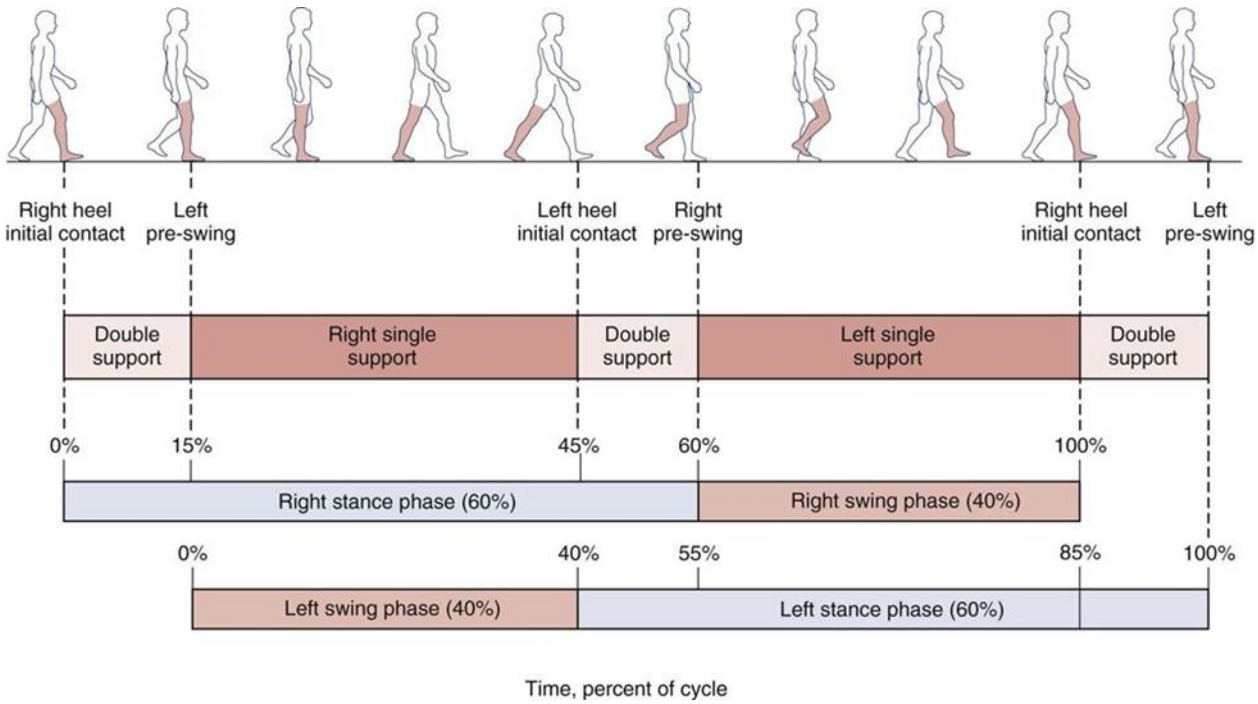


Figure 2.1 Gait cycle and events [20].

Some stride parameters typically used by clinicians are show in Table 2-1 [21].

Table 2-1 Stride parameters used for quantitative gait analysis in clinical applications.

Parameter	Description
Stride length	Distance covered by the foot between two consecutive foot strikes (same leg) along the sagittal plane (line of progression)
Stride time	Time between two consecutive foot strikes (same leg)
Stride speed	Stride length divided by stride time
Step length	Sagittal distance between right and left heels at foot strikes
Step width	Frontal distance between right and left heels at foot strikes
Cadence	Steps per minute
Stance time	Stance phase time
Swing time	Swing phase time
Foot angle	Angle of foot midline with sagittal plane when the foot is entirely in contact with the floor (mid-stance period)
Double support time	Time between foot strike and contralateral foot off

2.2 Physical Contact-based Systems (PCS)

Physical Contact-based Systems (PCS) are motion analysis systems where the participant must contact some part of the system. PCS systems use inertial measurement units such as accelerometers and gyroscopes [22]; electromechanical position sensors such as potentiometers (wearable motion capture suits) [23], [24]; or reflective markers paired with light emitting cameras.

Many of these methods require the participant to wear sensors or markers. These methods suffer from signal artifacts due to wearable sensor micro-movements caused by non-rigid connections [1]. Also, wearable sensor approaches can potentially affect a person's natural gait, thereby producing results that do not reflect the person's typical walking style [1].

Force sensing methods do not require the participant to wear sensors, but still require the participant to contact a sensing device. Force sensing methods capture changes in voltage caused by a load being applied to the sensor [1], [25]. Force plates are commonly used in a large laboratory setting where multiple plates can be positioned together to capture multiple steps [25]. Instead of embedded force plates, sensor mats such as the GAITRITE carpet are also available [26]. Disadvantages to this approach include sensor fusion techniques to chain the signals together that can cause noise in the final output and, without a large space, only data from one stride can be captured [1], [25]. Force sensing insoles can combine wireless data transfer with force sensing resistor (FSR), capacitance, or piezo-electric sensing fabrics to create a force map of the foot [1], [27]. Disadvantages of instrumented insoles is that shoes may inadvertently affect measurement validity and stride distances cannot be measured [27]. Force plate methods only capture data while the foot is contacting the ground and information is lost while in swing phase; however, insoles provide data for multiple gait cycles [27]. Instrumented exoskeletons and prostheses also allow data collection from participants that use a mechanical or mechatronic aid while walking [28]. All non-wearable sensor approaches, such as the ones mentioned above, also have a shortcoming in that they cannot measure body kinematics directly and instead rely on inverse kinematic calculations [1]. Figure 2.2 displays these gait analysis methods.

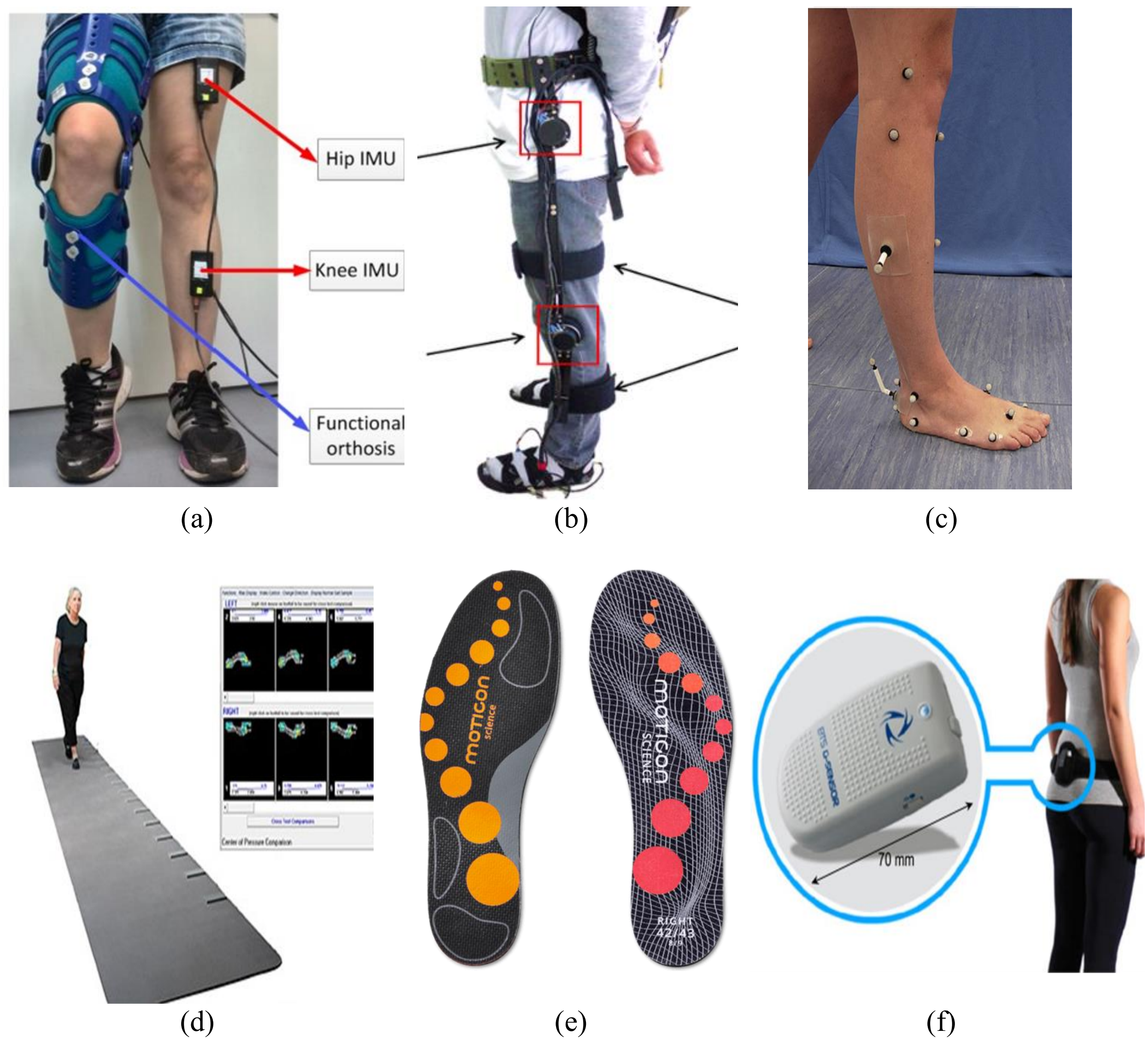


Figure 2.2 Motion capture tools: a) Inertial measurement units [21], b) Potentiometer connected to rigid orthosis to measure joint angle [22], c) VICON reflective markers [23], d) GAITRITE force sensing carpet [29], e) Moticon pressure sensing insoles [30], f) Wearable pelvis sensor [31].

From the literature, reflective marker approaches combined with light emitting diode (LED) ring cameras are the current gold standard for motion analysis [29], [32]. Vicon systems use an optoelectric approach to track reflective markers, generating high accuracy 3D locations that can be used to measure temporospatial gait features [2]. These optoelectric approaches require expensive hardware and dedicated lab space, along with the need to affix multiple markers to the participant. These disadvantages make these systems difficult to integrate outside of facilities containing a motion capture laboratory [33].

2.3 Physical Contact-less Systems (PCLS)

Physical contact-less systems (PCLS) track participant motion non-invasively (i.e., no participant contact with any part of the system). These systems can use RGB (Red-Green-Blue) cameras and visual hull reconstruction based on an *a priori* model of the participant [6], [34], [35]; depth sensors combined with RGB cameras to extract 3D points and segment key areas using machine learning to obtain a point cloud of the participant [36], [37], [38]; and RGB cameras paired with a pose inference AI model [39], [40], [41].

Methods that map their output point cloud to an *a priori* model are cumbersome since a full 3D model of the participant is needed before accurate data acquisition can occur, thus restricting flexibility and deployability [6]. Also, this brings into question how non-invasive this approach may be, since obtaining an accurate initial participant model may use a contact-based method. Furthermore, visual hull reconstruction is susceptible to noise due to clutter or other individuals that are in frame, which further constrains the area in which such a system can be used [6], [34].

Combining RGB cameras and depth cameras, such as Intel RealSense D435, has the disadvantage of poor depth resolution that creates noisy output results [42], [43]. Also, depth sensors relying on structured light can be susceptible to errors caused by surface reflectivity and interference with other structured light cameras [44], [45], [46].

Systems relying on RGB cameras paired with AI models for semantic segmentation rely on camera arrays to precisely find point correspondences and 3D position by reconstruction from multiple 2D points [47], [48]. These systems rely heavily on the quality of 2D point correspondences and geometric structuring of the cameras to obtain accurate results. However, accurate results have been achieved at distances greater than ten meters [49].

Depth measurement can be improved by using multiple calibrated cameras in a stereo-camera system. Stereo-system robustness is directly related to the number and quality of point correspondences [50]. Stereo-camera results could be improved by incorporating AI models trained to learn depth perception [51], [52]; however, few open-source AI models for monocular depth estimation are available and these models have not achieved appropriate accuracy [50], [53] (i.e., 0.5 m error at 5 m working distance, and 1 m error at 10 m working distance).

Various marker-less motion analysis systems have been reported. Labuguen *et al.* [54] implemented a four-camera system capturing at 720p resolution and 30 frames per second (fps) for 3D reconstruction of joint positions. OpenPose BODY25 was solely used to detect keypoints and no tracking, filtering, or foot event detection was implemented for full movement analysis. Results from this approach had a maximum average error of 7cm on joint position. This system was also limited in its ability to track highspeed movements due to the low capture rate. Other methods implementing Microsoft Kinect ® sensors are limited in scalability due to each sensor requiring its own computer for processing [55]. As well, depth sensor approaches have significant noise when assessing foot contact events and suffer due to increasing depth estimation error at long ranges [43]. Due to the useable range of Kinect sensors, they typically must be kept close to the participant, as in the system implemented by Rodrigues *et al.* [55] where sensors were placed at waist level within two to three meters of the participant. This system utilized an array of Kinect sensors calibrated to each participant and synchronized at 35fps. This method of calibration is too cumbersome to deploy for daily use and is therefore not adaptable to a Smart Hallway type application. Methods that implement RGB camera arrays with pose inference models have shown promising results for human motion analysis, such as the system implemented by Nakano *et al.* [56]. However, they did not implement proper synchronization methods or state-of-the-art calibration approaches needed for developing a useable system capable of producing semi-realtime results. Also, the Nakano *et al.* system does not automatically detect foot events and requires post-analysis of the data to segment strides, which does not meet the objectives for the Smart Hallway. Overall, these existing marker-less motion analysis approaches included some but not all the elements required for an automated Smart Hallway application.

2.4 Deep Learning for Pose Estimation

Deep-learning-based marker-less motion analysis tools rely heavily on AI pose model accuracy and precision to provide point correspondences that can be triangulated to produce high quality 3D data points. These AI models are commonly built with convolutional neural network (CNN) architecture, popularized for their performance in the ImageNet Large Scale Visual Recognition Challenge 2012 competition [57]. The CNN architecture finds features in images by applying different convolutional filters at each stage and extracting individual pixel values. These convolutional filters, also known as kernels, highlight certain features in an image such as edges and areas of high contrast (Figure 2.3).

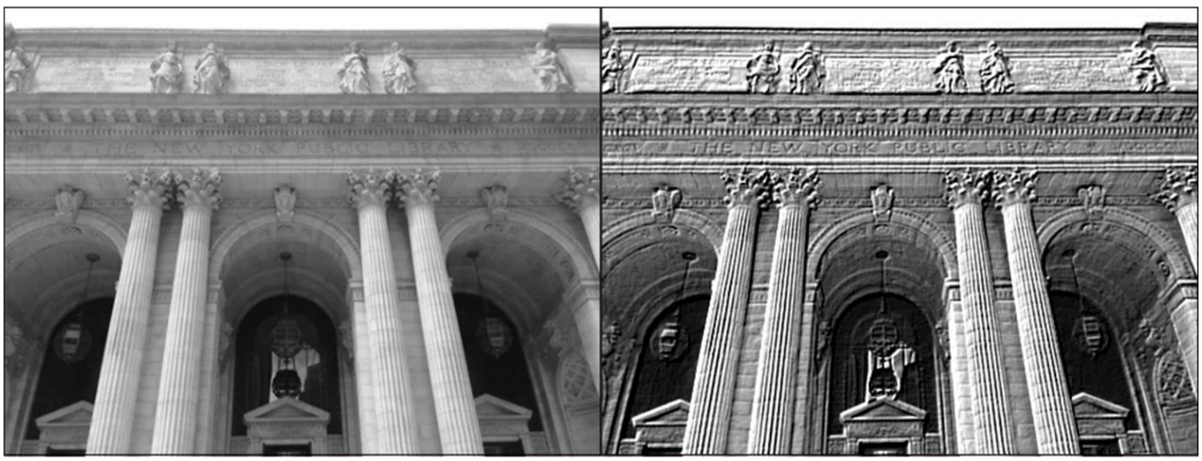


Figure 2.3 Embossing kernel applied to highlight areas of high contrast [58].

AI models focusing on a person's pose are termed pose estimation models. In the context of analyzing people, these models can describe the type of action a person is performing, such as sitting, standing, or falling [3], [59]; use semantic segmentation to describe where a person or persons are in an image [38], [60]; and determine semantic keypoint locations to produce a skeleton model for each person in frame [41], [61]. Figure 2.4 shows a pose estimation model for semantic segmentation and Figure 2.5 for pose classification. The disadvantage of action or gesture classification models is that minimum detail is provided to aid in decision making for clinically useful systems. Models that segment regions of the image to describe a person's location are susceptible to noise and produce point clouds with high variance that makes calculating accurate stride parameters difficult [62], [63]. Models that rely on accurately finding 2D joint centers can possibly provide high quality point correspondences that could be used for triangulation [64].

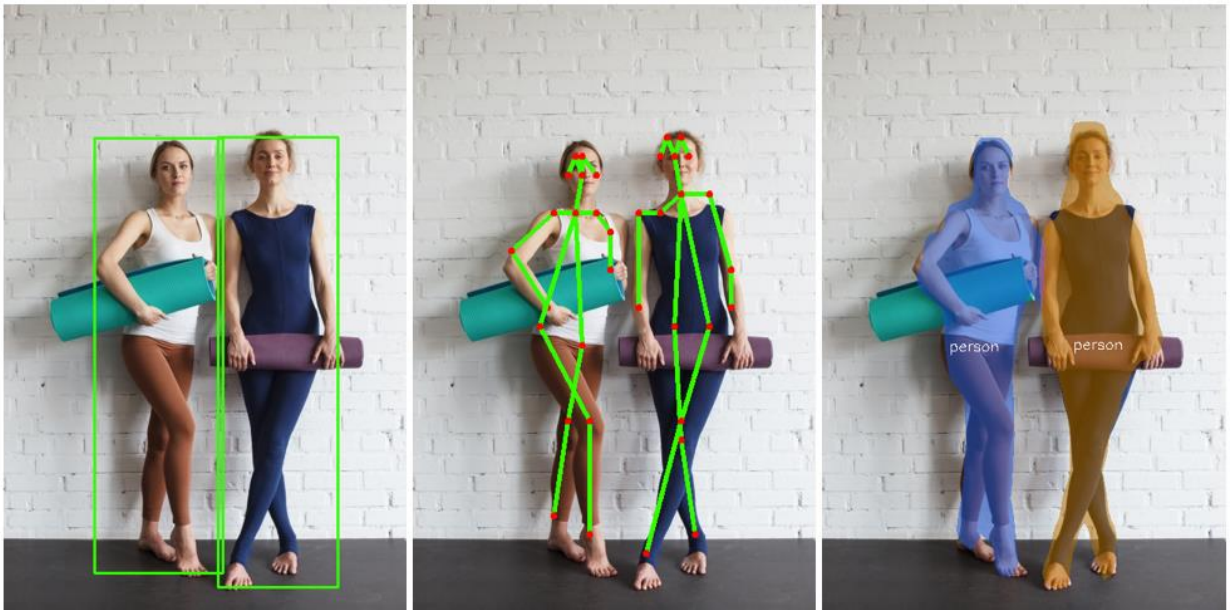


Figure 2.4 Pose inference models for detection, keypoint allocation, and semantic segmentation [65].

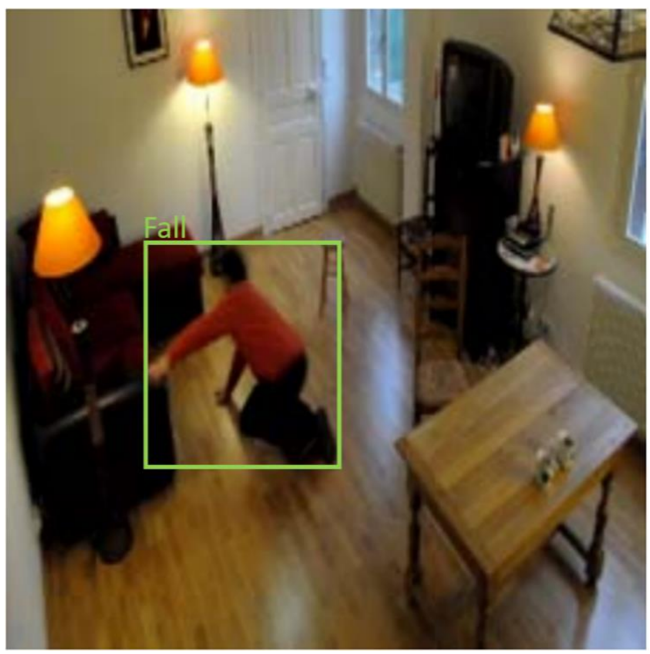


Figure 2.5 Pose inference model used for pose classification of a fall [4].

A model created by the Carnegie Melon University Perceptual Computing Lab, called OpenPose, has shown state-of-the-art results when detecting keypoints that relate to a person's joint centers [41], [66]. The training datasets used to learn these keypoint representations included Common Objects in Context (COCO) and MPII for foot keypoint labels. Both datasets are comprised of people in various poses performing different activities [66]. OpenPose has several variants such as mobile net, COCOv2, and BODY25 that make compromises to either increase accuracy and increase processing time or reduce accuracy and reduce processing time. OpenPose also implements new methods for intelligently assigning body parts to specific people, to handle situations where multiple people are detected in frame. For instances where multiple people are detected, techniques called part affinity fields are used by OpenPose to intelligently determine which limbs belong to which individual in frame (Figure 2.6).

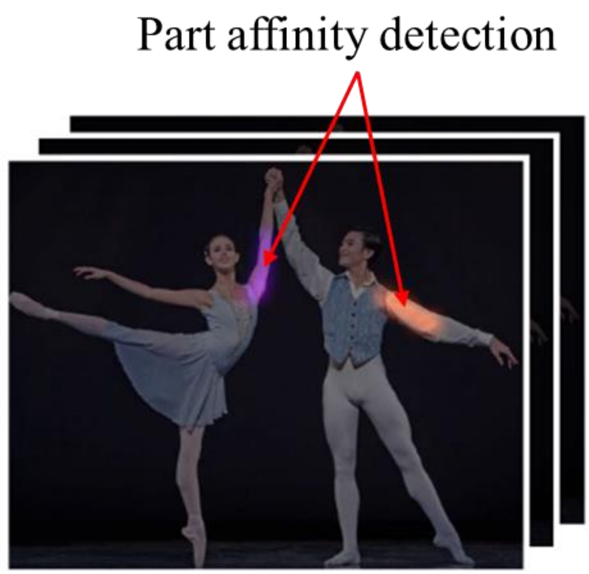


Figure 2.6 OpenPose making deterministic decision to assign limbs to the appropriate person [41].

Along with part affinity fields, OpenPose also implements methods to accurately obtain keypoint inferences. OpenPose models (Figure 2.7) use a multi-structured approach that combines two different CNN at each stage. Network one predicts a confidence map for each desired keypoint and network two predicts an affinity field describing the degree of association between distinct keypoints [41].

However, the high-accuracy OpenPose model processing time is too high for most real time applications. Recently released models adapted OpenPose's architecture to optimize several graphics processing unit (GPU) routines in an effort to provide high accuracy in real time. HyperPose, a variant of OpenPose's architecture, utilizes NVIDIA's GPU programming libraries to produce models with the same accuracy but reduced processing time [67]. These models combined with a multi-camera system could provide a tool for determining 2D point correspondences and calculating accurate 3D joint positions without markers.

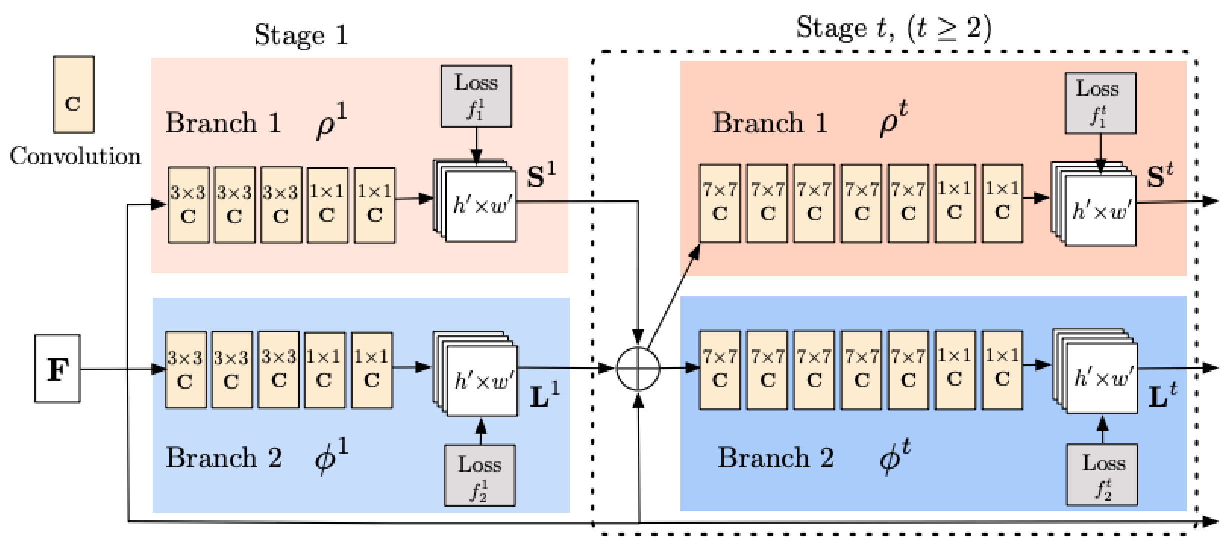


Figure 2.7 Multi-branch architecture used in OpenPose to generate confidence maps and part affinity fields for accurate keypoint allocation [41].

2.5 High Performance Computing Platforms for Movement Analysis

Motion analysis using data from multi-camera systems involves large amounts of computations and non-linear optimization methods to obtain quality 3D results. Motion analysis with marker-less systems adds AI model processing to this pipeline. Due to these requirements, high performance computing (HPC) is needed to extract quantitative motion analysis data in a reasonable time. HPC systems for gait labs, hospitals, or care facilities can use a local server for real-time data transmission and processing [68]; stand-alone high performance personal computer [69]; or single board HPC units [70], [71]. Passing data to internal or cloud-based servers has the disadvantage of being difficult to implement for many institutions or care facilities that lack the information technology staff to upkeep and maintain these systems. Also, passing patient data to a server that is not directly accessible increases personal data security risk and creates the need for data encryption [72]. Another disadvantage of this approach is that transferring multiple video clips or image sequences to the cloud requires time and large bandwidth, with a wired connection desirable but possibly not feasible, depending on the location of the camera system relative to the server [73]. A stand-alone personal computer solution would be easier to maintain and secure the HPC system; however, connection cables for wired file transfer may need to be routed distances that are not possible with some cable types. Also, the need for a dedicated setup near the multi-camera system is still prevalent. Single board HPC units can perform all data transfer and processing at the data collection source. This type of device provides flexibility in system placement, ease of setup, and is typically less expensive than a stand-alone personal computer or server that is capable performing the necessary computations. HPC device cost is important for maintaining a low "system cost to performance" ratio, which increases the likelihood Smart Hallway adoption into institutional environments such as rehabilitation centers or eldercare environments. Single board HPC units available to consumers include: Raspberry Pi 4, Google Coral, NVIDIA Jetson Nano, NVIDIA Jetson TX2, and NVIDIA Jetson AGX Xavier. The specifications of these are given below.

2.5.1 Raspberry Pi 4 [74]

- **Cost:** $104.00 CAD
- **CPU:** Broadcom BCM2711, ARM v8 64-bit
- **GPU:** None, external options available
- **RAM:** 8GB LPDDR4
- **Wireless:** 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE, Gigabit Ethernet
- **USB:** 2 x USB-A 3.0; USB-C 3.0; 2 x USB-A 2.0
- **PCIe Slot:** None
- **Storage:** 1 x Micro-SD
- Inexpensive but without AI optimization and lower processing capability. Peripheral options for multiple camera connections are lacking. Could act as a node to handle camera data transfer and image processing if used in conjunction with an AI capable module.

2.5.2 **Google Coral** [75]

- **Cost:** $198.00 CAD
- **CPU:** NXPi.MX 8M
- **GPU:** Integrated GC7000 Lite
- **Machine Learning Acceleration:** Google Edge TPU coprocessor (4 TOPS, int8)
- **RAM:** 4GB LPDDR4
- **Storage:** 1 x Micro-SD; 8GB eMMC
- **Wireless:** 802.11b/g/n/ac
- **USB:** 1 x USB-C; 1 x USB-A 3.0
- **PCIe Slot:** None
- Provides computing power necessary for AI applications; however, computing power is still less than the Jetson Nano. Lack of connectivity for imaging sensors. Would require a secondary module to handle video capture and data transfer.

2.5.3 **Jetson Nano** [76]

- **Cost:** $90.00 CAD
- **CPU:** ARM Cortex-A57
- **GPU:** NVIDIA Maxwell
- **RAM:** 4GB LPDDR4
- **Storage:** 1 x Micro-SD; 16GB eMMC
- **Wireless:** None
- **USB:** 4 x USB-A 3.0; 1 x USB-uB 2.0
- **PCIe Slot:** None
- Improved computing power compared to the Google Coral and improved connectivity for external imaging sensors. Expected >0.4 frames processed per second using OpenPose BODY25.

2.5.4 Jetson TX2 [76]

- **Cost:** $500.00 CAD
- **CPU:** 2 x Denver 64-bit; ARM A57 Complex
- **GPU:** NVIDIA Pascal
- **RAM:** 8GB LPDDR4
- **Storage:** 1 x Micro-SD; 32GB eMMC; M.2 Slot
- **Wireless:** 802.11ac
- **USB:** 1 x USB-A 3.0; 1 x USB-uB 2.0
- **PCIe Slot:** PCIe x4
- More than double expected performance compared to the Jetson Nano and increased RAM and storage capabilities. RAM is still insufficient for handling more computationally heavy versions of OpenPose (BODY25 increased net size). PCIe x4 allows for reasonable bandwidth and improved connectivity.

2.5.5 Jetson AGX [76]

- **Cost:** $900.00 CAD
- **CPU:** 8-core ARM v8.2; 8MB L2 + 4MB L3
- **GPU:** NVIDIA Volta
- **RAM:** 32GB LPDDR4
- **Storage:** 1 x Micro-SD; 32GB eMMC; M.2 Slot
- **Wireless:** None, external options available
- **USB:** 1 x USB-A 2.0; 2 x USB-C 3.1
- **PCIe Slot:** PCIe x16 interface with PCIe x8
- Provides triple expected performance compared to the Jetson TX2, however at a higher cost. Capable of performing inference with computationally heavy models. PCIe x8 allows for high bandwidth data transfer and connectivity for multiple sensors.

The NVIDIA Jetson line of products provided the best starting point for designing a system capable of handling data transfer from multiple cameras, inference using AI, and data processing. Connectivity options for sensors and HPC power on the single board computer make the NVIDIA Jetson products a flexible choice when designing a Smart Hallway.

2.6 Computer vision Methods for 3D Point Reconstruction

Reconstruction of a 3D real-world object from multiple 2D images involves many crucial steps. The process of reconstruction starts with selecting a mathematical camera model that describes the camera sensor and camera lens properties, such that pixels in the image can be associated with rays in space [77]. These camera models are used to transform points in a 2D image (image points) to a ray, or point up-to-scale, in the corresponding 3D camera frame. Conversely, a camera model is more commonly used in the reverse form to map 3D points in the camera frame to 2D points on the image plane. Camera models can broadly be divided into two groups by specifying if they capture perspective [78]. Perspective is the property that causes objects close to the camera to appear larger than objects that are further away [78]. Common models that do not capture perspective include the orthographic model, weak-perspective model, and affine model. For perspective models, the pinhole camera model is ubiquitous throughout computer vision research, and other perspective models are typically degenerate cases of the pinhole model [78], [79]. The pinhole camera model uses a three-by-three matrix to capture the focal length and camera center in the X and Y axes. Methods to model the non-linear properties inherent in camera sensor and lens combinations are coupled with these linear models. Non-linearities can be modelled by applying a distortion model to the pinhole camera model that accommodates non-linear distortions caused by lens edge and lens mounting effects [79], [80], [81]. Other distortion models can be used for wide field of view (FoV) cameras or cameras with unique sensor arrays [82]. Distortion parameters commonly used for regular cameras include three coefficients describing radial distortions and two coefficients describing tangential distortions [83], [84]. The pinhole camera model paired with a distortion model forms the basis for mapping points in the camera frame to the image plane. These intrinsic parameters are described by equations (2.1) and (2.2).

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 0 \end{bmatrix} \tag{2.1}$$

$$C_d = \begin{bmatrix} k_1 & k_2 & p_1 & p_2 & k_3 \end{bmatrix} \tag{2.2}$$

In equation (2.1), f_x and f_y represent the focal length in the x-axis and y-axis in pixels. c_x and c_y represent the camera image center in the x-axis and y-axis in pixels. In equation (2.2), k_1, k_2, and k_3 are coefficients to model radial distortion and p_1 and p_2, are coefficients for tangential distortion [84]. The distortion coefficient vector is used to undistort the points of interest, using (2.3) and (2.4) for radial and tangential distortion.

$$p_r(x,y): \begin{cases} x_{distorted} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ y_{distorted} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \end{cases} \tag{2.3}$$

$$p_t(x,y): \begin{cases} x_{distorted} = x + \left(2p_1 xy + p_2(r^2 + 2x^2)\right) \\ y_{distorted} = y + \left(p_1(r^2 + 2y^2) + 2p_2 xy\right) \end{cases} \tag{2.4}$$

With a set of undistorted points, mapping from the 3D camera frame to the 2D image plane is described in equation (2.5) [85].

$$sP_{image} = K[R|t]P_{camera}, \qquad s\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.5}$$

The matrix $[R|t]$ describes the six degree of freedom (DoF) camera pose with respect to the world-origin. In this scenario, the identity is used since the world origin is chosen as the camera frame origin. Homogenous coordinates simplify translation, rotation, scaling, and perspective transforms by allowing them to be implemented through matrix multiplication. With one camera, image points can only be traced to a ray in space and described up to an unknown scale factor, s [85].

Determining these parameters is possible through physical testing and measurement; however, a more practical approach is to robustly determine the parameters by minimizing pixel reprojection error. Reprojection error is defined as the distance between a detected point on the image plane and its respective point in 3D camera space transformed to the image plane. Thus, with a perfect determination of camera parameters, 3D points in the cameras space perfectly map to their counterpart on the 2D image plane. An accurate solution can be determined using the Levenberg-Marquardt algorithm and modelled as a least-squares curve fitting problem. Equation (2.6) shows the basis of the camera calibration problem and pixel reprojection error minimization [85].

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left\|m_{ij}-\hat{m}\left(\boldsymbol{K},\boldsymbol{k_1},\boldsymbol{k_2},\boldsymbol{p_1},\boldsymbol{p_2},\boldsymbol{k_3},\boldsymbol{R_i},\boldsymbol{t_i},M_j\right)\right\|^2 \tag{2.6}$$

In equation (2.6), m_{ij} is the detected image plane point and $\hat{m}\left(\boldsymbol{K},\boldsymbol{k_1},\boldsymbol{k_2},\boldsymbol{p_1},\boldsymbol{p_2},\boldsymbol{k_3},\boldsymbol{R_i},\boldsymbol{t_i},M_j\right)$ is the projection of point M_j from image i [85].

Multiple sets of image points are required for an optimization process that provides a robust estimate of camera intrinsic parameters. For this purpose, pattern identification algorithms used to detect corners, or binary markers, are employed to create a collection of image points for calibration [85], [86], [87]. From the literature, the chessboard augmented reality University of Cordoba (ChArUCo) calibration pattern offers an optimal method for robust detection of calibration points [87]. The chessboard pattern allows for easy detection and refinement of corner points, while the ArUCo markers improve robustness by allowing corners to be found even when partially occluded [87]. Figure 2.8 displays common calibration patterns used for intrinsic parameter optimization.

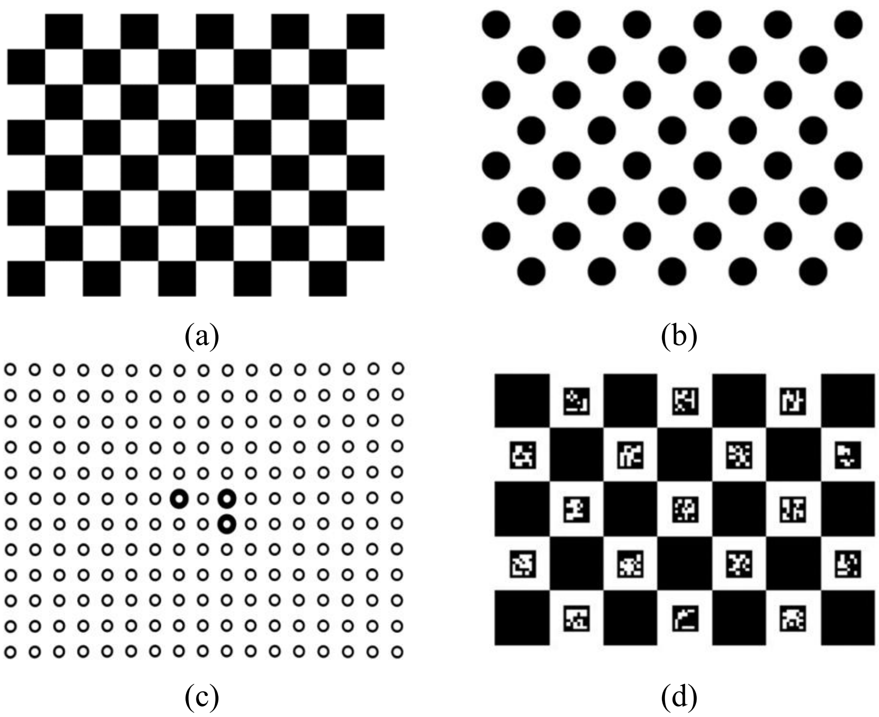


Figure 2.8 Common calibration patterns. a) Chessboard, b) Circular, c) Circular with Orientation Markers, d) ChArUCo

Once intrinsic parameters are obtained for each camera, a ray can be calculated in each camera's respective 3D camera frame. The set of rays ideally correspond to the same point in the world-coordinate system. Measurements with multiple cameras require a world-origin for points in each camera space to be transformed when calculating the overall intersection. Referring back to the intrinsic calibration process, the matrix $[R|t]$ can be obtained by solving the perspective-n-point (PnP) problem for a given set of points [88]. Figure 2.9 displays the coordinate systems of interest and the rigid transformation calculated by solving the PnP problem.

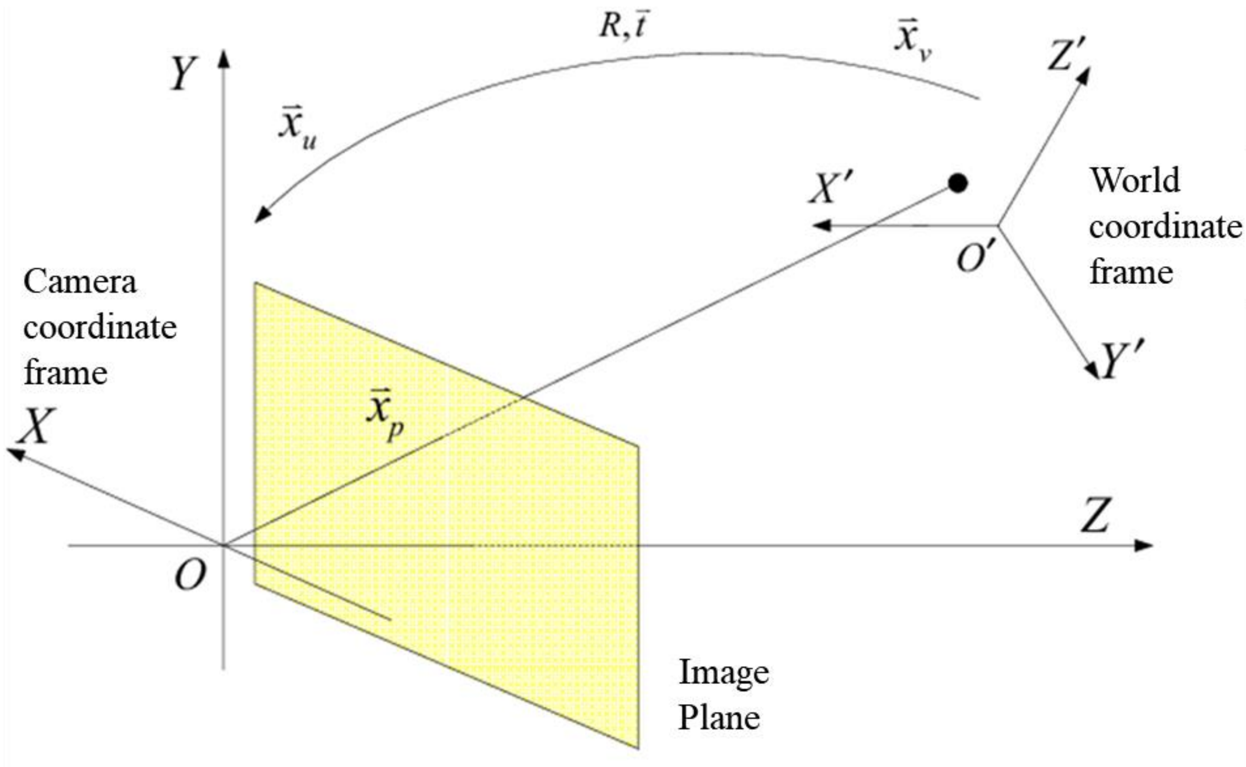


Figure 2.9 Image plane and camera coordinate frame with respect to the world coordinate frame [90].

For a multi-camera system, one camera is selected as the primary camera. Other cameras are calibrated with respect to the primary camera. The rigid transformation that maps points in the primary camera frame to the secondary camera frame is determined through an iterative approach, similar to intrinsic calibration. This is termed the extrinsic calibration process and provides the extrinsic rotation matrix and translation vector for each camera [88].

Bundle Adjustment (BA) approaches are the current gold standard for extrinsic parameter optimization and are a necessary final step when calibrating a multi-camera system [89]. BA finds image plane point correspondences between the primary and secondary camera. The primary is selected as the world origin and the transform that maps the points in the primary camera frame to the secondary camera frame is calculated. With a collection of image points, from multiple views of the calibration pattern, the extrinsic parameters between two cameras can be formulated as an optimization problem where the pixel reprojection error is minimized [88]. In this procedure, common practice keeps the intrinsic parameters as constants and iteratively adjusts the rotation matrix and translation vector parameters to reduce the dimensionality of the optimization problem.

Once a satisfactory solution for extrinsic parameter optimization is found, 3D points can be calculated through a triangulation process. Currently, extrinsic parameters (R and t) for each camera map world frame points to camera frame points, and intrinsic parameters (K and C_d) map camera frame points to image plane points. Triangulation uses the calibration parameters to find camera ray intersections that correspond to a point in the 3D world frame [90]. In a perfect system, the set of rays from each camera would intersect at a single 3D point; however, this may not occur due to measurement noise and errors in intrinsic and extrinsic calibration. Therefore, a naïve approach to triangulation that finds the intersection of such rays would not have a solution. More robust approaches for 3D point triangulation include midpoint calculation [91] and non-linear optimization approaches such as BA [89]. Figure 2.10 displays a calibrated multi-camera system and the desired intersections to be found during triangulation.

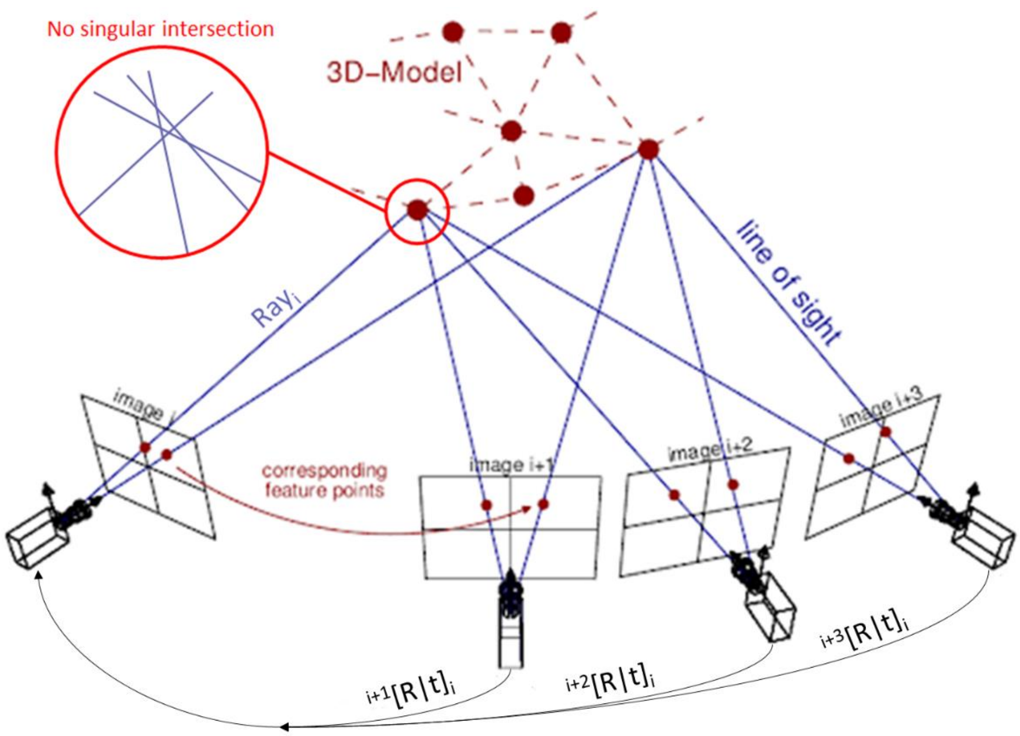


Figure 2.10 Array of cameras in space showing their corresponding rays and the rigid transformation between cameras.

For robust and accurate triangulation, a random sample consensus (RANSAC) BA approach that provides an optimal 3D point while being unaffected by outliers is ideal [89], [90]. The BA process fixes the intrinsic and extrinsic camera parameters as constants and iteratively selects 3D points that minimize the reprojection error in all views of the multi-camera system. RANSAC is a method used to reduce noise in measurements and aims to find the parameters that provide the greatest number of inliers in the given dataset [92]. Inliers are counted as points that provide a reprojection error below a desired threshold.

An optimized camera calibration procedure is necessary for accurate 3D reconstruction. Current gold standard approaches involve non-linear optimization algorithms, such as Levenberg-Marquardt, to obtain a robust solution for intrinsic and extrinsic camera parameters. RANSAC is a useful tool that can be employed at each stage to reduce outlier effects on the set of calibration image points. BA is important for calibrating multi-camera systems because BA combines information from all views in the system to find optimal extrinsic parameters that define all camera poses in the world frame.

2.7 Conclusion

Popular methods for PCS human motion analysis have successfully extracted stride parameters and body segment information from participants. However, these approaches affix sensors to the participant, or the participant directly interacts with a sensing device. This leads to poor acceptance in facilities without dedicated lab space and staff and requires testing to be planned in advance. PCLS have shown reasonable results that allow non-invasive data collection. PCLS could be deployed in a passive manner, allowing data capture as needed. With current advances in AI pose inference models and edge-compute devices, a PCLS could be designed to provide movement data with minimal human resource requirements. Using optimal computer vision algorithms for calibration and triangulation, AI models can provide accurate 2D and 3D data for clinically useful information in healthcare and long-term care.

3 Artificial Intelligence Model Selection

3.1 Overview

This chapter discusses artificial intelligence model selection for semantic keypoint inference when focusing on clinical movements and keypoint quality. A model is selected based on performance relating to clinically relevant movements used in physical assessments. This chapter addresses Objective 1 of this research.

The contents of this chapter were published as an IEEE conference paper.

F. Zhang, P. Juneau, C. McGuirk, A. Tu, K. Cheung, N. Baddour, and E. D. Lemaire, "Comparison of OpenPose and HyperPose artificial intelligence models for analysis of hand-held smartphone videos," in Proceedings of the 16th IEEE International Symposium on Medical Measurements and Applications, Neuchatel, Switzerland June 2021.

As third author, contributions included performing inference with AI models on batched videos, data processing, output rendering, data analysis, and co-writing the manuscript.

3.2 Abstract

Movement assessments are invaluable in clinical practice. However, the feasibility of in-person evaluation has been greatly affected due to the COVID-19 pandemic. To overcome this barrier, a virtual assessment system using AI and patient provided videos is needed. AI models for pose inference have produced viable results for identifying a person's joint centers. Identifying AI models for pose inference that provide clinically meaningful results is important for designing a virtual motion assessment tool. This study aims to evaluate the clinical usefulness of two popular pose inference models, OpenPose and HyperPose. Videos were recorded by two physicians, who independently performed movements they deemed clinically relevant. Keypoint skeletons were generated and manually inspected frame-by-frame to determine which model produced higher-quality pose inferences. OpenPose produced significantly better scores than HyperPose when comparing within videos ($p<0.001$). Right ankle and right wrist had the poorest performances. Best-practices to be used in the future design of a virtual motion assessment tool are required to improve video "AI-friendliness".

3.3 Introduction

Remote patient assessment is a rapidly growing area of interest for clinicians given challenges faced when delivering healthcare over large geographic regions [1], [2], [3]. With the advent of COVID-19 travel and contact restrictions, the need for this type of assessment has greatly increased. In particular, recent advancements in AI allow for assessment of gross extremity movements without the need for marker sets and expertise for data collection and processing. With the ubiquity of smartphones with high quality video, patients can record and submit movement videos that may be analysed by AI to aid clinical management.

Clinical motion assessment typically involves gait analysis, in addition to specific upper and lower extremity movements. Clinicians can use these evaluations to obtain qualitative and quantitative data relating to patient function. Evaluation tools such as the Shriners Hospital Upper Extremity Evaluation (SHUEE) are commonly used to assess a patient's motion and functional ability [4]. While these techniques provide a useful tool to aid in clinical assessment, they require an experienced physician or therapist to perform an in-person assessment and administer the test.

Recent advancements in AI techniques have given rise to human pose estimation tools, commonly referred to as pose machines [5]. Pose machines are built on convolutional neural network (CNN) architecture and utilize images to make inferences on where people may be in an image [5]. An open-source AI model called OpenPose has shown state-of-the-art results when detecting keypoints that relate to a person's joint centers [6]. These 2-dimensional (2D) points combine to create a skeleton model of the person of interest. However, AI models such as OpenPose can require a large amount of computational overhead that can hinder easy deployment [6]. Another model called HyperPose employs a less computationally heavy version of OpenPose that also optimizes various GPU processes [7]. HyperPose provides an optimal way of performing inference by utilizing NVIDIA's CUDA architecture and NVIDIA's TensorRT deep learning optimization library [7].

These pose machines can perform inference on images or videos provided by the user. OpenPose has successfully been used to assess patient posture during bed rest [8]. OpenPose has also been used to accurately track a dancer's joints for biomechanical analysis [9]. In both studies, low-resolution videos were used to identify keypoints on the participant. Since smartphones with high-quality cameras have become universal in our day-to-day lives, smartphone video could be used to collect clinical data remotely [10] [11]. Thus, a patient could be instructed to perform desired relevant movements while someone records the activity. The video can be processed by the AI model to aid in a remote assessment.

However, these models have yet to have their clinical usefulness evaluated. This study evaluated OpenPose BODY25 and HyperPose COCOv2 identifications of clinically relevant keypoints from smartphone collected video. This provides a basis for creating a remote assessment AI tool that clinicians may use for analysing gross physical movements. With these AI models, an invaluable tool for remote clinical assessment could be created to allow clinicians to easily assess and evaluate a patient's mobility status.

3.4 Methods

3.4.1 Smartphone Video Procurement

The smartphone videos procured for this study were recorded by physicians experienced in assessing and treating movement disorders. The physicians recorded themselves performing movements that they deemed clinically relevant when assessing patients with both upper and lower extremity conditions. Videos were recorded with an iPhone XS (2018) and iPhone XR (2018). Videos from the phones were down sampled to 1280 by 720 pixels for the iPhone XS and 568 by 320 pixels (30 fps) for the iPhone XR to allow for quick file transfer. A test set of six videos that covered a variety of movements were selected to evaluate the AI models. Videos were recorded at two meters from the person for standing movements and five meters for walking. The videos were converted to a normalized encoding format (MP4, H264) using the FFmpeg library prior to being passed to the AI models.

3.4.2 Data Acquisition and Preprocessing

Videos were passed to OpenPose BODY25 and HyperPose COCOv2. OpenPose BODY25 is the highest accuracy 25 keypoint model available from the OpenPose opensource models. HyperPose COCOv2 is an optimized version of an earlier model developed by the authors of OpenPose. The models were both deployed using an NVIDIA Jetson AGX Xavier high performance computing unit. Processing time for each video was noted. The output 2D pixel coordinates were stored for post-processing and rendering.

Both AI models output a per-detection score describing detection accuracy. Video keypoint data from each model were refined by deleting keypoints that were below a desired score threshold (10%). The data were then re-interpolated using a cubic spline approach with an interpolation limit of five missed keypoints. Data were filtered using a zero-phase second-order low-pass Butterworth filter with a cut-off frequency of 12Hz.

To determine the low-pass filter frequency, a pilot dataset was created and analysed. Videos of three movements (forearm supination/pronation, shoulder abduction, hip flexion) were recorded in the anterior/posterior plane and independently evaluated by two reviewers blinded to the filtering parameters. An unfiltered dataset was also used during the blinded evaluation as a control data set. OpenPose was used to process the keypoints for this pilot evaluation, to reduce the number of test variables. The pilot test used the evaluation method in Section C - Data Analysis. After evaluation, 12Hz filtering resulted in the best performance and was selected as the filtering frequency for the remainder of the analysis.

Refined detection data were rendered as a keypoint skeleton on top of the corresponding videos. To aid visual analysis, overlapping keypoints were redrawn as crosses and a note was appended to the frame. The OpenPose model outputs a 25-keypoint skeleton while HyperPose outputs an 18-keypoint skeleton. Figure 3.1 shows an example of how OpenPose and HyperPose keypoint skeletons are drawn. The models have an overlap of 12-keypoints that represent all major joints. Figure 3.2 shows a comparison of OpenPose and HyperPose skeletons applied to 3 sample movements: shoulder abduction, walking, and knee flexion. The videos with refined keypoint skeletons were given to reviewers for analysis.

3.4.3 Data Analysis

Two independent reviewers compared OpenPose and HyperPose keypoint skeletons. Keypoint quality was evaluated frame-by-frame using a 4-point scale:

1. Keypoint is absent.
2. Keypoint is outside the acceptable joint range, and/or keypoint is not on the body.
3. Keypoint is not on joint center but is within acceptable joint range.
4. Keypoint is directly on joint center.

"Acceptable joint range" was defined as a circular range whose diameter is no larger than the widest portion of the joint in that frame (Figure 3.3). Keypoints are still generated when a body segment is obstructed if the AI model is able to infer their position. If a joint center was obstructed from view by another body segment or clothing, but a keypoint was present, an evaluation was made using the reviewer's best judgement. Quality was calculated by taking the mean of a keypoint's score over all frames.

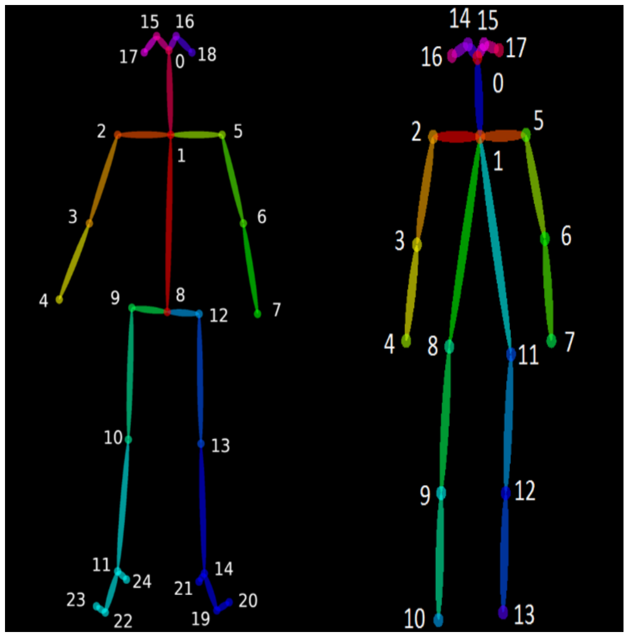
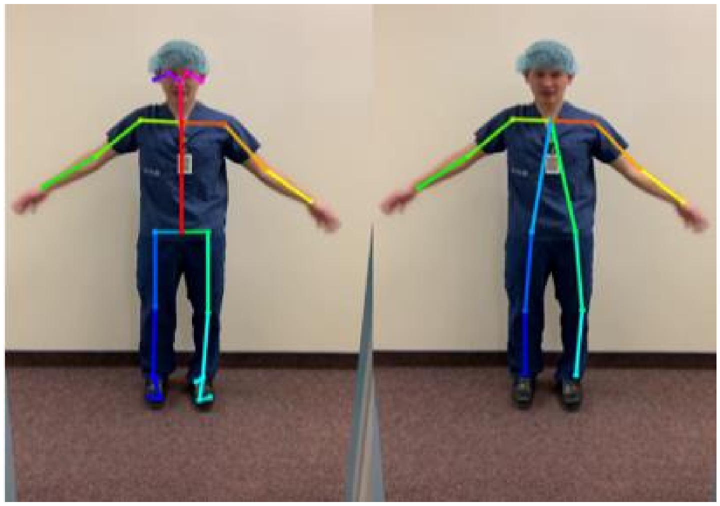
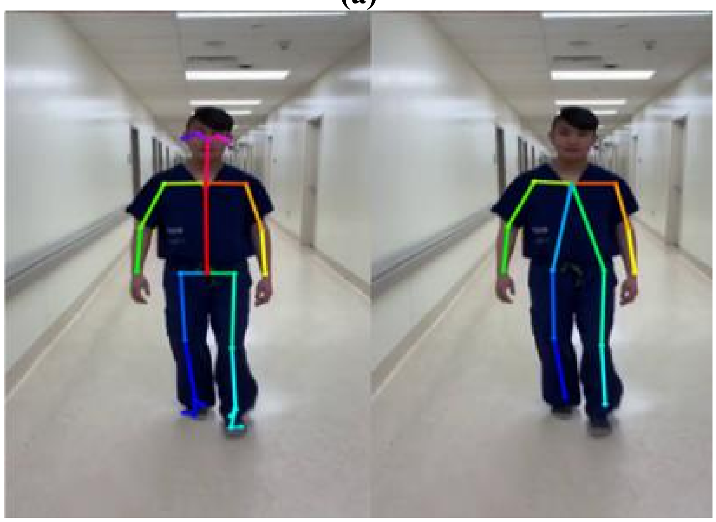


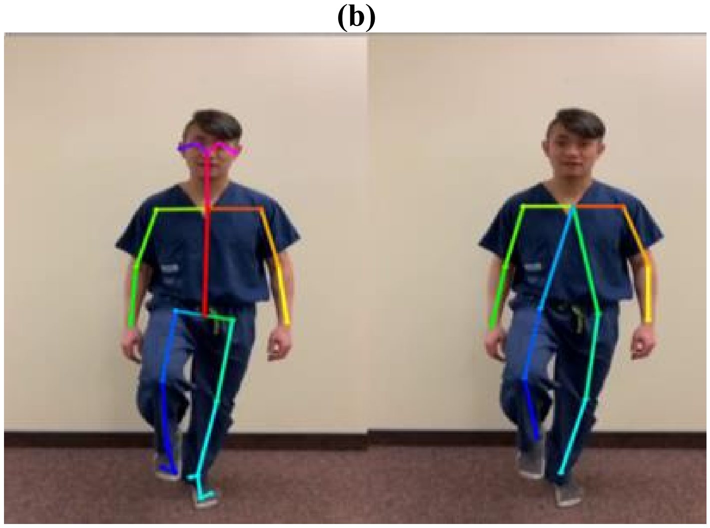
Figure 3.1 Keypoint skeletons from OpenPose BODY25 (left) and HyperPose COCOv2 (right). OpenPose BODY25 contains keypoints for the heels, big toes, small toes, and mid-hip that HyperPose COCOv2 does not contain.



(a)



(b)



(c)

Figure 3.2 Keypoint skeletons from OpenPose BODY25 (left) and HyperPose COCOv2 (right) applied to three sample movements: A) shoulder abduction B) walking and C) knee flexion.

A score of 1 or 2 was considered “unacceptable” and a score of 3 or 4 was considered “acceptable”. For this analysis, right refers to the participants’ anatomical right side and left refers to the participants’ anatomical left side.

Video scores were manually recorded and analysed using R (4.0.2) and base statistical functions. Comparisons between OpenPose and HyperPose keypoint skeletons were done on a pairwise basis (i.e., within each video and reviewer) using the Wilcoxon signed-rank exact test.

3.5 Validation

In total, 2,201 frames were reviewed for quality. Six movements were analysed: walking, shoulder abduction, shoulder external rotation, forearm supination/pronation, knee extension, and hip flexion. All movements were recorded in the frontal plane, except for hip flexion that was recorded in the sagittal plane.

Two raters achieved a 94% agreement on keypoint scores of acceptable (i.e., scores of 3 or 4) or unacceptable (i.e., scores of 1 or 2). Table 3-1 displays the agreement between reviewers for acceptable and unacceptable ratings. Agreement for all scores across all trials was 57% (Table 3-2).

Table 3-1 Agreement of two raters of unacceptable vs. acceptable scores.

		Rater 1	
Rater 2	**Score**	**Unacceptable**	**Acceptable**
	Unacceptable	787	365
	Acceptable	448	24848

Figure 3.3 Example frame demonstrating keypoint rating and acceptable joint ranges. Dashed circles are centered on observed anatomical joint centers, with diameter equal to the observed joint width. 1 = absent (arrows pointing to approximate locations of occluded left wrist, left elbow). 2 = outside of acceptable range or off body (right heel), 3 = within the acceptable joint range (left heel), 4 = directly on joint center (right knee).

Both models produced acceptable keypoint skeletons for the majority of joint centers. Thus, to sufficiently compare OpenPose and HyperPose keypoint skeletons, we examined the three keypoints with the lowest mean keypoint rating for all videos. The three keypoints with the lowest mean across all videos in OpenPose were the right knee (3.27 ± 0.27), right ankle (3.30 ± 0.44), and left knee (3.35 ± 0.42). For HyperPose, these keypoints were the right ankle (2.72 ± 0.84), right wrist (2.79 ± 0.86), and right knee (2.97 ± 0.19).

For a given video (i.e., movement, pose machine, and reviewer), the mean score of each keypoint was calculated (i.e., the sum of the ratings divided by the number of frames). Then, the mean of the lowest three keypoint results from the previous step was taken. We termed this latter number the “lowest-three-mean” score and used it to summarize individual videos in subsequent analyses.

In general, OpenPose outperformed HyperPose (**Table 3-3**). The mean lowest-three-mean scores across the six videos were 3.11 ± 0.25 for OpenPose and 2.68 ± 0.57 for HyperPose. In addition, OpenPose produced significantly better scores than HyperPose when comparing within videos ($p<0.001$).

We observed that video scores were affected by low-quality keypoints irrelevant to the extremity being examined (i.e., low quality wrist keypoints in a hip flexion video). Thus, we repeated the above analyses considering only wrist, elbow, and shoulder keypoints for upper-extremity movements (i.e., shoulder abduction, shoulder external rotation, forearm supination/pronation) and ankle, knee, and hip keypoints for lower-extremity movements (i.e., knee extension, and hip flexion). Walking was excluded for this analysis since both upper- and lower-extremity keypoints are relevant to clinically evaluate walking.

Table 3-2 Agreement of two raters on the 4-point rating scale.

	Rater 1				
	Score	**1**	**2**	**3**	**4**
Rater 2	**1**	422	0	0	0
	2	0	365	352	13
	3	0	396	8263	1751
	4	0	52	8833	6001

The upper-extremity keypoint with the lowest mean across upper-extremity movements was the left elbow, for both OpenPose (3.36 ± 0.39) and HyperPose (2.98 ± 0.18). The left ankle was the lower extremity keypoint with the lowest mean for both OpenPose (3.14 ± 0.17) and HyperPose (3.09 ± 0.07). Interestingly, when comparing OpenPose and HyperPose using joint keypoint means, the models were not significantly different ($p=0.91$).

The mean lowest-three-mean scores for OpenPose (3.25 ± 0.23) and HyperPose (3.03 ± 0.11) when keypoint skeletons were limited to extremities relevant to the motion performed were significantly different ($p<0.0001$).

OpenPose processed video slower than HyperPose (i.e., 1.86 versus 2.83 fps). Videos averaged 183.5 frames in length. This would result in an average processing time of 98.66 seconds for OpenPose and 64.84 seconds for HyperPose. HyperPose was 65.7% faster than OpenPose, resulting in an average processing time difference of 34 seconds per video.

Table 3-3 Lowest-three-mean score by model and rater.

	Movement	OpenPose lowest-three-mean score	HyperPose lowest-three-mean score
Rater 1	**Walking****	2.98 ± 0.03	2.52 ± 0.02
	Hip flexion	2.71 ± 0.25	1.61 ± 1.06
	Shoulder abduction*	3.56 ± 0.06	3.00 ± 0.00
	Shoulder external rotation	3.05 ± 0.08	2.84 ± 0.14
	Forearm supination/ pronation	3.00 ± 0.00	2.95 ± 0.09
	Knee extension	3.08 ± 0.06	3.02 ± 0.03
Rater 2	**Walking***	3.51 ± 0.03	2.78 ± 0.12
	Hip flexion	2.57 ± 0.01	1.53 ± 0.92
	Shoulder abduction	3.00 ± 0.00	3.04 ± 0.11
	Shoulder external rotation	3.49 ± 0.17	2.921 ± 0.17
	Forearm supination/ pronation	3.10 ± 0.13	2.985 ± 0.02
	Knee extension	3.30 ± 0.07	3.10 ± 0.06

*** p<0.01 ** p<0.001**

3.6 Discussion

This study evaluated the quality of AI-generated keypoint locations from clinically relevant movements. Correct identification of joint centers during movement is an important step in evaluating a patient’s movement ability. For example, keypoints could be analysed to determine range of motion differences between the left and right side of the body and evaluate a patient’s change in ability over time. Obtaining accurate data from these videos is crucial for a virtual assessment. Only then can clinicians be confident in the ability of AI to aid in patient assessment.

This study also demonstrated that HyperPose (COCOv2) processes videos faster than OpenPose (BODY25). This processing time difference is important when analysing long videos or large datasets, where up to an hour per video could be needed for the AI component. Faster versions of HyperPose can perform inference at up to 60.0 frames per second [6], though the results would be even less accurate than the COCOv2 model used in this study. Researchers reviewing large video files or completing real-time evaluations of movement and gait data might consider using HyperPose over OpenPose for this reason. However, this must be weighed against the reduction in quality observed with HyperPose, as compared with OpenPose. Future research in real-time processing could evaluate the mobile version of OpenPose [6] and Google ML Kit Pose detection [93]. This would allow for reasonable processing time performance on a smartphone device.

The evaluation results show how AI systems, as well as human reviewers, can be challenged to correctly identify anatomical landmarks when the body segment is not clearly visible. This explains why keypoints on the participants left side, specifically the left elbow and ankle, had the lowest mean score when recorded in the sagittal plane (i.e., left side was occluded). A number of factors can affect the ability of both AI and humans to correctly label a joint center. For example, baggy clothing, poor lighting conditions, shaky hand-held video, and further distances from the person to the camera. These can also result in variability when human evaluators make a judgement between two scores. Potential options for improving inference, or "AI friendliness", can be separated into scene-based and software-based. Scene-based guidelines relate to the background, patient's clothing, and lighting and could include reminders to ensure that:

- Patient clothing has a contrast relative to the background.
- Lighting in the video is balanced, and the person is easily visible.
- No pictures or objects in the background that may be mis-detected as people.
- Use a tripod or other approach to orient the phone appropriately without camera movement (i.e., avoid handheld camera movement).

Software implementations that could improve video capture of the relevant movements include:

- Video stabilization to ensure that handheld videos are invariant to camera movement.
- Monitoring if the person is in frame for all portions of the relevant movement.
- Management of the video compression relative to available bandwidth, such that the resolution provided to OpenPose and file transfer time is optimized.

For future applications, directions like these should be provided to participants to improve "AI friendliness".

3.7 Conclusion

This investigation determined that OpenPose (BODY25) provided better clinically relevant keypoint location labelling than HyperPose (COCOv2). However, quality was dependent on body segment visibility in the frame, reaffirming the need for "AI-friendly" videos. Future participants should be provided with a set of instructions and best-practices for recording videos for AI-assisted remote movement assessment. Also, a new video capture application could help guide the videographer to have the person appropriately in frame and verify quality before processing. Future applications include evaluating a larger collection of movements using OpenPose. COVID-19 has created or heightened existing barriers for patients, limiting the clinician's ability perform typical in person movement assessments. A successful smartphone-based system can enable patients to better communicate how they move to physicians. Such videos can improve clinical decision-making by providing quantitative movement information irrespective of the patient's physical location.

4 Smart Hallway Design

4.1 Overview

This chapter addresses objective 2 of this research. This chapter discusses the design of an AI-based marker-less human motion analysis system to be deployed in an institutional hallway. Various aspects of the design such as components, physical setup, time synchronization, and camera calibration are described and were validated.

4.2 Abstract

This research compared cameras for multi-sensor systems, peripheral component interconnect express (PCIe) cards for data streaming, high-speed non-volatile memory express (NVME) solid-state drive (SSD) storage options, and high-performance computing options for AI inference. Selected components were modelled using Blender (Blender Foundation, Blender 2.91) to determine optimal camera placement. Multi-camera synchronization and calibration are described and validated through physical tests.

Selected components for the final system included: NVIDIA Jetson AGX, Samsung 970 EVO Plus NVME SSD, StarTech PEXUSB3S44V peripheral card, FLIR BlackFly S USB3 cameras, and Fujinon 3MP lenses. Blender simulations determined an optimal four-camera arc layout, providing a 5 m by 2.4 m walking area and 29 m^3 total capture volume visible to all cameras. Synchronization testing showed that cameras captured images within 5 us. Camera intrinsic and system extrinsic parameters were calibrated to sub-pixel accuracy.

4.3 Introduction

Motion capture systems require a variety of components to work optimally in tandem to obtain accurate results. Motion capture systems can be generally classified as physical contact systems (PCS) or physical contact-less systems (PCLS). PCS requires a clinician or researcher to affix sensors to the participant or ensure that the participant contacts a sensing device. PCLS do not require the participant to contact any part of the system, thus removing the possibility that their natural gait could be affected [94]. PCLS also offer the ability to assess populations during their day-to-day routine. Thus, designing a system where the participant only needs to be within a virtual capture volume to perform motion analysis is highly desirable. This would not only allow for natural gait to be tracked, but also may allow for a security camera type tool for motion analysis at the data source. AI models for semantic keypoint location (i.e., locating a set of 2D coordinates, or keypoints, in a meaningful manner on a given image) have had acceptable results for detecting joint centers in 2D and could provide a basis for determining 3D points when using correspondences from multiple cameras.

PCLS systems typically use multi-camera arrays and AI models for pose inference. Cameras are normally RGB cameras or RGB-depth cameras, depending on the use case. Component selection and setup for multi-camera systems are often overlooked and often only system design and accuracy results are reported. This creates a gap for facilities or labs interested in deploying their own multi-camera arrays but without the knowledge to select appropriate equipment. Compounding this issue is the need for a wide variety of components that are required to work together to obtain even passable results.

For this type of PCLS to be deployed in an institutional setting and be integrated with ease, the system must be non-contact (i.e., direct contact with the person of interest is not needed) and not hinder any processes at the institution [95], [96]. An ideal area for this type of motion analysis system is in an institutional hallway where patients, residents, staff, and visitors consistently move through [97]. By designing a system that does not obstruct other hospital or elderly care residence processes, a highly useful motion analysis system could be created.

To extract the most accurate motion data from a marker-less setup, optimal camera positioning is essential [98]. Variables such as the number of cameras, camera pose, and camera parameters should be optimized based on the desired output quality and processing time of the final system. In this investigation, geometrical system setup was simulated along with real world testing of camera synchronization to perform preliminary analysis of the marker-less motion capture system.

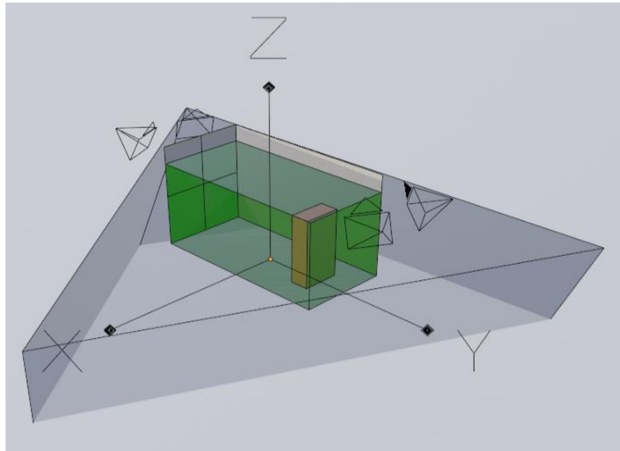
A final multi-camera calibration process is also outlined, describing how to achieve sub-pixel reprojection error for intrinsic and extrinsic parameters. Camera calibration is a crucial step for marker-less multi-camera systems because 3D information must be determined through triangulation processes that rely heavily on the calibration results.

4.4 Simulation of Geometrical Setup and Capture Volume

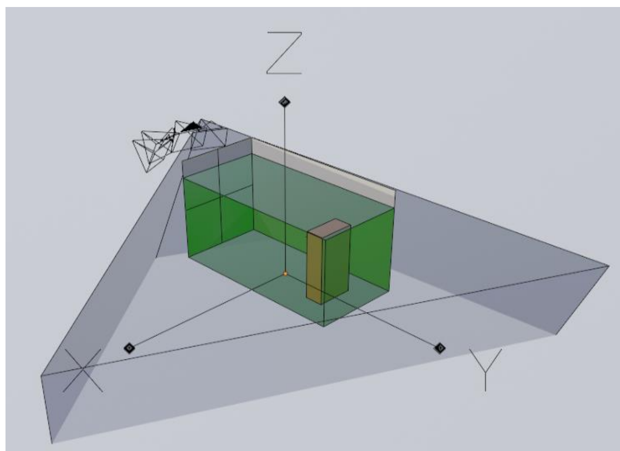
This section discusses the physical setup of three multi-camera arrays optimally positioned in an institutional hallway. Cameras are modelled in 3D space to determine the volumetric coverage achievable with four and eight cameras. For this research, three different arrays of FLIR BlackFly® S USB3 (BFS-U3-16S2C-CS) machine vision cameras with Fujinon 3 MP Varifocal Lenses (YV4.3X2.8SA-2) were simulated to determine the volumetric coverage and camera positioning.

4.4.1 Methods

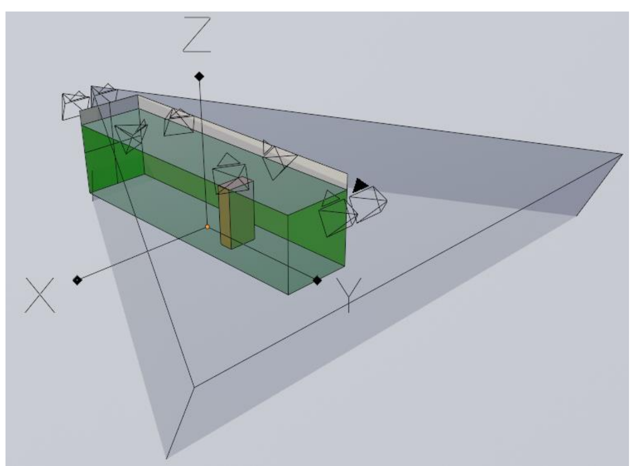
Camera FoV during simulation was 70°×52.5°, based on lens specifications [99]. In this simulation, an arc layout and corner layout were evaluated. Setup 1 had four cameras, with one camera placed at each corner of the simulated hallway capture volume. Setup 2 had four cameras placed in an arc at one end of the simulated hallway. For each setup, three iterations were performed by adjusting camera poses. Both setup 1 and setup 2 aimed to capture a 5m×2.4m×2.5m volume in the simulated hallway. Setup 3 had eight cameras placed equidistant around a 10m×2.4m×2.5 capture volume, to test how the system could scale to a greater number of cameras. For each setup, camera poses were varied to maximize FoV overlap and the percentage of useable volume. Useable volume was defined as the desired walking distance length (5m or 10m), the width of the institutional hallway (2.4m), and a conservative estimate of typical participant height (2.5m). Figure 4.1 displays an example of the four-camera corner, four camera arc, and eight camera layouts simulated in Blender (Blender Foundation, Blender 2.91).



(a)



(b)



(c)

Figure 4.1 Simulated camera layouts created in Blender. a) Four-camera corner, b) Four-camera arc, c) Eight-camera perimeter.

Properties of the useable capture volume were calculated by determining the intersection of each camera's respective FoV within the simulated hallway. A volume mesh describing the space in which all cameras have a view of the scene is formed from these intersections. Using Blender's Boolean intersection method, several 3D meshes were produced and compared. The capture volume meshes generated from each layout were compared to determine a camera layout that provided desirable features in the context of an institutional hallway setting. Individual camera capture volumes are defined in (4.1) as

$$CV_i = \frac{\left|\overrightarrow{A_\iota D_\iota} \times \overrightarrow{AB_i}\right| \cdot \overrightarrow{P_\iota A_\iota} \cdot \left(\overrightarrow{A_\iota D_\iota} \times \overrightarrow{A_\iota B_\iota}\right)}{3\left|\overrightarrow{A_\iota D_\iota} \times \overrightarrow{A_\iota B_\iota}\right|} \tag{4.1}$$

with CV as the camera capture volume and i indicating the camera number in a multi-camera system. Vectors $\left[\overrightarrow{A_\iota B_\iota},\ \overrightarrow{AD_i},\ \overrightarrow{P_\iota A_\iota}\right]$ are defined in Figure 4.2.

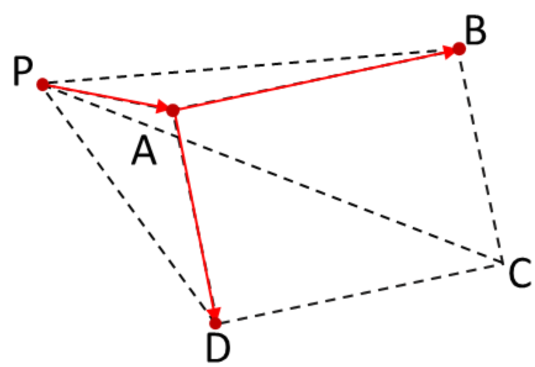


Figure 4.2 Vector definition of a single camera capture volume used to determine FoV intersections with multiple cameras.

The intersection of the camera volumes is defined in (4.2) as

$$\sum_{i=1}^{n} CV_1 \bigcap CV_{i+1} = \sum_{i=1}^{n} \{\vec{v} \mid \vec{v} \in CV_1 \ and \ \vec{v} \in CV_{i+1}\} \tag{4.2}$$

where CV_1 defines the capture volume of camera 1, CV_{i+1} defines the capture volumes of the other cameras in the multi-camera system, and $\vec{v}$ defines the set of vectors that define each camera capture volume.

4.4.2 Validation

The total capture volume for each of the three setups and the total ground coverage areas are presented in Table 4-1. The capture volume meshes (Figure 4.3) are shown for each layout with varying camera pitch angles. The four-camera arc layout provided more coverage of the simulated institutional hallway compared to the four-camera corner layout. For the four-camera corner layout, some stereo-cameras pairs were too far apart to reliably achieve an accurate calibration. Also, 3D reconstruction accuracy for a given stereo-camera pair is dependent on the incidence angle between the two cameras [100]. The four-camera corner layout requires some pairs of stereo-cameras to exceed the desirable incidence angle, which could negatively affect the final 3D reconstruction accuracy. The eight-camera perimeter layout provided the best coverage overall (i.e., total capture volume, ground coverage area). Increasing the number of cameras also improves the likelihood of more accurate 3D reconstruction. An optimization procedure could be performed to determine camera poses and placement based on capture volume, ground coverage, and the number of cameras to further improve these results.

Table 4-1. Capture volume and ground coverage area from the simulated camera layouts. R is the range of camera poses in the X, Y, and Z-axis for each camera layout.

Number of Cameras	Layout	Pitch Angle Variation (°)	Total Capture Volume (m^3)	Ground Coverage Area (m^2)
4	**Arc**	$R_x = [60° : 65°]$ $R_y = 0°$ $R_z = \pm [12.5°, 25.0°]$	34 ($R_x = 65°$) 31 ($R_x = 62.5°$) 29 ($R_x = 60°$)	10 ($R_x = 65°$) 11 ($R_x = 62.5°$) 11 ($R_x = 60°$)
4	**Corner**	$R_x = [60° : 65°]$ $R_y = 0°$ $R_z = \pm 25.0°$	33 ($R_x = 65°$) 30 ($R_x = 62.5°$) 28 ($R_x = 60°$)	8 ($R_x = 65°$) 10 ($R_x = 62.5°$) 11 ($R_x = 60°$)
8	**Perimeter**	$R_x = [62.5°, 65°]$ $R_y = 0°$ $R_z = \pm [12.5°, 25.0°]$	61 ($R_x = 62.5°$)	20 ($R_x = 62.5°$)

Meshes obtained using the intersection procedure provided visual confirmation of the expected capture volume in the simulated institutional hallway. Geometric features of the capture volume mesh could be used to obtain a desirable layout of the cameras and aid in camera positioning if certain features are desired. This may include maximizing the total volume, maximizing the camera capture volume overlap, or minimizing the angle of incidence between cameras. Figure 4.3 shows the meshes obtained from the four-camera corner, four-camera arc, and eight-camera perimeter layouts.

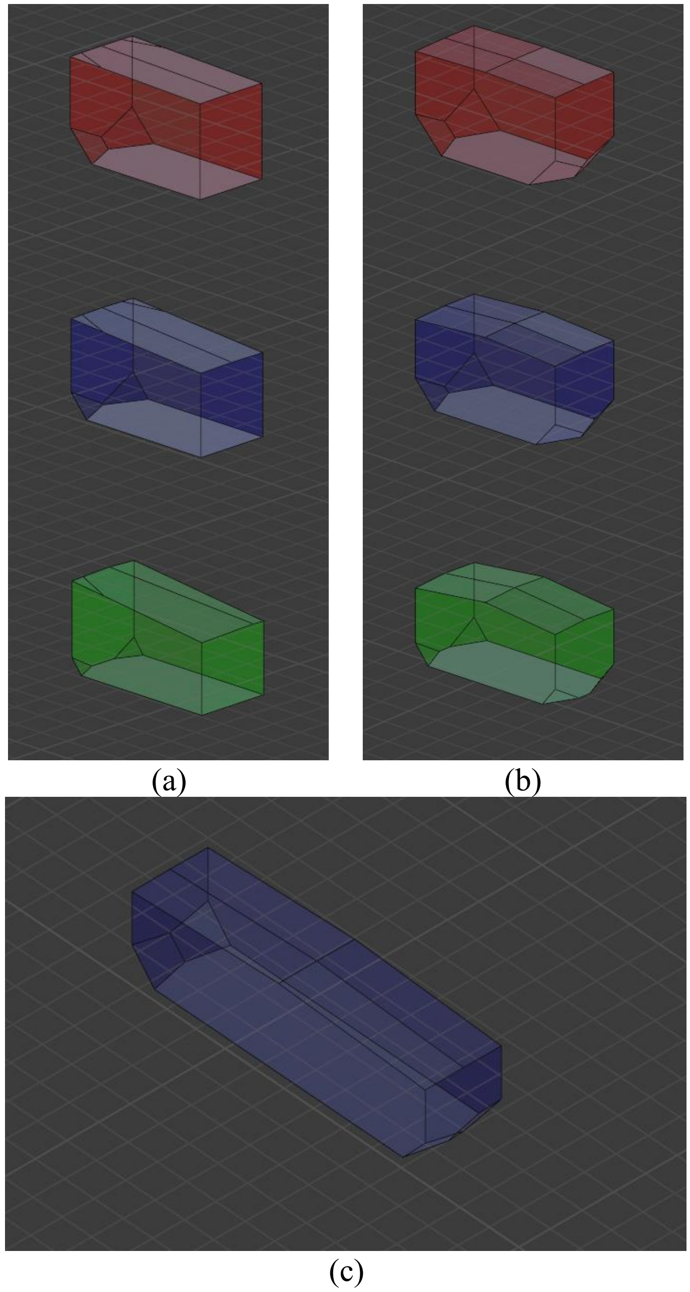


Figure 4.3 Capture volume meshes: a) Four-camera corner, b) Four-camera arc, c) Eight-camera perimeter.

4.4.3 Physical Setup

For multi-camera system validation, the four-camera arc layout (Figure 4.4) was selected based on the simulated capture volume and ground coverage results. Cameras were positioned and posed, using manual measurement tools, as close as possible to the simulated institutional hallway setting.

Figure 4.4 Physical setup of the four-camera arc layout. The top image slice shows the cameras arranged in the arc layout. The bottom image shows the virtual border of the hallway capture volume in red.

4.5 Smart Hallway Hardware

This research analysed system components to determine a suitable hardware design for the Smart Hallway. A variety of components are required to work in conjunction to have a Smart Hallway that provides useable 3D reconstruction and outcome measures. Components were selected based on bandwidth, interconnectivity, and output capabilities compared to desired precision and accuracy of the final system. Components selected were: FLIR BlackFly S USB3 cameras, StarTech PEXUSB3S44V PCIe card, Jetson NVIDIA AGX HPC unit, Samsung 970 EVO Plus NVME SSD.

4.5.1 Hardware Selection

Components were selected based on geometric and data transfer constraints. Geometric constraints were based on the institutional hallway simulations and the maximum cable lengths for each communication standard (USB, GiGE, etc.). Data transfer constraints were based on the desired multi-camera system performance in terms of resolution, pixel format, and frame capture rate. Bandwidth was calculated based on an RGB8 pixel format using the following equation,

$$BW = N_{Camera} f\left(W_{px} H_{px}\right) P_{format} \quad \textbf{(4.3)}$$

where BW is bandwidth, N_{Camera} is the number of cameras streaming data, f is the frame rate per second, $\left(W_{px} H_{px}\right)$ is the width and height resolution in pixels, and P_{format} is the pixel format.

Imaging devices compared were the GoPro Hero 7, Lorex 4K camera system, FLIR BlackFly S GigE, and FLIR BlackFly S USB3 cameras used for stereo reconstruction, synchronization, and data streaming quality. PCIe cards compared were the HighPoint RocketU 1344A, MosChip MCS9990, and the StarTech PEXUSB3S44V for high-speed and high-bandwidth data transfer. NVME SSD storage options compared were the Western Digital Black (WDS256G1X0C), Intel 660p (SSDPEKNW512G8X1), and the Samsung 970 EVO Plus (MZ-V7S500/AM) for write speed timing. High-performance computing (HPC) units compared were the Google Coral, Jetson TX2, and Jetson AGX Xavier Developer Kit for determining expected performance when running large-scale pose inference AI models. The selected components that best addressed the Smart Hallway requirements are detailed in (Table 4-2).

Table 4-2 Smart Hallway hardware selections detailing components and data transfer capabilities.

Component	Name	Description	Data Handling
Cameras	FLIR BlackFly S USB3 (BFS-U3-16S2C-CS)	Resolution: 1440×1080 Frame rate: 1 - 226 fps Sensor: Sony IMX273	Output: 280 MB/s (USB) Cameras: 4
Data Transfer Cables	USB-A with active extension cable	Length: 3m + 5m Format: 1×USB3.0	Bandwidth: 625 MB/s (USB) Cables: 4
PCIe Card	StarTech (PEXUSB3S44V)	Format: 4×USB3.0, 1×PCIe x4 Gen 2.0	Bandwidth: 4×625 MB/s (USB), 4 GB/s (PCIe) Ports: 4
HPC Unit	NVIDIA Jetson AGX Xavier	Format: 1×PCIe x8 Gen 4.0	Bandwidth: 16 GB/s (PCIe)
Data Storage	Samsung 970 EVO Plus (MZ-V7S500/AM)	Format: NVME M.2	Read/Write: 3.5 GB/s

4.5.2 Multi-Camera Hardware Synchronization Cable

This section discusses the multi-camera image capture synchronization method designed for this research to ensure temporally synchronized image sequences from the multi-camera array. A hardware synchronization method was selected for its reliability over software-based approaches [101]. For this multi-camera system, a low-profile synchronization cable using a primary camera as the trigger source was fabricated to reduce the footprint of the overall multi-camera array and its components.

A. Methods

The FLIR BlackFly S USB3 cameras all have a general-purpose input and output (GPIO) port that allows access to the camera auxiliary power input, auxiliary power ground, non-isolated input, and opto-isolated input. Camera software can be used to specify how GPIO will be used [102]. The hardware synchronization cable provided external power to the cameras while transferring the trigger signal produced by the primary camera. Cameras were powered externally to improve overall reliability and reduce the load on the NVIDIA Jetson AGX Xavier. The primary camera was formatted through software commands so that opto-isolated output produced a square wave as a function of the internal exposure time and selected frame rate. The secondary cameras were formatted similarly to the primary camera, except that the opto-isolated input was enabled to receive the primary camera's trigger signal. To produce the desired trigger signal at the primary cameras opto-isolated output, an external connection to a power source was required since the camera's 3.3V input was occupied by the external camera power supply. Figure 4.5 displays connections to the primary and secondary camera GPIO ports.

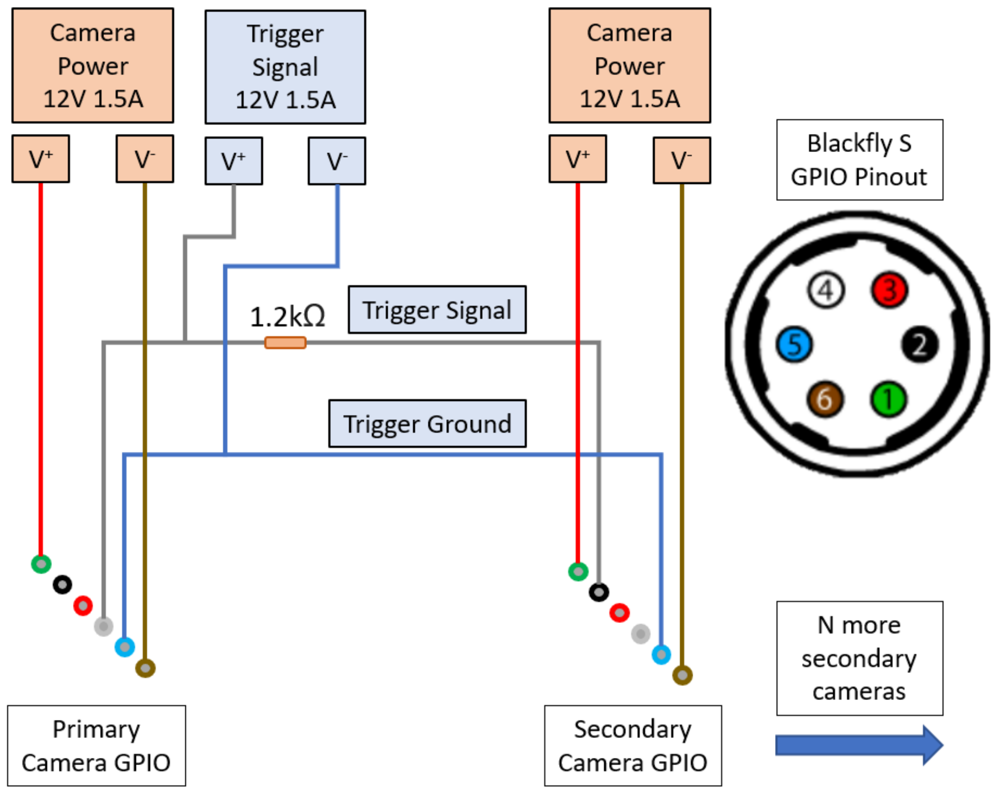


Figure 4.5 Hardware synchronization cable GPIO pin connections for primary and secondary cameras.

The cable was built longer than necessary to accommodate the distance between cameras and variety of camera array layouts tested. The distance between camera nodes was 7m and the connection to each camera was 0.75m, to allow for variability in placement. Due to the size of the cable and manufacturing capabilities, the external power connections were not consolidated into one connection and were not run alongside the trigger signal wiring. Thus, camera power supplies were spliced into the cable near each camera GPIO connector. This is detailed in Figure 4.6 where the overall cable layout is shown along with all connections.

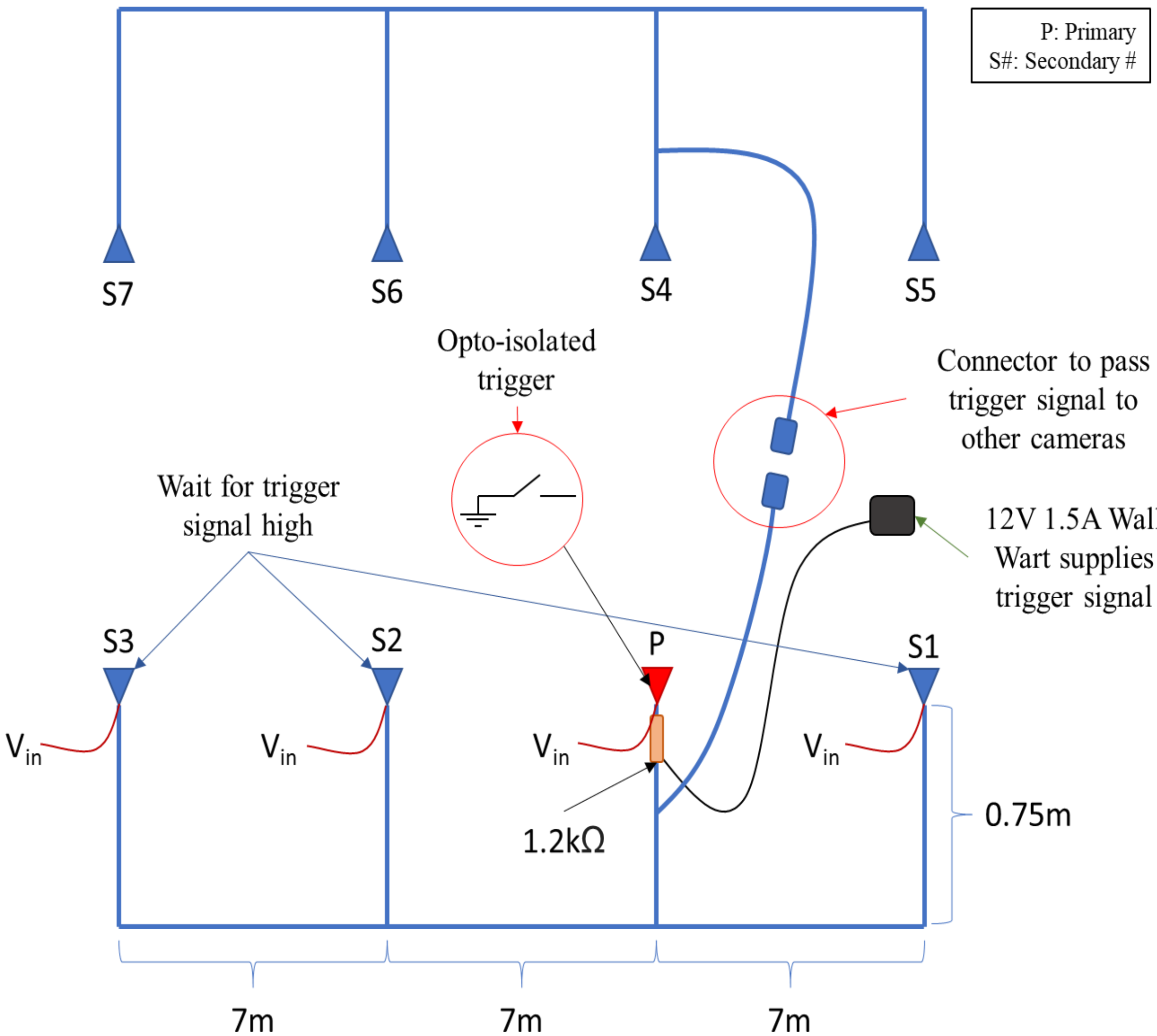


Figure 4.6 Multi-camera synchronization cable expanded to an eight-camera setup.

B. Validation

Preliminary validation of the synchronization cable was performed by using a multimeter to measure current while running the cameras. Based on the inner circuitry of the opto-isolated GPIO, a 1.5mA activation current was required at the LED to enable triggering [102] (opto-isolated output allows for a maximum current draw of 25mA). The values in Table 4-3 were measured using three different pull-up resistors at the primary camera.

Table 4-3 Current draw measurements while triggering multiple cameras. Activation current per-camera must be >1.5mA to enable triggering. Bolded values caused triggering to fail.

Pull-Up Resistor (kΩ)	Number of Cameras	Current Steady State (mA)	Current Active (mA)	Current Active (mA/Camera)	Trigger Success
2.4	2	4.73	3.01	1.55	Yes
	3	**4.74**	**4.23**	**1.41**	**No**
	4	**4.74**	**4.37**	**1.09**	**No**
1.2	2	9.06	3.56	1.78	Yes
	3	9.07	6.15	2.05	Yes
	4	9.08	7.84	1.96	Yes
0.6	6	18.10	11.88	1.98	Yes
	8	18.11	15.44	1.93	Yes

To ensure cameras were being triggered properly, an oscilloscope was used to measure voltage changes at the output of the primary and input of the secondary cameras. Ideally, the signal produced by the primary camera should be identical to the signal received by each of the secondary cameras, without lag to ensure that cameras are triggered at the same instant. Figure 4.7 shows the trigger wave, including the trigger and exposure portion of the signal.

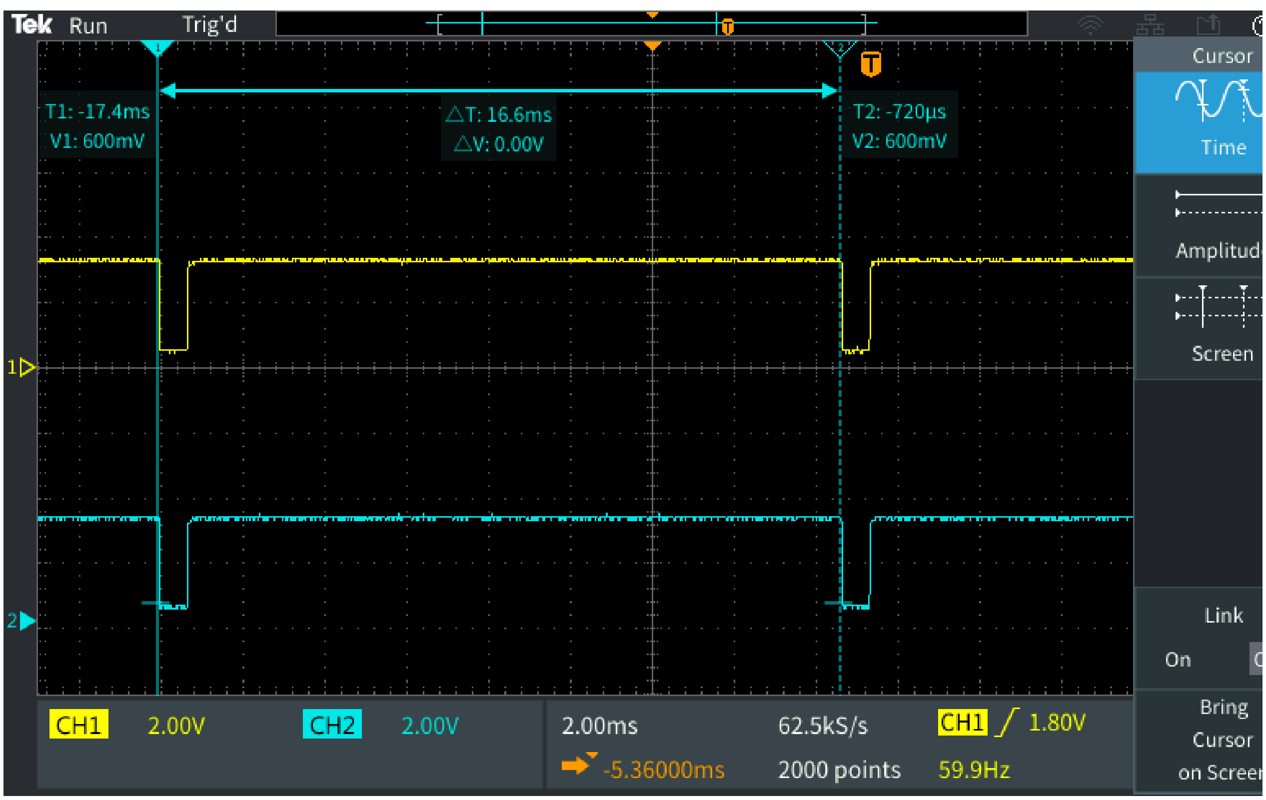


Figure 4.7 Trigger signal produced by the primary (yellow) and received by the secondary (blue) cameras.

Once the trigger signal was validated, the lag between primary and secondary cameras was measured by capturing images (60 fps) of a millisecond clock (monitor refresh rate 60 fps). A test using four cameras was performed and after 167 seconds of image capture (10,000 frames) the final frame from each camera was compared. Figure 4.8 displays the setup and an example of the camera synchronization.

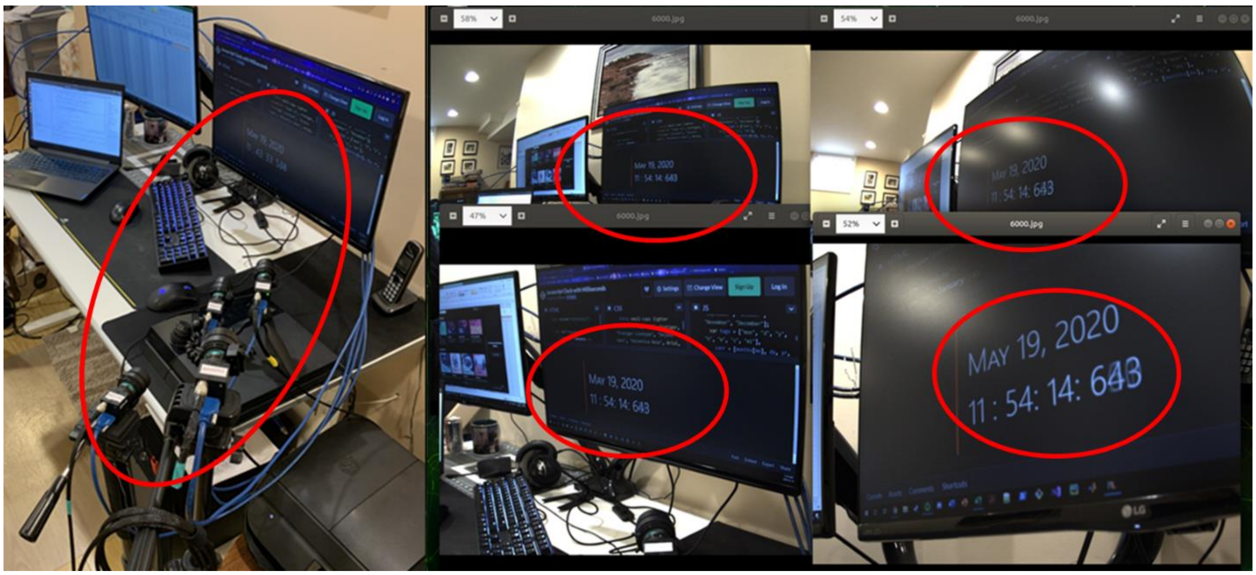


Figure 4.8 Synchronization test layout and validation images for a four-camera setup capturing at 60 fps. All cameras show the same frame of the millisecond clock after 10,000 images.

To further validate camera synchronization, image time stamps were converted to a standardized CPU time on the Jetson NVIDIA AGX Xavier. Three tests using four cameras capturing 10,000 synchronized images were performed to determine the robustness of the trigger synchronization cable. Table 4-4 displays the average difference between the measured time stamp and the target 60 fps (16667 µs/frame).

Table 4-4 Timing differences between camera images during triggered image capture. Camera 20023229 (bolded) is the primary camera sending a trigger signal to each secondary camera. Timings were measured in nanoseconds and converted to microseconds.

	Image Timing Characteristics, Δ (µs)				
Camera	**µ**	**σ**	**σ^2**	**Max Δ**	**Min Δ**
20023229	**-0.00006**	**0.39491**	**0.15595**	**4**	**-3**
20010192	-0.00011	0.39193	0.15361	5	-5
20010190	0.03866	0.56200	0.31585	5	-4
20010189	0.02373	0.49665	0.24666	4	-5

Table 4-4 shows the stability of the camera synchronization over 10,000 images and how closely in time the individual images are captured (on average). Camera positions relative to Figure 4.6 are P: 20023229, S1: 20010192, S2: 20010189, and S3: 20010190. The cameras drift at maximum 5 µs from the desired 60 fps and maintain a narrow distribution (Figure 4.9).

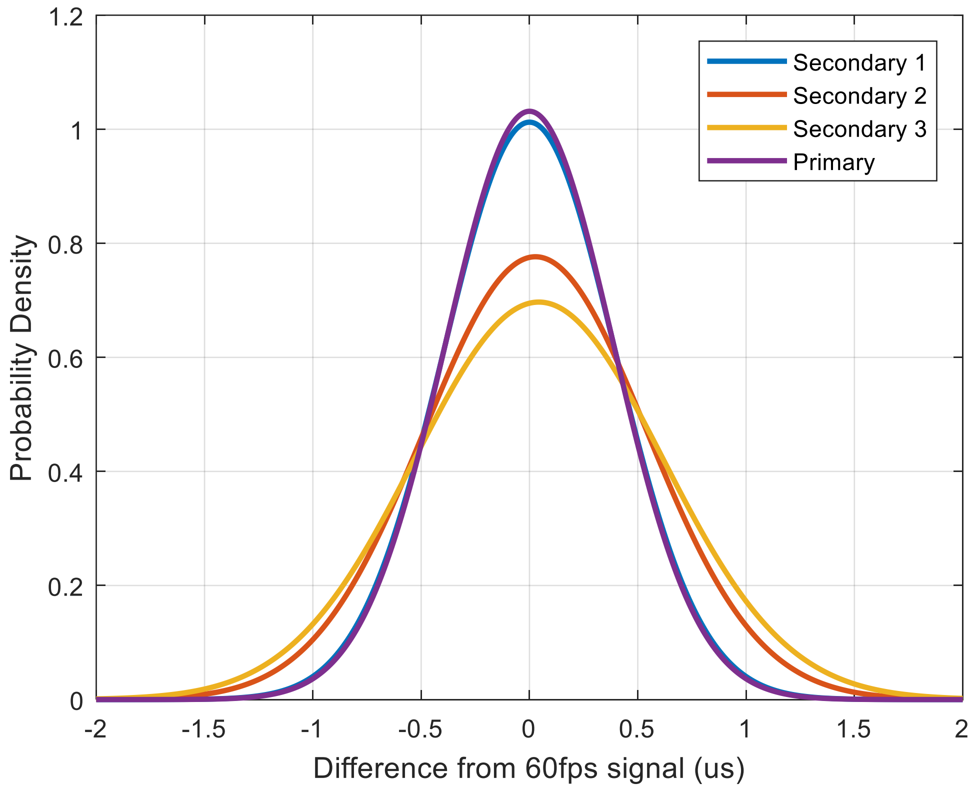


Figure 4.9 Distributions of camera timing differences relative to the desired frame rate.

Figure 4.9 displays the narrow distribution of each camera's synchronized image capture. Based on the standard deviations from Table 4-4, on average 99.7% of the images captured remain within 2 µs of the desired 60 fps capture rate.

4.6 Multi-Camera Calibration

This section discusses the calibration procedure for a multi-camera system and how to achieve optimal calibration results. A combination of state-of-the-art techniques for calibration pattern detection, outlier removal, and optimization were used in conjunction to obtain sub-pixel error. For this research, a calibration process for a multi-camera system in a large capture volume is designed and validated using existing methods tailored to this research.

4.6.1 Methods

Calibration of individual camera intrinsic parameters and the multi-camera systems extrinsic parameters was accomplished using a pattern calibration approach. The calibration pattern was an 8×7 ChArUCo board with 4×4 (16 bit) dictionary of ArUCo randomly generator markers [102], [103]. Chessboard squares were 110 mm and ArUCo markers were 80 mm on each side.

For the intrinsic and extrinsic calibration process, a minimum of 200 images with a successfully detected calibration board were captured at a resolution of 1440×1080 pixels. The desired capture volume was outlined with markers to guide calibration and ensure that the entire volume was covered. During calibration, cameras were set to capture at 10 fps to reduce the total number of images passed to the ChArUCo board detection and calibration pipeline.

For intrinsic camera calibration, the set of images contained a variety of calibration board poses that spanned a range of distances in the cameras FoV. A RANSAC approach was used to determine each camera matrix and set of distortion coefficients [91]. After a set of images were captured for each camera, intrinsic calibration was performed using ChArUCo detection to obtain points on the calibration board and OpenCV's extended library for access to the ChArUCo calibration functions. Images with poor reprojection error or too few detected ChArUCo markers were ignored during calibration. The calibration results were only accepted when a sufficiently low reprojection error was obtained from a given set of images (less than 1 pixel).

For extrinsic calibration, calibration board images were captured from a variety of distances and angles relative to the cameras. For the extrinsic calibration process to successfully determine a valid rotation and translation between the camera pairs, the calibration board must be clearly visible in all images. Ideally, a quarter of the calibration board's ChArUCo markers should be detected in an image to accurately determine calibration board points. If too few markers are detected, then the calibration board points may have a poor reprojection error. For all frames with a partially detected ChArUCo calibration board, an image point recovery algorithm was implemented to greatly increase the number of usable frames. With a set of at least 200 successfully detected calibration boards, the calculated intrinsic parameter values were used to obtain initial rotation matrix and translation vector connecting each secondary camera with the primary reference frame camera. Initial extrinsic calibration results were only accepted when a reprojection error of less than 1 pixel was achieved. The set of extrinsic parameters were further improved using bundle adjustment, a modified version of OpenPose's pipeline built using the Google Ceres library [65], [104]. The modified bundle adjustment approach utilizes some advantages of the ChArUCo calibration board to recover calibration points from images where only partial detections were obtained. Calibration was performed iteratively until extrinsic parameter pixel reprojection error was less than 0.5 pixels.

For output parameter calculations, the global coordinate system was transformed to the lab floor. An image of the calibration board on the ground was captured, and rotation and translation needed to transform the coordinate system to the board plane was calculated. This was performed various times with the board in a desired location to ensure that the new coordinate system's X-axis lined up with the virtual capture volume width and Y-axis aligned with the length.

4.6.2 Validation

Multi-camera system calibration was validated qualitatively and quantitatively. Qualitative assessment was performed by analysing images from each camera to determine the effectiveness of the distortion model and by analysing stereo-pairs of images to assess epipolar geometry characteristics. Removal of image distortions was verified by checking images for pin-cushioning or barrel distortion (Figure 4.10).

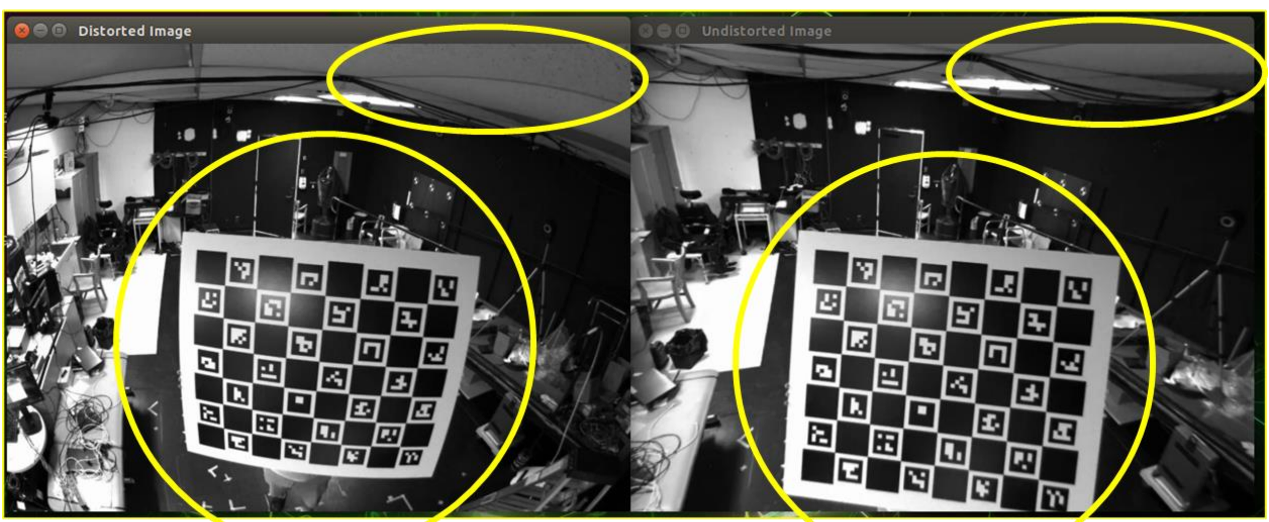


Figure 4.10 Barrel distortion removal by camera distortion model.

The above figure shows the removal of curvature caused by camera lens distortion. Features such as the square calibration board edges and ceiling tile supports become straight in the undistorted view, as opposed to the distorted view.

Images from stereo-pairs of cameras were rectified and stitched together to determine whether epipolar constraints were violated. The epipolar constraint ensures that the projection of a given point from one image must lie on the epipolar line defined by the projected point and the imaging plane epipole of the other image. Points in the left camera view were tracked along the corresponding epiline in the right camera to ensure that the same point was found. Figure 4.11 shows an example of a corner point in the left image being tracked along its epiline to the corresponding point in the right image.

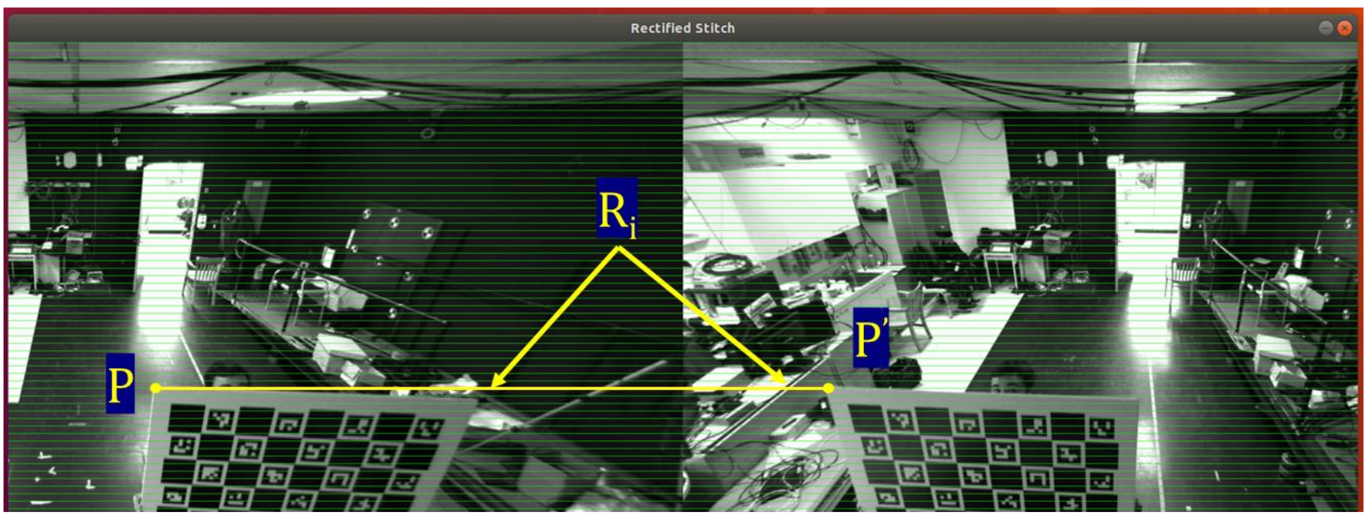


Figure 4.11 Finding a point P in the left image and point P' in the right image along the same epiline to validate the epipolar constraint.

Quantitative validation was performed using the pixel reprojection error obtained at each calibration stage. Final intrinsic parameter calibration results are documented in Table 4-5. The average extrinsic reprojection error was 0.21 pixels.

Table 4-5 Reprojection error in pixels after camera intrinsic and extrinsic parameter calibration.

Camera ID	Intrinsic Reprojection Error (pixels)
20010190	0.64
20023191	0.52
20023230	0.51
20023235	0.48

A pixel reprojection error of less than 1 pixel is generally desirable; however, this can vary depending on the image sensor resolution. The average error after intrinsic parameter calibration was less than 1 pixel for all cameras and the overall average reprojection error calculated during the extrinsic parameter bundle adjustment was 0.21 pixels. The depth accuracy of the calibrated multi-camera system was tested by measuring the capture volume length, width, and height. These values were obtained by selecting corresponding points in each camera view by placing markers on the ground or ceiling. The point correspondences were then passed to the triangulation pipeline to determine measures of the capture volume dimensions. The triangulation pipeline was written in Python 3.7 and used code from the AniposeLib GitHub repository as a structure for the triangulation procedure [12], [105]. The triangulation procedure used for the Smart Hallway applied a RANSAC approach to remove outliers in the detected points. The selected points were then triangulated using a bundle adjustment approach where the final 3D keypoint was iteratively adjusted to reduce the overall reprojection error in each camera view. After verifying the final calibration, the measured capture volume length was 5.04 ± 0.015 m, width was 2.42 ± 0.012 m, and height was 3.02 ± 0.014 m. Figure 4.12 displays the capture volume with the true dimensions and markers used to find point correspondences.

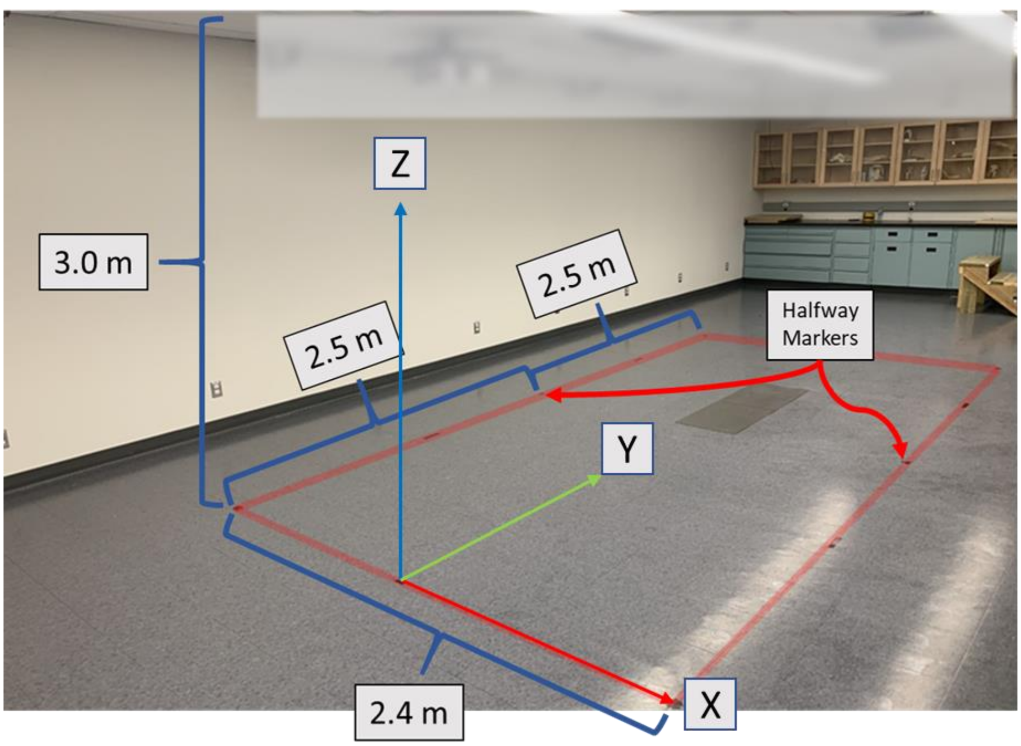


Figure 4.12 Hallway capture volume with true dimensions used to compare against dimensions obtained from triangulation. Cameras not used in the Smart Hallway are blurred.

4.7 Conclusion

From the simulations, a four-camera arc layout was the preferred choice for initial validation of the multi-camera system. The four-camera arc layout provided 11 m^2 of ground coverage and a 29 m^3 capture volume. An eight-camera perimeter layout could also be deployed to capture a longer portion of the institutional hallway with improved 3D accuracy.

Hardware selections were based on geometric and data transfer rate constraints. A list of components was detailed, and system connectivity was laid out. Based on the USB3 communication standard, 8 m cables were selected. Bandwidth calculations were completed to determine that the system data transfer capability was 380 MB per second for each camera.

A hardware synchronization cable was designed and created to ensure reliable image capture for the multi-camera system. The cable was tested in multiple scenarios and cameras remained synchronized within 5 µs when capturing at 60 fps (16667 µs/image).

A spatial synchronization method was validated for the multi-camera system using a ChArUCo calibration pattern and employing RANSAC and bundle adjustment techniques. A ChArUCo calibration approach was also selected due to its robustness against occlusion when calibrating near the edge of the camera frame or reflections due to fluorescent lighting in the hallway environment. Final reprojection error from the intrinsic and extrinsic parameters was less than 1.0 pixels. The multi-camera system calibration was tested by measuring known dimensions of the capture volume and the X-axis error was 1.7 (SD=1.2) cm, Y-axis error was 2.4 (SD=1.5) cm, and Z-axis error was 1.9 (SD=1.4) cm. This was a viable result for a clinical Smart Hallway system.

5 Validation of Stride Parameters and Limb Length Analysis

5.1 Overview

This chapter addresses Objective 3 of this research, describing Smart Hallway testing and validation in the context of an institutional hallway setting. Trials with a single participant in the capture volume were performed using a variety of walking conditions. In this chapter, the testing methodology is outlined, including Smart Hallway test setup, trials, and data processing. Results for participant body segment lengths, stride parameters, and leg angles during gait are provided and discussed. Several methods for foot event detection were implemented to determine their viability when calculating stride parameters with data obtained from the marker-less system. Stride parameters from detected foot events were compared to stride parameters obtained using ground truth foot events.

5.2 Abstract

In this chapter, the Smart Hallway was implemented, and a participant's motion was analysed after walking through the virtual hallway's capture volume. Video data were captured at 60 fps using the temporally synchronized and spatially calibrated multi-camera system, then passed to the OpenPose BODY25 pipeline on the NVIDIA Jetson AGX for 2D keypoint inference. Keypoints from each camera view were used as point correspondences and passed to the triangulation pipeline to obtain 3D joint keypoints in the lab coordinate system. A variety of walking conditions were tested, including walking with movement aids. The ground truth foot off (FO) and foot strike (FS) events from the videos were manually labelled for comparison with the foot event detection algorithm. 3D keypoints from each trial were passed to a software pipeline to calculate the body segment lengths, stride parameters, and leg angles during gait. Body segment lengths were compared with ground truth segment lengths measured directly from the participant; the measurement error was 1.56 (2.77) cm across all trials. Stride parameters calculated using the foot event detection algorithm were compared to stride parameters calculated with the ground truth foot events. Leg angles during gait were ensemble averaged and qualitatively compared with ensemble averages obtained from an able-bodied data set collected using a Vicon system [106]. Increased variance in keypoint placement accuracy was seen in endpoints of the BODY25 skeleton. Segments using endpoints had an average standard deviation of 3.66 cm compared to segments that did not

use endpoints which had an average standard deviation of 2.14 cm. Increased measurement error was also seen in keypoints identifying small joints such as the toes and ankle. Trials involving movement aids had greater standard deviation in the body segments holding the movement aid; without movement aids the average standard deviation was 1.82 cm and with movement aids the average standard deviation was 6.24 cm. Results provided insight on the overall accuracy of the Smart Hallway for analyzing human movement and validate this approach for movement analysis in future research studies.

5.3 Introduction

Human motion analysis is typically performed in a structured clinical setting where participants are instrumented with IMU sensors, or reflective markers for 3D motion tracking approaches such as Vicon or Optitrack systems [2], [21]. These tools provide accurate body kinematics results; however, they require a large amount of setup time, can cause discomfort, or can change the participants natural gait [33], [93]. Thanks to advancements in technology such as high-performance computing (HPC) units like the NVIDIA Jetson Xavier and AI models like OpenPose BODY25, a non-invasive approach to human movement analysis could be developed to capture data and provide clinicians with data for clinical decision making.

Commercially available HPC devices are capable of performing inference using state-of-the-art AI models. These HPC units also process incoming data at high rates, allowing for results in seconds or minutes depending on the data input format and AI model. A HPC unit paired with a multi-camera approach for triangulation provides a promising method for extracting accurate 3D position data [63], [69], [70].

AI for human motion analysis is an expanding field of study that has had success in motion tracking applications. Particularly, an open-source model called OpenPose has shown state-of-the-art results when performing 2D keypoint inference. OpenPose takes image or video data as input and produces a set of 2D coordinates that describe the position of a person's joint centers. OpenPose has been used to assess human motion in various settings, such as dancing or static pose-detection, to a high degree of accuracy, thus providing support for its use in a multi-camera setting to infer 3D position data [63], [65]. With accurate 3D position information, it is possible to further utilize these data for fall risk classification, fall detection, progression of neurodegenerative disorders, or rehabilitation progress assessment [16], [17], [18]. OpenPose has been used, and optimized for use with NVIDIA CUDA powered AI products, making it a desirable option for 2D keypoint inference with the Smart Hallway.

HPC units can also provide connectivity for multiple external sensors to aid in creating a Smart Hallway type system. By having a single HPC unit handle data input and data processing, the system can be deployed in a variety of situations without causing obstruction due to the small available space. HPC units can handle this overhead, while also being connected to multiple image sensors. Of particular interest is Smart Hallway system deployment in a non-invasive manner inside an institutional hallway, making use of previously unused space in a typical institutional setting by mounting the system's cameras and HPC unit to the ceiling. Institutional hallways are common areas where visitors, patients, or staff move through, making them a good candidate for non-invasive marker-less motion analysis [96].

To determine the viability of such a system and validate the accuracy of results, motion analysis testing is required. Validation included limb measurement testing to quantify 3D triangulation accuracy, stride parameter comparison between outcome measures calculated using ground truth foot events, and stride parameter outcome measures calculated using detected foot events. Of interest as well is determining hip, knee, and ankle angles during gait by comparing measured angles to typical Vicon motion lab results. Successful measurement of these parameters and implementation of the Smart Hallway system would provide a basis for future research into marker-less motion analysis tools used in intuitional hallway settings.

5.4 Methods

5.4.1 Smart Hallway Setup

The Smart Hallway was implemented based on findings from Chapter 3 and Chapter 4. A 5 m by 2.4 m area was mapped out on the laboratory floor to act as the ground plane capture area. The ceiling height was measured to be approximately 3 m. Markers were place on the floor to indicate the start and end of the capture volume, as well as 25%, 50%, and 75% points along the capture volume length. These markers were used to determine 3D reconstruction scale error. Once the capture volume was mapped out, the four-camera arc layout was setup on the ceiling based on dimensions from the Blender simulation (Figure 4.4). The camera layout and poses were measured to ensure similarity to the Blender simulation.

The hardware synchronization cable was connected to the cameras and routed along the ceiling. USB3 camera cables were connected to active extension cables and routed to the NVIDIA Jetson AGX. The cabling was routed along the ceiling so as not to obstruct the camera views or area around the capture volume. Camera power was also supplied by routing an external camera power cable along the hardware synchronization cable.

5.4.2 Calibration and Coordinate System Orientation

Calibration was performed by recording video at 10 fps and capturing a variety of calibration board poses. During calibration, the board was moved throughout the entire capture volume to cover a range of depths. This process was followed for all cameras to record four different intrinsic calibration videos, one per camera in the multi-camera array. Intrinsic calibration results were obtained using the intrinsic parameter calibration software written for this research using OpenCV's ArUCo library [107].

The primary camera that provided the hardware synchronization signal was selected as the camera for the initial global coordinate system. The secondary cameras were transformed to the primary cameras coordinate system during extrinsic calibration. Extrinsic calibration was performed by capturing three different videos, matching each secondary camera with the primary camera. In each video, the camera desired to be the primary camera and the corresponding secondary camera captured synchronized images of the calibration board. The same process for video capture used during intrinsic calibration was used for extrinsic calibration. Initial estimates of the extrinsic parameters were calculated with the extrinsic calibration videos and camera intrinsic parameters, using the extrinsic parameter calibration software written with OpenCV's ArUCo library. The results were passed to the bundle adjustment software adapted from Google Ceres and OpenPose's library [107], [108]. Bundle adjustment was performed on the extrinsic parameters, keeping the intrinsic parameters obtained during the initial calibration step as constants. The final reprojection error after extrinsic parameter bundle adjustment was 0.17 pixels for the 1440 pixel by 1080 pixel cameras. These results were stored as the final calibration parameters and only modified if a new global coordinate system was desired.

For Smart Hallway outcome measures, the coordinate system was transformed to be in line with the virtual hallway capture volume. The global coordinate system's XY plane was mapped to the laboratory floor where the X-axis was parallel to the 2.4 m width and the Y-axis was parallel with the 5 m length. The global coordinate Z-axis positive direction was mapped in the opposite direction of gravity. This transform from the primary camera global coordinate system was obtained by aligning the calibration board's local coordinate system with the desired capture volume coordinate system. Figure 5.1 shows an example of the global coordinate transformed from the primary camera's position to the laboratory ground plane.

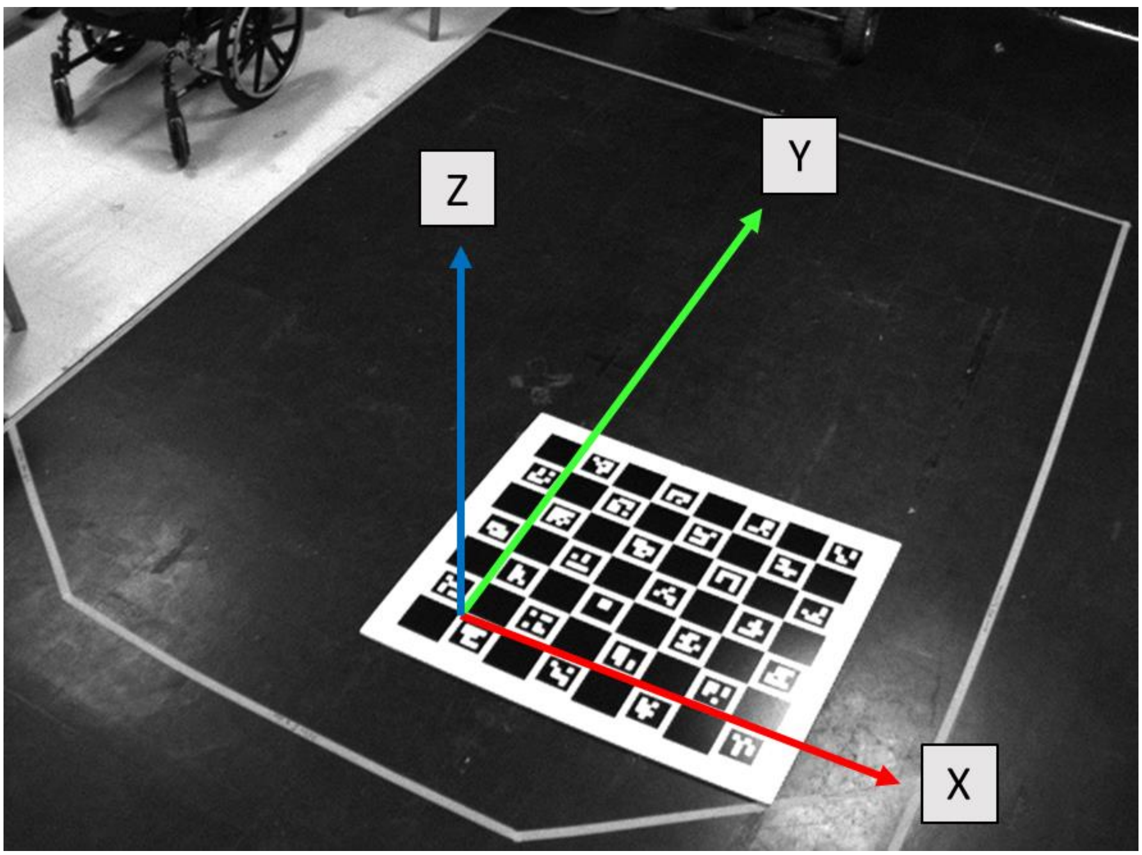


Figure 5.1 Global coordinate system transformed to the laboratory ground plane.

Software was written to determine the lab ground plane transform and allow the user to edit the XY plane position to align with markers on the ground, if needed. From Figure 5.1 the X axis is the virtual hallway width, the Y axis is the length, and the Z axis is the height.

5.4.3 Testing Protocol

The OpenPose BODY25 model was selected for inference on the videos recorded during testing. This model selection was based on findings from Chapter 3. Testing was performed with two male graduate students working on the Smart Hallway project (age: 28, height: 180 cm; weight: 64 kg and age: 25, height: 178 cm; weight: 90 kg). Recruitment of a larger test sample was not possible due to gathering limitations caused by COVID-19 restrictions.

Outcome measures for the Smart Hallway included body segment lengths, stride parameters, and leg angles during gait. Prior to testing, each participant's body segment lengths were measured using an anthropometric tape. The segment matched OpenPose's BODY25 model (Figure 3.1) and were measured by palpating each participant and finding the joint centers of interest for each measurement. Only measuring the participants body segment lengths allowed for reduced close-contact to abide by COVID-19 restrictions. Additionally, this approach did not require affixing reflective markers to the participant, further reducing close-contact time.". Each participant completed five separate trials of the five walking conditions detailed in Table 5-1. The participants were recorded with the four-camera array at 60 fps. This protocol was approved by the Research Ethics Board at the University of Ottawa [109].

Table 5-1 Test conditions, protocol, and notes.

Condition	Protocol	Notes
Walking straight	Start one meter outside the capture volume and walks straight. Once through the capture volume, turn around and walk back to the initial position.	The turn occurs outside of the camera field of view.
Walking and turning	Start at the edge of the capture volume and walks towards a marker positioned 50 cm from the end of the capture volume. Turn around the marker and walks back to the initial position.	The turn occurs within the camera field of view.
Walking in a curved path	Starts at the edge of the capture volume and walks in a curved path around the capture volume. The test ends once the participant reaches their initial position.	The participant performs each test in the same direction.
Walking with a cane	Follow the "Walking straight" protocol while using a cane as a walking aid.	Cane held in the same hand for all trials. Participants were instructed on how to properly use a cane.
Walking with a walker	Follow the "Walking straight" protocol while using a wheeled walker as a walking aid.	Participants were instructed on how to properly use a walker.

5.4.4 2D Keypoint Processing

Videos of each participant were recorded and stored on the NVIDIA Jetson AGX's solid state drive. The videos were passed to the OpenPose BODY25 model to perform inference and create a set of 2D keypoints, locating participant joint centers for every video frame. For every video, the 2D keypoint data contained confidence scores that described the likelihood of correct marker location. Data from each video were preprocessed by thresholding points below 5% confidence and using a cubic spline to interpolate gaps in the data set that were five frames (0.083 s) or shorter. The dataset was then filtered using a zero-phase low-pass 12 Hz Butterworth filter. Figure 5.2 shows an example of the 2D keypoint data after preprocessing.

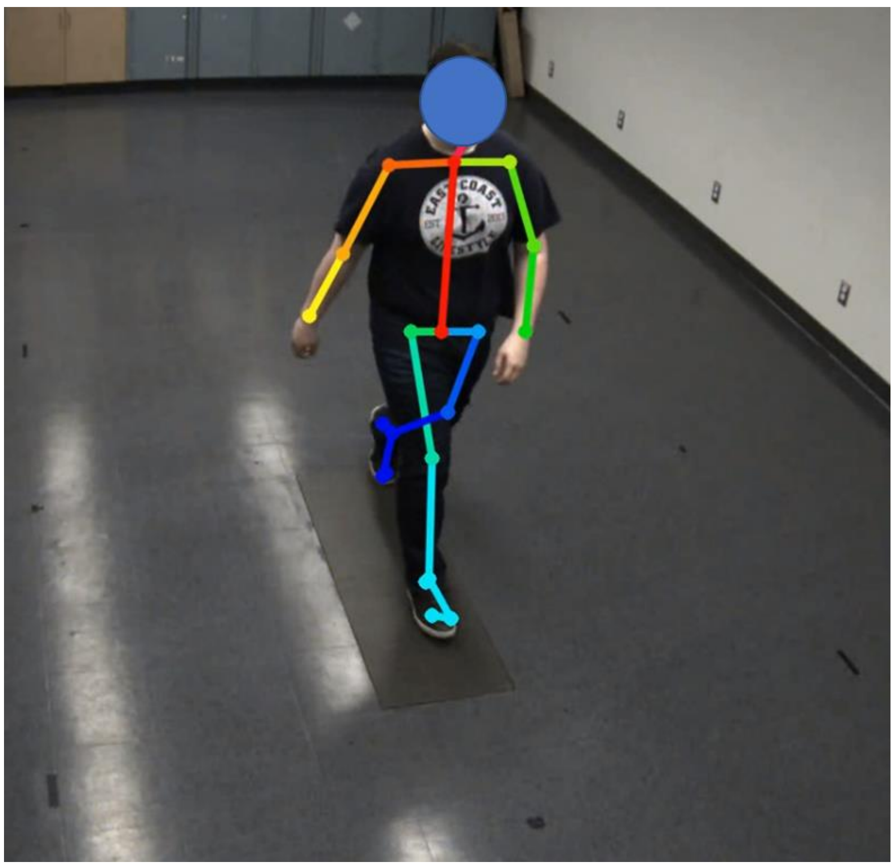


Figure 5.2 Example output from the OpenPose BODY25 model after data processing.

For every trial, the 2D keypoint data from each camera were passed to the triangulation pipeline along with the intrinsic and extrinsic parameters. The 2D keypoints were used to find an optimal set of 3D points that located the participant's joint centers in space throughout the video.

5.4.5 3D Keypoint Processing and Outcome Measures

Software was written in Python 3.7 to calculate the body segment lengths, stride parameters, and hip, knee, and ankle angles during gait. For each trial's input 3D data, the regions where 3D keypoint data reprojection error exceeded two standard deviations from the mean were dropped to reduce the effects of outliers. The post processing pipeline filtered the input 3D data using a zero-phase low-pass 5 Hz Butterworth filter, based on findings from other research involving analyses of OpenPose keypoint inferences and marker-based approaches [8], [110]. Figure 5.3 shows the filtered X, Y, and Z axis coordinates of the OpenPose BODY25 model at each timestep.

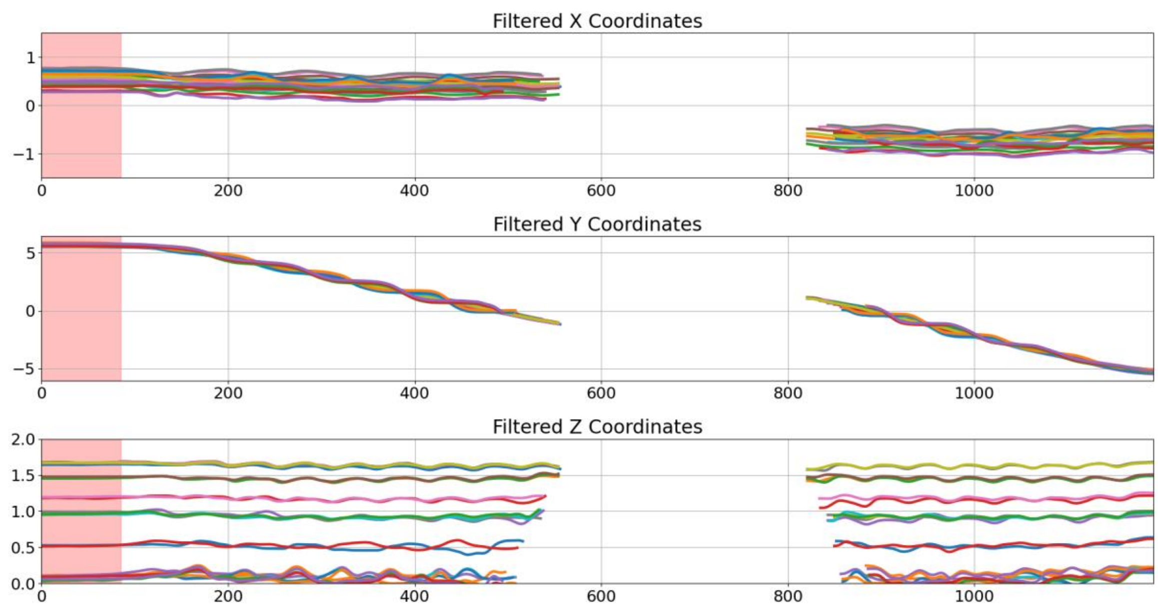


Figure 5.3 Filtered 3D coordinates from a single trial. The gaps in the data are regions where the participant is out of frame. The horizontal axis is the frame number, and the vertical axis is distance measured in meters.

Regions where no person is detected in the frame were ignored during data processing, as shown in Figure 5.3. These points show up as NaN (not-a-number) in the Python data structures. Regions in the data where stops or turns occurred were ignored for stride parameter and leg angle calculations due to the variability in turning compared to a person's straight walking gait. Turns were detected by finding an optimal crossing point between the shoulder and hip keypoints. The optimal turning point was determined by tracking the participant's shoulders and hips positions in the X-axis and selecting a turning point that provided the minimum intersection error across all the keypoints of interest. Figure 5.4 shows an example of how the turning point was detected based on the curve crossing points.

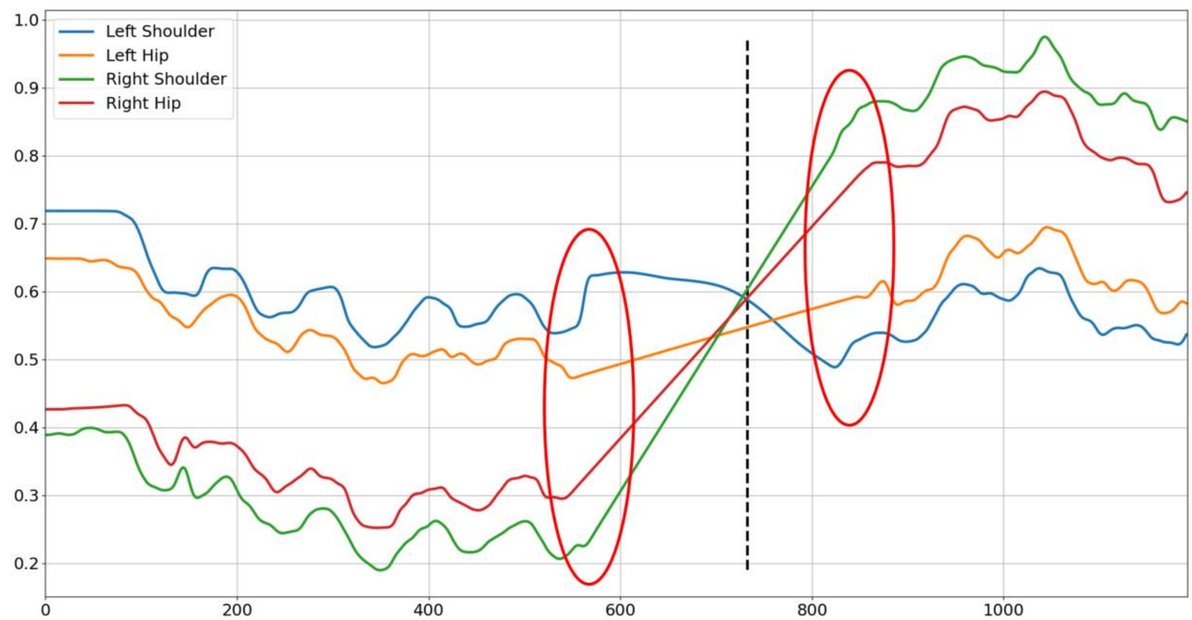


Figure 5.4 Selection of turning points based on the hip and shoulder keypoints. The red ellipses identify the beginning and end of the turn. The horizontal axis is the frame number, and the vertical axis is distance measured in meters.

The point detected as the turn was the turn midpoint. Regions before and after the turn midpoint were searched to determine the turn start and end. This search is performed based on the X-axis change over time, where the beginning of the turn is the earliest point where the slope begins to change, and the end of the turn is the latest point with a slope change.

Stops were detected based on a torso velocity threshold. Regions at the start or end of videos where the participant was not moving were removed for analyses. Together, the removed NaN regions, turns, and stopped regions resulted in dataset windows for outcome measure calculations. Figure 5.5 shows the torso position over time and how the stopped region and NaN region were identified.

From Figure 5.5, stride parameters would be calculated in the white regions and data inside the red or blue regions were ignored. The participants body segment lengths were calculated when a person was in frame.

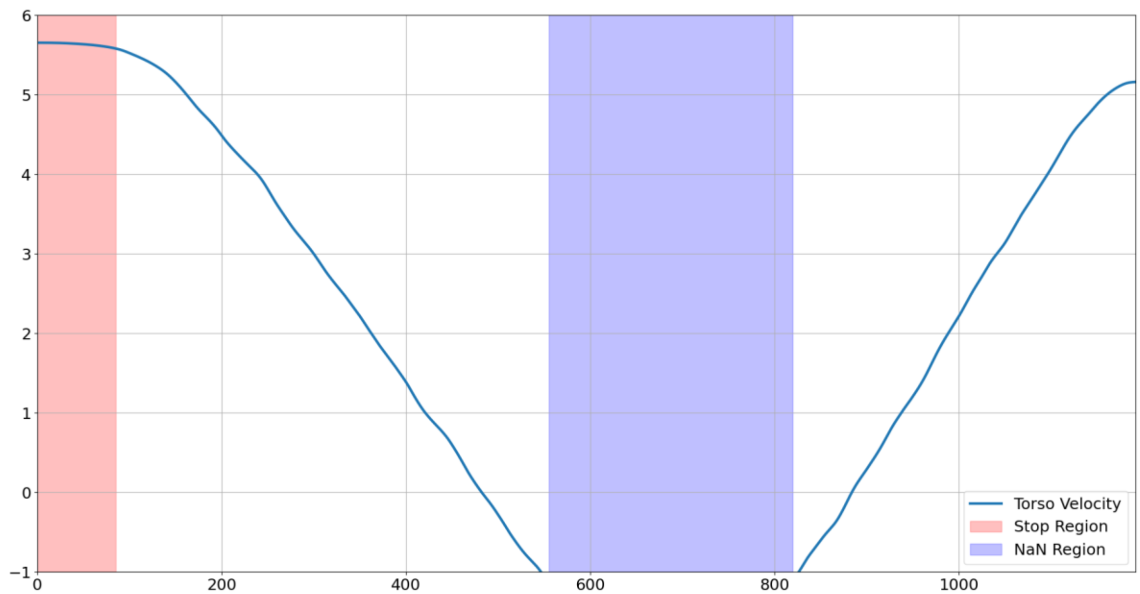


Figure 5.5 Example of the torso position over time. The red shaded region indicates that the person is not moving. The blue shade region indicates that no one is detected in frame. The horizontal axis is the frame number, and the vertical axis is distance measured in meters.

A. Body Segment Lengths

Body segments lengths were calculated using the Euclidean distance between the keypoints of interest. For example, forearm length was the Euclidean distance between the elbow and wrist keypoints. Outliers were removed from the list of body segment lengths by ignoring values outside two standard deviations from the mean. Body segment length deltas were calculated by subtracting the calculated length from the measured ground truth value obtained through direct measurement prior to testing.

B. Stride Parameters

Stride parameters calculations were performed with data between stopped regions and NaN regions. Ground truth stride parameters were calculated from ground truth foot events obtained through direct observation by manually labelling foot offs and foot strikes in each video. For each participant, an Excel file of ground truth foot events was created to compare with the detected foot events. Detected foot events were obtained by using the Zeni, et al. algorithm [13]. The set of detected foot events was improved by implementing an algorithm to recover gait initiation and gait termination (Initiation Termination Recovery, IT Recovery). Figure 5.6 shows an example of the detected foot events prior to the initiation and termination recovery algorithm.

Figure 5.6 Ground truth foot events (green) and detected foot events (red). The red ellipse shows a gait initiation missed by the Zeni foot event detection algorithm. The horizontal axis is the frame number, and the vertical axis is distance measured in meters.

Regions like the one highlighted by the red ellipse in Figure 5.6 were recovered by searching the window between the stop and the next detected foot event. The algorithm determined if an initiation occurred by analyzing the linear fit of the curve in a calculated search region. The search region was assessed by detecting a potential foot event, using SciPy's signal.find_peaks function, and fitting a line between the potential foot event and the next detected foot event [111]. Linear fits above an R-squared value of 0.85 were selected as gait initiations and added to the list of detected foot events. Foot events that were not recovered, or missed by Zeni's algorithm, were backfilled using the algorithm proposed by Capela, Lemaire, and Baddour [112], [113].

Stride parameters were calculated for ground truth foot events and detected foot events (Table 5-2). Results for comparison included mean (μ) and standard deviation (σ) across all trials of the same walking condition using equations (5.1) and (5.2), respectively. The mean delta (μ_Δ), and standard deviation of the delta (σ_Δ), between the two sets were also determined using equation (5.3) and (5.4), respectively. The stride parameter deltas were calculated by temporally synchronizing the detected and ground truth stride parameters and subtracting the detected stride parameter value from the ground truth value.

$$\mu = \frac{\sum_{i=1}^{N} x_i}{N} \tag{5.1}$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \mu)^2}{N}} \tag{5.2}$$

$$\mu_\Delta = \frac{\sum_{i=1}^{N} x_i^g - x_i^d}{N} \tag{5.3}$$

$$\sigma_\Delta = \sqrt{\frac{\sum_{i=1}^{N}\left(x_i^g - x_i^d - u_\Delta\right)^2}{N}} \tag{5.4}$$

N is the number of specific calculated stride parameters and x_i is the value of a calculated stride parameter. For delta calculations, the superscript d indicates detected and g indicates ground truth. Table 5-2 describes the stride parameters obtained from the foot events and 3D data.

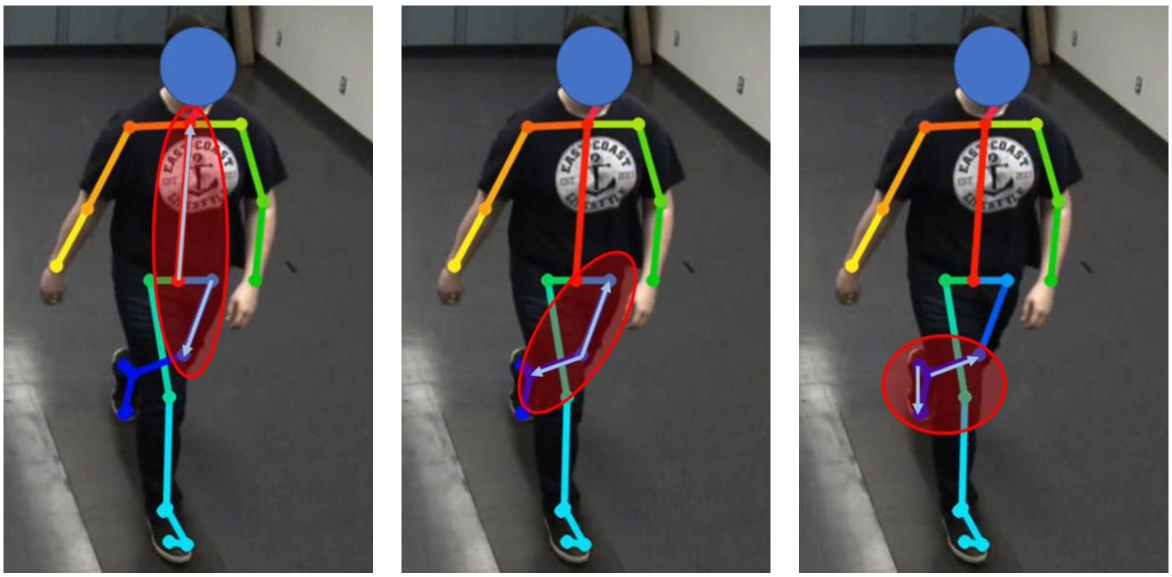
Table 5-2 Stride parameters definitions. FS=foot strike, FO=foot off.

Parameter	Definition
Stride Length	Distance covered by the heel between two consecutive FS along the sagittal plane (line of progression)
Stride Time	Number of frames between two consecutive FS divided by the fps
Stride Speed	Stride Length divided by Stride Time
Step Length	Distance between the heel of interest and the contralateral heel along the sagittal plane (line of progression) at the heel of interest's FS
Step Width	Distance between heels along the frontal plane (perpendicular to the line of progression) at FS
Step Time	Number of frames between two consecutive FS divided by the fps
Cadence	Steps per minute
Stance Time	Frames between FS and FO divided by the fps
Swing Time	Frames between FO and FS divided by the fps
Stance Swing Ratio	Stance Time divided by Swing Time
Double Support Time	Frames between FS and the contralateral FO divided by fps
Foot Angle	Angle of the foot midline relative to the sagittal plane (line of progression) when the foot is in contact with the ground plane

Events unintentionally calculated due to the ambiguity between initiations and termination during turns versus true gait initiation or termination were removed from the stride parameter dataset.

C. Gait Leg Angles

Hip, knee, and ankle angles for the left and right leg during gait were calculated based on the consecutive FS regions defined by the ground truth and detected foot events described in section 5.4.5B. Figure 5.7 shows how the vectors used in the angle calculations were defined.



(a) **(b)** **(c)**

Figure 5.7 Leg angle vector definitions: a) Torso vector and the thigh vector, b) Thigh vector and the shank vector, c) Shank vector and the foot vector.

Hip angle was the angle between the torso vector and thigh vector, knee angle was the inner angle between the thigh and shank vector, and ankle angle was the inner angle between the shank and foot vector.

5.5 Validation

From Table 5-3 and Table 5-4, mean differences between calculated and ground truth values were small for the majority of body segment lengths. In general, the calculated body segment lengths were less than the ground truth values. The average difference across all test conditions was 1.56 (SD=2.77) cm.

Body segment length measurements using foot (ankle, heel, toe) or wrist keypoints had a greater standard deviation than other measurements. Foot keypoints had greater variation in placement due to being a part of smaller joints [65], being the furthest keypoints from the cameras, and having more occlusion during gait. Combined with these reasons, the toe, heel, and wrist keypoints are end points of the OpenPose BODY25 skeleton model and are not as tightly constrained compared to other keypoints [65].

From analysing results across test conditions, there was negligible difference between body segment lengths measured during walking straight, walking turn, and walking curve tests. Of note, walking aids (cane, walker) increased variability in some body segment length measurements. During the cane tests, the hand that was holding the cane had reduced keypoint location accuracy and increased forearm length standard deviation compared to the contralateral forearm. During walker tests, both forearm measurements had increased standard deviations, and increased standard deviation was seen in the leg measurements due to occlusion caused by the walker.

Table 5-3 and Table 5-4 showed that segments with a large range of motion during typical gait had higher standard deviations compared to body segments such as shoulder width and upper arms.

Table 5-3 Body segment lengths for the walking straight, walking turn, and walking curve test conditions (mean and standard deviation in brackets). Smart Hallway (SH) are calculated using the 3D reconstructed data and Delta is the difference between the Smart Hallway and ground truth segment lengths.

Walking Condition	**Walking Straight**		**Walking Turn**		**Walking Curve**	
Limb Segment (cm)	**SH**	**Delta**	**SH**	**Delta**	**SH**	**Delta**
Left Arm	29.60 (1.92)	0.10 (1.92)	28.49 (1.63)	-1.01 (1.63)	28.75 (1.33)	-0.75 (1.33)
Left Forearm	25.37 (2.41)	-1.13 (2.41)	25.75 (1.53)	-0.75 (1.53)	25.68 (1.15)	-0.82 (1.15)
Right Arm	29.07 (1.93)	0.07 (1.93)	28.74 (1.65)	-0.26 (1.65)	28.88 (1.24)	-0.12 (1.24)
Right Forearm	24.91 (2.76)	-2.09 (2.76)	25.24 (2.12)	-1.76 (2.12)	25.18 (1.51)	-1.82 (1.51)
Left Thigh	38.29 (2.51)	-3.21 (2.51)	38.45 (2.59)	-3.05 (2.59)	38.05 (2.26)	-3.45 (2.26)
Left Shank	39.01 (3.94)	-0.99 (3.94)	39.62 (4.11)	-0.38 (4.11)	39.38 (3.17)	-0.62 (3.17)
Right Thigh	37.88 (2.29)	-3.12 (2.29)	37.93 (1.93)	-3.07 (1.93)	37.30 (2.00)	-3.70 (2.00)
Right Shank	38.51 (3.15)	-1.49 (3.15)	39.17 (3.84)	-0.83 (3.84)	38.78 (3.13)	-1.22 (3.13)
Left Ankle to Heel	6.91 (2.51)	-0.09 (2.51)	7.50 (2.81)	0.50 (2.81)	6.76 (1.73)	-0.24 (1.73)
Left Ankle to Big Toe	17.16 (5.46)	-0.84 (5.46)	18.12 (4.40)	0.12 (4.40)	16.75 (2.88)	-1.25 (2.88)
Left Ankle to Small Toe	14.37 (4.49)	-2.63 (4.49)	15.29 (4.30)	-1.71 (4.30)	13.90 (2.60)	-3.10 (2.60)
Left Toe Width	7.40 (2.37)	-0.10 (2.37)	7.72 (2.96)	0.22 (2.96)	6.76 (1.86)	-0.74 (1.86)
Right Ankle to Heel	7.76 (2.90)	0.26 (2.90)	7.09 (2.68)	-0.41 (2.68)	6.87 (1.90)	-0.63 (1.90)
Right Ankle to Big Toe	15.63 (4.65)	-2.87 (4.65)	16.91 (5.09)	-1.59 (5.09)	16.21 (3.08)	-2.29 (3.08)
Right Ankle to Small Toe	12.87 (4.01)	-4.63 (4.01)	14.39 (3.22)	-3.11 (3.22)	13.96 (2.46)	-3.54 (2.46)
Right Toe Width	8.85 (3.51)	1.35 (3.51)	7.73 (2.40)	0.23 (2.40)	7.08 (2.62)	-0.42 (2.62)
Shoulder Width	35.33 (1.77)	-0.67 (1.77)	35.24 (1.35)	-0.76 (1.35)	34.62 (2.09)	-1.38 (2.09)
Hip Width	22.45 (1.33)	-1.05 (1.33)	22.25 (1.33)	-1.25 (1.33)	22.15 (1.70)	-1.35 (1.70)
Chest Height	53.91 (1.93)	-1.09 (1.93)	53.57 (1.45)	-1.43 (1.45)	54.27 (1.24)	-0.73 (1.24)

Table 5-4 Body segment lengths for the walking straight, cane, and walker test conditions (mean and standard deviation in brackets). Smart Hallway (SH) are calculated using the 3D reconstructed data and Delta is the difference between the Smart Hallway and ground truth segment length.

Walking Condition	Walking Straight		Cane		Walker	
Limb Segment (cm)	SH	Delta	SH	Delta	SH	Delta
Left Arm	29.60 (1.92)	0.10 (1.92)	29.36 (1.18)	-0.14 (1.18)	28.82 (1.64)	-0.68 (1.64)
Left Forearm	25.37 (2.41)	-1.13 (2.41)	25.62 (1.27)	-0.88 (1.27)	30.79 (7.50)	4.29 (7.50)
Right Arm	29.07 (1.93)	0.07 (1.93)	28.59 (1.66)	-0.41 (1.66)	28.61 (1.86)	-0.39 (1.86)
Right Forearm	24.91 (2.76)	-2.09 (2.76)	26.20 (3.92)	-0.80 (3.92)	30.57 (7.29)	3.57 (7.29)
Left Thigh	38.29 (2.51)	-3.21 (2.51)	38.19 (2.33)	-3.31 (2.33)	39.32 (2.90)	-2.18 (2.90)
Left Shank	39.01 (3.94)	-0.99 (3.94)	39.02 (2.75)	-0.98 (2.75)	40.29 (3.65)	0.29 (3.65)
Right Thigh	37.88 (2.29)	-3.12 (2.29)	38.20 (2.45)	-2.80 (2.45)	38.90 (3.19)	-2.10 (3.19)
Right Shank	38.51 (3.15)	-1.49 (3.15)	38.85 (3.04)	-1.15 (3.04)	40.47 (4.07)	0.47 (4.07)
Left Ankle to Heel	6.91 (2.51)	-0.09 (2.51)	7.17 (2.66)	0.17 (2.66)	6.87 (2.39)	-0.13 (2.39)
Left Ankle to Big Toe	17.16 (5.46)	-0.84 (5.46)	16.65 (5.02)	-1.35 (5.02)	14.89 (4.47)	-3.11 (4.47)
Left Ankle to Small Toe	14.37 (4.49)	-2.63 (4.49)	14.27 (3.82)	-2.73 (3.82)	12.34 (3.90)	-4.66 (3.90)
Left Toe Width	7.40 (2.37)	-0.10 (2.37)	7.29 (2.36)	-0.21 (2.36)	7.52 (2.41)	0.02 (2.41)
Right Ankle to Heel	7.76 (2.90)	0.26 (2.90)	6.51 (2.56)	-0.99 (2.56)	5.94 (1.96)	-1.56 (1.96)
Right Ankle to Big Toe	15.63 (4.65)	-2.87 (4.65)	15.10 (5.25)	-3.40 (5.25)	15.36 (4.07)	-3.14 (4.07)
Right Ankle to Small Toe	12.87 (4.01)	-4.63 (4.01)	12.39 (3.96)	-5.11 (3.96)	13.34 (3.75)	-4.16 (3.75)
Right Toe Width	8.85 (3.51)	1.35 (3.51)	7.05 (2.67)	-0.45 (2.67)	7.19 (2.02)	-0.31 (2.02)
Shoulder Width	35.33 (1.77)	-0.67 (1.77)	34.99 (1.26)	-1.01 (1.26)	35.33 (1.22)	-0.67 (1.22)
Hip Width	22.45 (1.33)	-1.05 (1.33)	22.21 (0.93)	-1.29 (0.93)	22.05 (0.96)	-1.45 (0.96)
Chest Height	53.91 (1.93)	-1.09 (1.93)	54.34 (1.26)	-0.66 (1.26)	54.45 (1.47)	-0.55 (1.47)

From Table 5-5 through Table 5-10, calculated stride parameters were comparable to ground truth results. Table 5-5 shows the mean absolute error of the foot event detection algorithms compared to ground truth values obtained from manual labelling. The error in ground truth labelling was three frames and the average error in detected events across all trials was four frames. Prior to implementing the Capela algorithm for foot event backing filling, 86.1% of foot events were properly detected. The combined approach including all algorithms was 98.1% accurate.

Stride parameters that were only reliant on a single type of foot event (e.g., only left foot strikes) were highly accurate. Increased error and standard deviation in stride parameter measurements was seen in stride parameters that relied on multiple types of foot events, including, step length, step width, and stance swing ratio. This is likely due to compounding error effects caused by combining either contralateral foot events, foot strikes and foot offs, or foot events and 3D data.

Table 5-5 Detected foot events frame offset from ground truth values (mean and standard deviation in brackets). The percentage of events detected using Zeni [13] and IT Recovery algorithms in combination is shown alongside the percentage of events detected using all the proposed foot event detection methods.

Condition	Offset (Frames, (μ (σ)))	Zeni and IT Recovery (%)	Zeni, IT Recovery, and Capela (%)
Walking Straight	4.38 (2.72)	89.2	98.2
Walking Turn	4.13 (2.60)	83.0	98.6
Walking Curve	3.99 (3.42)	84.3	97.4
Cane	3.01 (2.52)	85.5	97.7
Walker	5.25 (3.57)	88.9	98.7

Table 5-6 Stride parameters for the Walking Straight test (mean and standard deviation in brackets).

	Left			Right		
Parameters (units)	**SH**	**Ground Truth**	**Delta**	**SH**	**Ground Truth**	**Delta**
Stride length (m)	1.60 (0.26)	1.62 (0.26)	0.03 (0.07)	1.63 (0.21)	1.63 (0.19)	-0.01 (0.08)
Stride time (s)	1.18 (0.05)	1.19 (0.05)	0.00 (0.05)	1.19 (0.06)	1.19 (0.07)	0.00 (0.04)
Stride speed (m/s)	1.35 (0.22)	1.36 (0.21)	0.01 (0.03)	1.38 (0.16)	1.37 (0.14)	0.00 (0.06)
Step length (m)	0.50 (0.19)	0.38 (0.16)	-0.13 (0.06)	0.44 (0.19)	0.36 (0.16)	-0.08 (0.06)
Step width (m)	0.09 (0.03)	0.11 (0.02)	0.02 (0.01)	0.12 (0.04)	0.13 (0.03)	0.01 (0.02)
Step time (s)	0.61 (0.10)	0.60 (0.08)	-0.01 (0.06)	0.59 (0.06)	0.59 (0.08)	0.00 (0.05)
Cadence (steps/min)	98.13 (4.54)	99.73 (3.29)	1.60 (4.87)	102.41 (6.30)	101.45 (8.68)	-0.95 (4.60)
Stance time (s)	0.73 (0.04)	0.78 (0.04)	0.05 (0.05)	0.72 (0.06)	0.77 (0.05)	0.05 (0.07)
Swing time (s)	0.46 (0.10)	0.40 (0.04)	-0.06 (0.10)	0.50 (0.11)	0.43 (0.03)	-0.08 (0.10)
Stance swing ratio (NA)	1.61 (0.35)	1.94 (0.26)	0.33 (0.36)	1.44 (0.22)	1.82 (0.20)	0.38 (0.34)
Double support time (s)	0.13 (0.10)	0.18 (0.03)	0.05 (0.11)	0.13 (0.03)	0.18 (0.03)	0.05 (0.05)
Foot angle (°)	12.82 (4.86)	12.94 (4.70)	0.12 (1.04)	14.24 (6.14)	14.57 (7.30)	0.34 (2.33)

Table 5-7 Stride parameters for the Walking Turn test (mean and standard deviation in brackets).

	Left			Right		
Parameters (units)	**SH**	**Ground Truth**	**Delta**	**SH**	**Ground Truth**	**Delta**
Stride length (m)	1.32 (0.36)	1.33 (0.39)	0.01 (0.13)	1.41 (0.35)	1.40 (0.37)	-0.01 (0.14)
Stride time (s)	1.18 (0.09)	1.19 (0.06)	0.00 (0.08)	1.24 (0.10)	1.21 (0.08)	-0.03 (0.09)
Stride speed (m/s)	1.12 (0.30)	1.11 (0.32)	-0.01 (0.10)	1.15 (0.35)	1.17 (0.35)	0.02 (0.06)
Step length (m)	0.41 (0.18)	0.33 (0.15)	-0.08 (0.07)	0.40 (0.17)	0.32 (0.14)	-0.08 (0.07)
Step width (m)	0.12 (0.05)	0.13 (0.04)	0.01 (0.02)	0.12 (0.05)	0.12 (0.04)	0.01 (0.01)
Step time (s)	0.61 (0.13)	0.59 (0.12)	-0.01 (0.08)	0.68 (0.25)	0.65 (0.22)	-0.03 (0.08)
Cadence (steps/min)	98.24 (8.45)	102.36(12.35)	4.13 (7.25)	90.53 (10.52)	92.61 (10.40)	2.07 (2.60)
Stance time (s)	0.74 (0.09)	0.78 (0.06)	0.04 (0.07)	0.74 (0.10)	0.78 (0.08)	0.04 (0.07)
Swing time (s)	0.48 (0.07)	0.43 (0.05)	-0.06 (0.08)	0.50 (0.08)	0.43 (0.06)	-0.08 (0.09)
Stance swing ratio	1.57 (0.34)	1.81 (0.29)	0.25 (0.40)	1.47 (0.22)	1.79 (0.22)	0.32 (0.30)
Double support time (s)	0.13 (0.07)	0.17 (0.04)	0.04 (0.06)	0.13 (0.06)	0.17 (0.04)	0.04 (0.05)
Foot angle (°)	13.06 (4.92)	13.41 (4.93)	0.35 (0.91)	13.81 (6.71)	14.54 (7.37)	0.72 (1.52)

Table 5-8 Stride parameters for the Walking Curve test (mean and standard deviation in brackets).

	Left			Right		
Parameters (units)	**SH**	**Ground Truth**	**Delta**	**SH**	**Ground Truth**	**Delta**
Stride length (m)	1.29 (0.33)	1.31 (0.24)	0.02 (0.16)	1.37 (0.27)	1.39 (0.25)	0.01 (0.12)
Stride time (s)	1.21 (0.13)	1.23 (0.05)	0.02 (0.10)	1.25 (0.09)	1.25 (0.06)	0.00 (0.06)
Stride speed (m/s)	1.05 (0.23)	1.06 (0.18)	0.01 (0.08)	1.11 (0.13)	1.12 (0.13)	0.00 (0.03)
Step length (m)	0.38 (0.18)	0.30 (0.13)	-0.08 (0.08)	0.41 (0.15)	0.32 (0.13)	-0.08 (0.06)
Step width (m)	0.29 (0.16)	0.24 (0.17)	-0.04 (0.09)	0.30 (0.18)	0.27 (0.15)	-0.04 (0.04)
Step time (s)	0.61 (0.16)	0.63 (0.10)	0.02 (0.10)	0.65 (0.12)	0.64 (0.12)	-0.01 (0.05)
Cadence (steps/min)	98.18 (7.68)	96.63 (6.01)	-1.55 (5.31)	93.06 (8.22)	93.80 (7.32)	0.75 (4.52)
Stance time (s)	0.76 (0.11)	0.83 (0.07)	0.07 (0.08)	0.78 (0.10)	0.84 (0.04)	0.06 (0.07)
Swing time (s)	0.45 (0.08)	0.40 (0.04)	-0.06 (0.07)	0.46 (0.07)	0.41 (0.04)	-0.05 (0.10)
Stance swing ratio	1.70 (0.29)	2.08 (0.20)	0.38 (0.32)	1.70 (0.28)	2.04 (0.24)	0.34 (0.27)
Double support time (s)	0.17 (0.07)	0.21 (0.03)	0.04 (0.08)	0.18 (0.06)	0.22 (0.03)	0.05 (0.06)
Foot angle (°)	32.09 (7.04)	32.59 (6.77)	0.50 (3.81)	36.83(11.21)	37.93 (11.49)	1.10 (2.41)

Table 5-9 Stride parameters for the Cane test (mean and standard deviation in brackets).

	Left			Right		
Parameters (units)	**SH**	**Ground Truth**	**Delta**	**SH**	**Ground Truth**	**Delta**
Stride length (m)	1.34 (0.18)	1.32 (0.16)	-0.01 (0.08)	1.34 (0.20)	1.34 (0.21)	0.00 (0.05)
Stride time (s)	1.90 (0.16)	1.88 (0.11)	-0.02 (0.10)	1.85 (0.13)	1.85 (0.13)	0.00 (0.06)
Stride speed (m/s)	0.71 (0.10)	0.71 (0.09)	0.00 (0.02)	0.73 (0.07)	0.73 (0.07)	0.00 (0.02)
Step length (m)	0.42 (0.15)	0.37 (0.14)	-0.05 (0.03)	0.43 (0.14)	0.41 (0.14)	-0.02 (0.03)
Step width (m)	0.13 (0.03)	0.14 (0.03)	0.01 (0.01)	0.14 (0.03)	0.14 (0.03)	0.00 (0.01)
Step time (s)	0.95 (0.18)	0.90 (0.19)	-0.05 (0.06)	0.96 (0.17)	1.00 (0.16)	0.04 (0.07)
Cadence (steps/min)	63.72 (6.00)	67.25 (7.59)	3.53 (2.38)	62.58 (4.62)	60.15 (3.65)	-2.43 (2.13)
Stance time (s)	1.21 (0.13)	1.31 (0.10)	0.09 (0.08)	1.15 (0.10)	1.18 (0.10)	0.03 (0.06)
Swing time (s)	0.66 (0.10)	0.57 (0.06)	-0.10 (0.08)	0.71 (0.08)	0.67 (0.09)	-0.04 (0.07)
Stance swing ratio	1.85 (0.37)	2.28 (0.34)	0.43 (0.26)	1.62 (0.22)	1.75 (0.31)	0.14 (0.24)
Double support time (s)	0.28 (0.10)	0.35 (0.06)	0.07 (0.07)	0.22 (0.06)	0.27 (0.07)	0.05 (0.06)
Foot angle (°)	11.86 (3.51)	12.11 (3.48)	0.25 (0.79)	12.44 (5.74)	12.55 (5.69)	0.12 (0.65)

Table 5-10 Stride parameters for the Walker test (mean and standard deviation in brackets).

	Left			**Right**		
Parameters (units)	**SH**	**Ground Truth**	**Delta**	**SH**	**Ground Truth**	**Delta**
Stride length (m)	1.07 (0.15)	1.07 (0.14)	0.00 (0.07)	1.07 (0.16)	1.07 (0.15)	0.00 (0.07)
Stride time (s)	1.79 (0.15)	1.78 (0.15)	-0.01 (0.08)	1.77 (0.19)	1.77 (0.17)	-0.01 (0.11)
Stride speed (m/s)	0.60 (0.09)	0.60 (0.09)	0.00 (0.02)	0.61 (0.09)	0.61 (0.09)	0.00 (0.02)
Step length (m)	0.34 (0.14)	0.29 (0.12)	-0.05 (0.04)	0.33 (0.12)	0.29 (0.11)	-0.04 (0.04)
Step width (m)	0.09 (0.03)	0.11 (0.03)	0.01 (0.01)	0.12 (0.02)	0.13 (0.02)	0.01 (0.01)
Step time (s)	0.89 (0.12)	0.86 (0.13)	-0.03 (0.09)	0.93 (0.24)	0.94 (0.24)	0.01 (0.09)
Cadence (steps/min)	67.62 (5.24)	69.90 (5.72)	2.28 (2.53)	64.03 (7.74)	63.51 (8.44)	-0.52 (2.25)
Stance time (s)	1.14 (0.12)	1.29 (0.12)	0.14 (0.08)	1.15 (0.11)	1.22 (0.12)	0.08 (0.10)
Swing time (s)	0.64 (0.08)	0.50 (0.06)	-0.14 (0.07)	0.63 (0.13)	0.50 (0.09)	-0.12 (0.09)
Stance swing ratio	1.80 (0.25)	2.57 (0.33)	0.77 (0.37)	1.81 (0.20)	2.43 (0.28)	0.62 (0.35)
Double support time (s)	0.30 (0.20)	0.40 (0.18)	0.10 (0.08)	0.26 (0.07)	0.37 (0.06)	0.11 (0.08)
Foot angle (°)	12.89 (4.24)	13.10 (4.28)	0.21 (0.67)	13.05 (6.34)	13.38 (6.30)	0.33 (1.68)

Figure 5.8, Figure 5.9, and Figure 5.10 show the ensemble averaged leg angles during gait, from FS to FO. These results were compared to able-bodied ensemble average leg angles obtained from a Vicon marker system during a different study. Curves obtained from the keypoint data had greater standard deviations, yet similar overall shape and range of motion relative to the comparator Vicon results. Increased standard deviations occurred where occlusions occurred or large variance in the Y depth coordinate between the keypoints of interest. Greater error in the lab Y-axis is expected since this axis is related to scene depth and is sensitive to the accuracy of the triangulation method. These occlusions and points of greater variance generally occurred at foot strike and foot off, where the leg is either at the maximum distance in front of the body or at the maximum distance behind the body, causing a greater variation in the Y-axis.

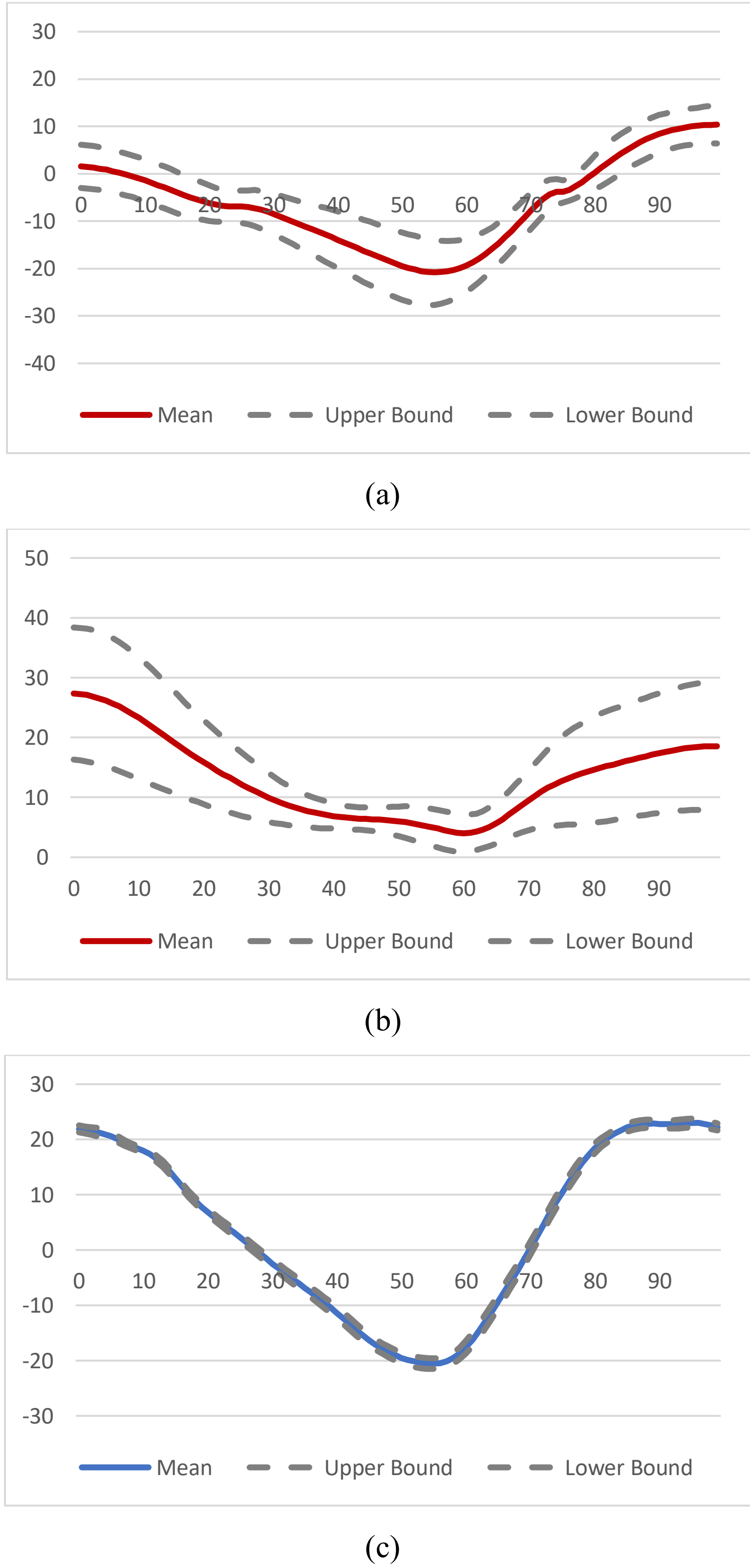
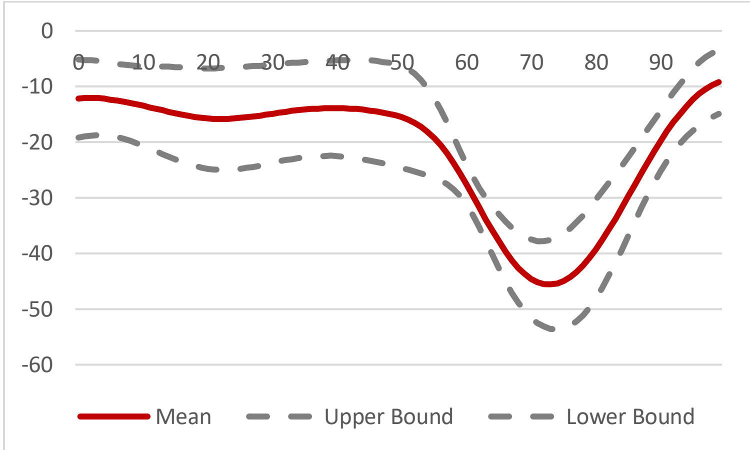
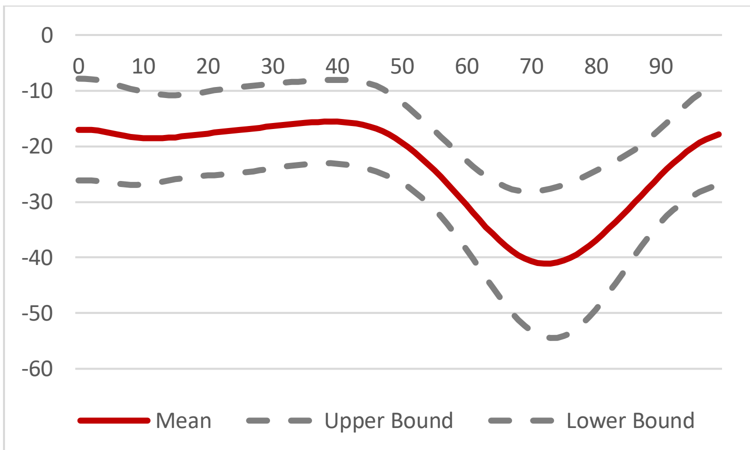


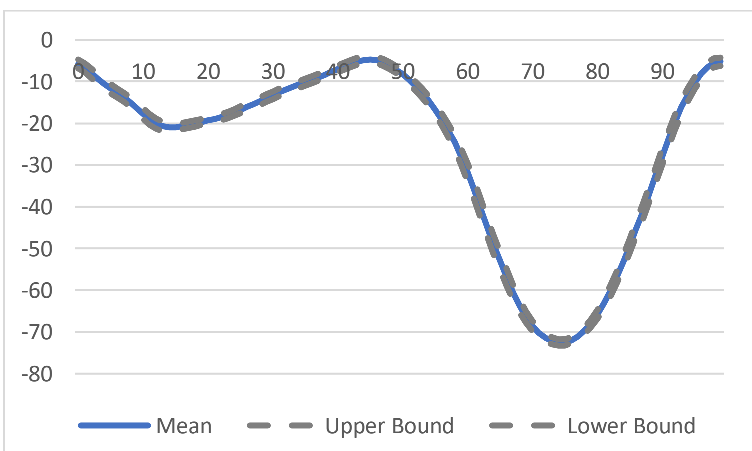
Figure 5.8 Ensemble averaged hip angles: a) Walking towards the camera array, b) Walking away from the camera array, c) Vicon normal walking comparator. The horizontal axis is the ensemble averaged time steps, and the vertical axis is the angle in degrees.



(a)

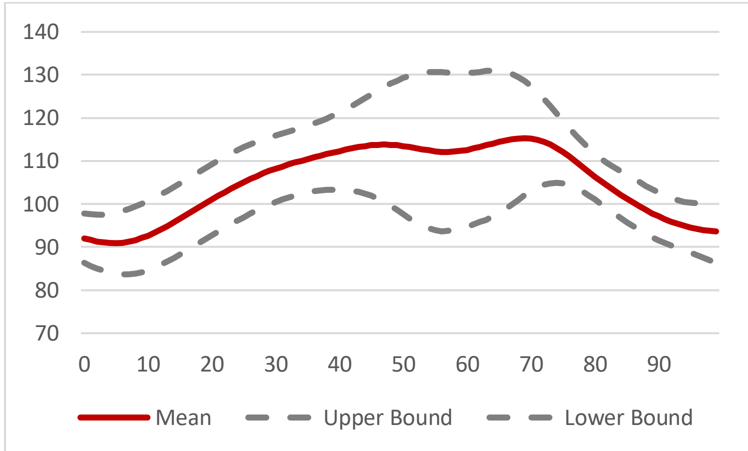


(b)

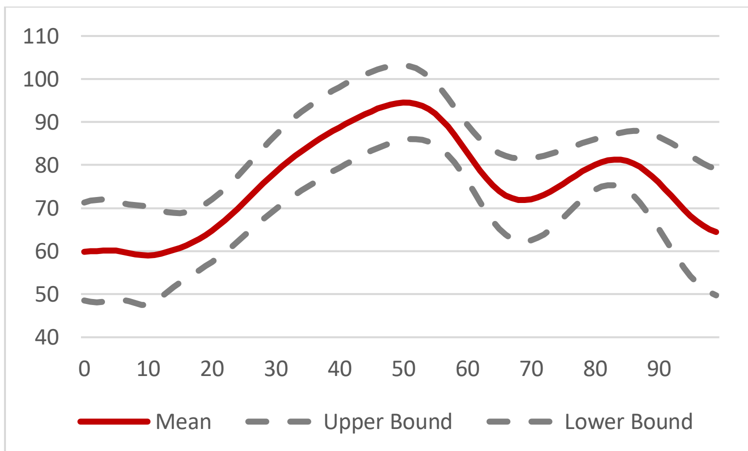


(c)

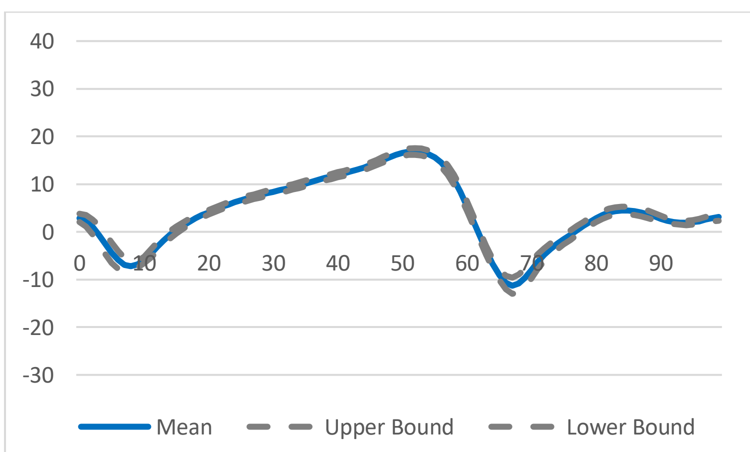
Figure 5.9 Ensemble averaged knee angles: a) Walking towards the camera array, b) Walking away from the camera array, c) Vicon normal walking comparator. The horizontal axis is the ensemble averaged time steps, and the vertical axis is the angle in degrees.



(a)



(b)



(c)

Figure 5.10 Ensemble averaged ankle angles: a) Walking towards the camera array, b) Walking away from the camera array, c) Vicon normal walking comparator. The horizontal axis is the ensemble averaged time steps, and the vertical axis is the angle in degrees.

5.6 Conclusion

A Smart Hallway setup for marker-less human motion analysis was setup in a laboratory to replicate an institutional hallway setting, using an array of four temporally and spatially synchronized cameras. The temporal synchronization was achieved by designing a hardware synchronization cable for the multi-camera array, while the spatial synchronization was achieved through a rigorous calibration procedure using state-of-the-art techniques. A total of twenty-five videos for each participant were captured, including five different walking conditions and containing approximately 500 strides. 3D joint keypoints were calculated from each video. To validate the Smart Hallway's ability to measure body segment lengths, detect foot events, calculate stride parameters, and measure leg angles during gait; ground truth body segment lengths and foot strikes-foot offs were obtained as comparators to calculated outcome measures from the OpenPose generated keypoint data.

Body segment lengths were within 1.56 (SD = 2.77) cm of the ground truth values, on average, across all test conditions. Stride parameters were comparable between keypoint-based and ground truth results. Across all test conditions stride parameters measuring distances were 3.16 (SD = 3.26) cm from the ground truth on average; stride parameters measuring time were 0.047 (SD = 0.037) s from the ground truth on average; and stride parameters measuring velocity were 0.74 (SD = 0.75) cm/s from the ground truth on average. Stride parameters were affected by the walking aids used; however, results were still comparable with measures from the walking straight condition. The ensemble average curves obtained from the leg angle calculations showed similar shapes compared to ensemble averages taken from gold standard Vicon methods.

The Smart Hallway produced accurate results across all outcome measures with small variations. Foot event detection can be improved to reduce temporal synchronization differences between the detected and ground truth events. Improvements to the OpenPose BODY25 model (or other models) will aid in more accurately detecting the feet and handling body part occlusion. Currently, OpenPose's processing is a system bottleneck; results from a 10 second trial can be returned after approximately 120 seconds. The current implementation is limited to one person in frame at a time. However, with upgrades to existing outcome measurement software, groups of people could be processed.

The Smart Hallway was successfully deployed in a manner that was non-invasive to the hallway environment. This means that it could be setup and left to gather data on multiple individuals that walk throughout its capture volume daily, without obstructing any existing institutional processes. The Smart Hallway system was able to successfully capture data from individuals in a non-invasive manner that did not require markers, or a data collection device being affixed to the participants.

6 Thesis Summary and Conclusions

This thesis aimed to design and validate a Smart Hallway for marker-less AI-based human motion analysis. The OpenPose BODY25 AI model was selected for the Smart Hallway based on the quality of its results when applied to relevant clinical movement videos. An institutional hallway was simulated to determine camera positioning, camera pose, and system components needed to create a multi-camera capture array. The components were selected based on system capture requirements and connections were detailed to explain the physical setup of the final Smart Hallway. An approach to camera temporal synchronizations was designed and validated for this system and camera parameter calibration was performed using state-of-the-art techniques from the literature. The final system deployed for this work was a four-camera arc layout capturing at 1440×1080 at 60 fps. Cameras were synchronized within 5 μs and 3D reprojection error was less than 1.0 pixel. The cameras were setup in a lab using the four-camera arc layout based on results from the institutional hallway simulation. Participants completed five repetitions of each of the following trials: natural-pace straight walk, natural-pace walk with turn at the end, natural-pace walk along a curved path, natural-pace straight walk using a walker, and natural-pace straight walk using hand-crutches. The validation data comprised approximately 500 strides for analysis. Based on results to date from the outcome measure validation, the Smart Hallway provides useful data for clinical decision making when used in the constrained test environment during the system validation. Future work could improve some existing system limitations relating to processing time, and the handling of groups of people for outcome measurement tasks. The thesis objectives were outlined and results for each are detailed below:

Objective 1: Determine an appropriate artificial intelligence model for pose inference within an institutional hallway.

Two high-end open-source pose inference models were compared based on their output quality when applied to clinically relevant movements. OpenPose BODY25 outperformed HyperPose COCOv2 ($p<0.001$) when compared between videos after scoring the relevant keypoints. Keypoint filtering and interpolation methods were compared in order to select a post-processing approach for the output 2D keypoint data. OpenPose BODY25 was selected for further usage in the thesis work along with an interpolation routine for missing or poor quality keypoints.

Objective 2: Design hardware and software for a multi-camera image collection and processing system.

A 3D simulation of an institutional hallway was created, and various camera layouts were tested to determine total capture volume and ground area coverage. The capture volume was 29 m^3 and the ground area coverage was 11 m^2. Components were selected based on physical constraints and desired output quality. Camera mounting, synchronization cabling, and system power were developed to ensure that the final system would be non-invasive wherever it is deployed. A method for camera synchronization was designed and validated through various tests. Cameras remained within 5 μs of the desired capture rate and tested at up to 100 fps. Camera intrinsic and stereo-camera extrinsic parameters were calibrated such that they were less than 1.0 pixel.

Objective 3: Validate Smart Hallway performance through analysis of stride parameters and limb lengths obtained from participants.

Body segment lengths, stride parameters, and gait leg angles were analysed to determine the Smart Hallway's capability for human motion analysis. Body segment lengths, across all test conditions, were within 1.56 (SD = 2.77) cm of ground truth values on average. Foot event detection was within 4 frames (67 ms) of the ground truth foot events. Stride parameters obtained using detected foot events were similar to stride parameters calculated using ground truth foot events. More work is needed to improve foot event detection accuracy when calculating stride parameters that rely on multiple data types, such as stance swing ratio or contralateral step parameters. Ensemble average leg angles were calculated and compared to gold standard Vicon results and showed similar overall shape.

Thesis Contributions: This thesis produced several important contributions for knowledge and marker-less motion analysis technology.

- Created the first viable Smart Hallway system for 3D pose inference in an institutional hallway. Through design, development, and analysis, a basis was provided for integrating 3D motion analysis in eldercare and clinical environments.
  - Components were selected based on design criteria and deployed non-invasively in a simulated hallway environment. The multi-camera array was temporally and spatially synchronized using a robust calibration routine. The Smart Hallway was used to successfully calculate various outcome measures.
- Determined that OpenPose BODY25 is superior to HyperPose COCOv2 for marker-less clinically relevant human motion assessment.
  - OpenPose BODY25 provided better keypoint quality compared to HyperPose COCOv2. OpenPose BODY25 required more memory and processing time compared to the TensorRT implementation of HyperPose COCOv2. Overall, OpenPose BODY25 was selected as the superior model and implemented with the Smart Hallway system because accurate keypoint locations were required to produce viable outcome measures.
- Determined that, for a multi-camera system, keypoints connected to multiple segments (i.e., keypoints that are not endpoints) are more accurate. Therefore, a model with multiple connected segments that are constrained relative to each other is recommended for any application that reports stride parameters.
  - The forearms and feet were terminated body segments, with an average segment length standard deviation of 3.66 cm, compared to other body segments that had an average standard deviation of 2.14 cm. Walking aids were also found to interfere with OpenPose's keypoint placement. Improvements could be made to the AI model to improve inferences on people with walking aids, specifically the upper limbs that interact with the assistive device and lower limbs that are occluded by the device.

- Established a viable method to implement synchronization and calibration that can be easily implemented in institutional environments.
    - The hardware synchronization cable designed for this work and ChArUco calibration implementation provided highly accurate results for time synchronization and for 3D reconstruction that can be easily reproduced in future systems. The Smart Hallway consistently maintains a microsecond synchronization level and an average position error of 3.5 (SD=2.4) cm across multiple trials within the $29m^3$ capture volume.
- Determined that the Zeni et al. [13] foot event detection algorithm was appropriate for use with the Smart Hallway's 3D marker-less foot tracking keypoints.
    - The Zeni et al. [13] foot event detection algorithm performed well with the marker-less method. Initiation and Termination Recovery and Capela et al. [113] algorithms further improved foot event detection results. The foot event detection absolute error was 4 frames (67 ms) with an error of 3 frames (50 ms) on the ground truth labelling.
- Determined that the Smart Hallway outcome measures (body segment lengths, stride parameters, joint angles, and limb angles) were viable for use in clinical decision-making.
    - Body segment lengths were on average 1.56 (SD = 2.77) cm different from the ground truth measurements. Stride parameters were within a reasonable range compared to the parameters calculated using ground truth foot events. Joint and limb angles over time had similar shapes compared to Vicon comparator data. The outcome measures provided useful information for understanding how the participant moved.

6.1 Future Work

6.1.1 Transfer Learning

The current OpenPose BODY25 AI model was found to be the best opensource model available for semantic keypoint allocation. Further improvements could be made to this model to improve performance in the institutional hallway setting. Two main factors relating to the Smart Hallway implementation effect the model's current performance; participant pose relative to the camera frame and robustness of the foot model. Training data used for OpenPose primarily contains images of camera-facing individuals. This contrasts with the image streams used in this study where the participant is at an angle relative to the camera frame. The BODY25 foot model was added by labelling approximately 5% of the original COCO dataset with foot annotations [114]. A potential solution to both problems is to improve the model via transfer learning, providing images with varying person poses relative to the camera frame and to increase the number of training data points containing the foot model. Combined with this approach, more data containing instances of people using walking aids could be included to improve the accuracy in these cases.

6.1.2 AI Model Physical and Kinematic Constraints

The current OpenPose BODY25 AI model does not account for physical and kinematic constraints such as consistent joint-to-joint segment lengths and range of motion of certain joints. Some recent approaches to keypoint pose inference models, such as MotioNet, have included encodings for bone lengths and 3D joint rotation [115]. These models rely on a scaled estimation of the depth coordinate that is learned through AI training processes that are also seen in Google ML Kit Pose Detection [92]. Future work could include these models along with tradition 3D reconstruction techniques to determine their viability.

6.1.3 GPU Processes Optimization

Some Smart Hallway pipelines could be further optimized to improve live processing time for future implementations. The OpenPose BODY25 model is not currently implemented using NVIDIA's TensorRT model optimization framework and processing time may be substantially improved. Work could also be done to implement OpenCV's GPU version with the HPC unit. Currently, the OpenCV GPU version is not directly supported by NVIDIA's Jetson units.

6.1.4 Physical Placement Optimization

The current physical setup was selected based on 3D simulations of an institutional hallway. An optimization procedure for camera placement and pose could aid in improving outcome measure accuracy from the Smart Hallway by improving point correspondences between cameras. This optimization procedure could be constrained by parameters such as the desired capture volume, ground area coverage, and number of cameras. To better implement these results motorized mounts could be used to accurately position cameras after the optimization procedure.

6.1.5 Camera Capture Synchronization

Camera synchronization is currently achieved using a custom hardware synchronization cable. This cable could be further improved by running power lines along with the synchronization signal to reduce the number of power connections needed along the length of the Smart Hallway. Synchronization could also be handled by a signal generator as opposed to selecting a primary camera, which could provide centralized synchronization and camera power unit alongside the Jetson AGX Xavier.

6.1.6 Improvement of 3D Reconstruction

Currently, 3D reconstruction is achieved using a multi-camera triangulation procedure with bundle adjustment for optimization. Depth could be improved using an AI model for depth estimation along with the current algorithm to reduce error in the Z-coordinate of each 3D point. Reconstruction results could also be improved by implementing a point correspondence correction algorithm.

6.1.7 Gait Classification Applications

The current Smart Hallway gathers 3D points and produces stride parameters as clinical outcome measures. This could be further built on by developing a cascaded set of AI models that enable outcome measure data to be used for further inference. These models could allow the Smart Hallway to also provide identification for fall risk, progressive brain disorders such as dementia or Alzheimer's, and classification of disorder progression.

6.1.8 3D Modelling of Participants

Validation of the segment length, or stride parameter results, could be performed by producing a visual 3D reconstruction of the OpenPose BODY25 keypoints obtained from the participant walking through the capture volume. This could provide a tool for further analysis from clinicians by allowing them to manipulate a 3D skeleton model of the participant.

References

[1] W. Tao, T. Liu, R. Zheng, and H. Feng, "Gait analysis using wearable sensors," *Sensors*, vol. 12, no. 2, pp. 2255–2283, Feb. 2012, doi: 10.3390/s120202255.

[2] P. Merriaux, Y. Dupuis, R. Boutteau, P. Vasseur, and X. Savatier, "A study of Vicon system positioning performance," *Sensors*, vol. 17, no. 7, p. 1591, Jul. 2017, doi: 10.3390/s17071591.

[3] P. Jenpoomjai, P. Wosri, S. Ruengittinun, C. L. Hu, and C. Chootong, "VA algorithm for elderly's falling detection with 2d-pose-estimation," in *Proceedings - 2019 12th International Conference on Ubi-Media Computing, Ubi-Media 2019*, Aug. 2019, pp. 236–240. doi: 10.1109/Ubi-Media.2019.00053.

[4] L. Anishchenko, "Machine learning in video surveillance for fall detection," in *Proceedings - 2018 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology, USBEREIT 2018*, Jun. 2018, pp. 99–102. doi: 10.1109/USBEREIT.2018.8384560.

[5] X. Wang, M. Kyrarini, D. Ristić-Durrant, M. Spranger, and A. Gräser, "Monitoring of gait performance using dynamic time warping on IMU-sensor data," Aug. 2016. doi: 10.1109/MeMeA.2016.7533745.

[6] S. Corazza, L. Mündermann, A. M. Chaudhari, T. Demattio, C. Cobelli, and T. P. Andriacchi, "A marker-less motion capture system to study musculoskeletal biomechanics: Visual hull and simulated annealing approach," *Annals of Biomedical Engineering*, vol. 34, no. 6, pp. 1019–1029, Jun. 2006, doi: 10.1007/s10439-006-9122-8.

[7] A. Rampp, J. Barth, S. Schülein, K. G. Gaßmann, J. Klucken, and B. M. Eskofier, "Inertial sensor-based stride parameter calculation from gait sequences in geriatric patients," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 4, pp. 1089–1097, Apr. 2015, doi: 10.1109/TBME.2014.2368211.

[8] E. Auvinet, F. Multon, C. E. Aubin, J. Meunier, and M. Raison, "Detection of gait cycles in treadmill walking using a Kinect," *Gait and Posture*, vol. 41, no. 2, pp. 722–725, Feb. 2015, doi: 10.1016/j.gaitpost.2014.08.006.

[9] A. Viswakumar, V. Rajagopalan, T. Ray, and C. Parimi, "Human gait analysis using OpenPose," in *Proceedings of the IEEE International Conference Image Information Processing*, Nov. 2019, vol. 2019-Novem, pp. 310–314. doi: 10.1109/ICIIP47207.2019.8985781.

[10] A. Joshi, S. Kale, S. Chandel, and D. Pal, "Likert Scale: Explored and explained," *British Journal of Applied Science & Technology*, vol. 7, no. 4, pp. 396–403, Jan. 2015, doi: 10.9734/bjast/2015/14975.

[11] A. Patrizi, E. Pennestrì, and P. P. Valentini, "Comparison between low-cost marker-less and high-end marker-based motion capture systems for the computer-aided assessment of working ergonomics," *Ergonomics*, vol. 59, no. 1, pp. 155–162, 2016, doi: 10.1080/00140139.2015.1057238.

[12] P. Karashchuk *et al.*, "Anipose: a toolkit for robust marker-less 3D pose estimation," *bioRxiv*, p. 2020.05.26.117325, May 2020, doi: 10.1101/2020.05.26.117325.

[13] J. A. Zeni, J. G. Richards, and J. S. Higginson, "Two simple methods for determining gait events during treadmill and overground walking using kinematic data," *Gait and Posture*, vol. 27, no. 4, pp. 710–714, May 2008, doi: 10.1016/j.gaitpost.2007.07.007.

[14] R. Baker, A. Esquenazi, M. G. Benedetti, and K. Desloovere, "Gait analysis: Clinical facts," *European Journal of Physical and Rehabilitation Medicine*, vol. 52, no. 4, pp. 560–574, Aug. 2016.

[15] R. Poppe, "Vision-based human motion analysis: An overview," *Computer Vision and Image Understanding*, vol. 108, no. 1–2, pp. 4–18, Oct. 2007, doi: 10.1016/j.cviu.2006.10.016.

[16] M. E. Taylor, K. Delbaere, A. S. Mikolaizak, S. R. Lord, and J. C. T. Close, "Gait parameter risk factors for falls under simple and dual task conditions in cognitively impaired older people," *Gait and Posture*, vol. 37, no. 1, pp. 126–130, Jan. 2013, doi: 10.1016/j.gaitpost.2012.06.024.

[17] A. Schinkel-Ivy, J. S. Wong, and A. Mansfield, "Balance confidence is related to features of balance and gait in individuals with chronic stroke," *Journal of Stroke and Cerebrovascular Diseases*, vol. 26, no. 2, pp. 237–245, Feb. 2017, doi: 10.1016/j.jstrokecerebrovasdis.2016.07.022.

[18] O. Beauchet, G. Allali, C. Launay, F. R. Herrmann, and C. Annweiler, "Gait variability at fast-pace walking speed: A biomarker of mild cognitive impairment?," *Journal of Nutrition, Health and Aging*, vol. 17, no. 3, pp. 235–239, Mar. 2013, doi: 10.1007/s12603-012-0394-4.

[19] R. Caldas, M. Mundt, W. Potthast, F. Buarque de Lima Neto, and B. Markert, "A systematic review of gait analysis methods based on inertial sensors and adaptive algorithms," *Gait and Posture*, vol. 57. Elsevier B.V., pp. 204–210, Sep. 01, 2017. doi: 10.1016/j.gaitpost.2017.06.019.

[20] "Orthopedic physical assessment - 6th edition." https://www.elsevier.com/books/orthopedic-physical-assessment/magee/978-1-4557-0977-9 (accessed Feb. 18, 2021).

[21] A. Muro-de-la-Herran, B. García-Zapirain, and A. Méndez-Zorrilla, "Gait analysis methods: An overview of wearable and non-wearable systems, highlighting clinical applications," *Sensors (Switzerland)*, vol. 14, no. 2. Multidisciplinary Digital Publishing Institute (MDPI), pp. 3362–3394, Feb. 19, 2014. doi: 10.3390/s140203362.

[22] Y. C. Han, K. I. Wong, and I. Murray, "Gait phase detection for normal and abnormal gaits using IMU," *IEEE Sensors Journal*, vol. 19, no. 9, pp. 3439–3448, May 2019, doi: 10.1109/JSEN.2019.2894143.

[23] T. Liu, Y. Inoue, and K. Shibata, "Development of a wearable sensor system for quantitative gait analysis," *Measurement: Journal of the International Measurement Confederation*, vol. 42, no. 7, pp. 978–988, Aug. 2009, doi: 10.1016/j.measurement.2009.02.002.

[24] T. Cloete and C. Scheffer, "Benchmarking of a full-body inertial motion capture system for clinical gait analysis," in *Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS'08 - "Personalized Healthcare through Technology,"* 2008, vol. 2008, pp. 4579–4582. doi: 10.1109/iembs.2008.4650232.

[25] C. M. O'Connor, S. K. Thorpe, M. J. O'Malley, and C. L. Vaughan, "Automatic detection of gait events using kinematic data," *Gait and Posture*, vol. 25, no. 3, pp. 469–474, Mar. 2007, doi: 10.1016/j.gaitpost.2006.05.016.

[26] K. E. Webster, J. E. Wittwer, and J. A. Feller, "Validity of the GAITRite® walkway system for the measurement of averaged and individual step parameters of gait," *Gait and Posture*, vol. 22, no. 4, pp. 317–321, Dec. 2005, doi: 10.1016/j.gaitpost.2004.10.005.

[27] B. J. Braun *et al.*, "Validation and reliability testing of a new, fully integrated gait analysis insole," *Journal of Foot and Ankle Research*, vol. 8, no. 1, pp. 1–7, Sep. 2015, doi: 10.1186/s13047-015-0111-8.

[28] A. De Luca *et al.*, "Exoskeleton for gait rehabilitation: Effects of assistance, mechanical structure, and walking aids on muscle activations," *Applied Sciences*, vol. 9, no. 14, p. 2868, Jul. 2019, doi: 10.3390/app9142868.

[29] R. G. Cutlip, C. Mancinelli, F. Huber, and J. Dipasquale, "Evaluation of an instrumented walkway for measurement of the kinematic parameters of gait," *Gait and Posture*, vol. 12, no. 2, pp. 134–138, Oct. 2000, doi: 10.1016/S0966-6362(00)00062-X.

[30] "Smart insoles for gait and motion analysis in research - Moticon." https://www.moticon.de/ (accessed Feb. 18, 2021).

[31] M. Pau *et al.*, "Clinical assessment of gait in individuals with multiple sclerosis using wearable inertial sensors: Comparison with patient-based measure," *Multiple Sclerosis and Related Disorders*, vol. 10, pp. 187–191, Nov. 2016, doi: 10.1016/j.msard.2016.10.007.

[32] A. Pfister, A. M. West, S. Bronner, and J. A. Noah, “Comparative abilities of Microsoft Kinect and Vicon 3D motion capture for gait analysis,” *Journal of Medical Engineering & Technology*, vol. 38, no. 5, pp. 274–280, Jul. 2014, doi: 10.3109/03091902.2014.909540.

[33] “Vicon in use | Case studies | Motion capture systems.” http://www.vicon.com/resources/case-studies/the-gait-lab-of-the-future/ (accessed Feb. 03, 2021).

[34] A. Laurentini, “The visual hull concept for silhouette-based image understanding,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 150–162, 1994, doi: 10.1109/34.273735.

[35] T. Alldieck, M. Magnor, W. Xu, C. Theobalt, and G. Pons-Moll, “Video based reconstruction of 3D people models,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8387–8397. doi: 10.1109/CVPR.2018.00875.

[36] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, “Deep learning for 3D point clouds: A survey,” *arXiv*. arXiv, Dec. 27, 2019. doi: 10.1109/tpami.2020.3005434.

[37] A. Behl, D. Paschalidou, S. Donné, and A. Geiger, “PointFlowNet: Learning representations for rigid motion estimation from point clouds,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2019-June, pp. 7954–7963, Jun. 2018.

[38] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 386–397, 2020, doi: 10.1109/TPAMI.2018.2844175.

[39] T. Wolf, M. Babaee, and G. Rigoll, “Multi-view gait recognition using 3D convolutional neural networks,” in *Proceedings - International Conference on Image Processing, ICIP*, 2016, pp. 4165–4169.

[40] K. Shiraga, Y. Makihara, D. Muramatsu, T. Echigo, and Y. Yagi, “GEINet: View-invariant gait recognition using a convolutional neural network,” 2016. doi: 10.1109/ICB.2016.7550060.

[41] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, “OpenPose: Realtime multi-person 2D pose estimation using part affinity fields,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172–186, Dec. 2018, Accessed: Feb. 03, 2021. [Online]. Available: http://arxiv.org/abs/1812.08008

[42] S. Giancola, M. Valenti, and R. Sala, “Metrological qualification of the intel D400™ active stereoscopy cameras,” in *SpringerBriefs in Computer Science*, Springer, 2018, pp. 71–85. doi: 10.1007/978-3-319-91761-0_6.

[43] V. Gutta, “Development and validation of a smart hallway for human stride analysis using marker-less 3D depth sensors,” 2020. Accessed: Feb. 16, 2021. [Online]. Available: https://ruor.uottawa.ca/handle/10393/40266

[44] D. Jiménez, D. Pizarro, M. Mazo, and S. Palazuelos, “Modeling and correction of multipath interference in time of flight cameras,” *Image and Vision Computing*, vol. 32, no. 1, pp. 1–13, Jan. 2014, doi: 10.1016/j.imavis.2013.10.008.

[45] S. Fuchs, “Multipath interference compensation in time-of-flight camera images,” in *Proceedings - International Conference on Pattern Recognition*, 2010, pp. 3583–3586. doi: 10.1109/ICPR.2010.874.

[46] J. Marco *et al.*, “DeepToF: Off-the-shelf real-time correction of multipath interference in time-of-flight imaging,” in *ACM Transactions on Graphics*, Nov. 2017, vol. 36, no. 6, pp. 1–12. doi: 10.1145/3130800.3130884.

[47] W. Zong *et al.*, “The method of multi-camera layout in motion capture system for diverse small animals,” *Applied Sciences*, vol. 8, no. 9, p. 1562, Sep. 2018, doi: 10.3390/app8091562.

[48] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H. P. Seidel, "Motion capture using joint skeleton tracking and surface estimation," in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, 2009, vol. 2009 IEEE, pp. 1746–1753. doi: 10.1109/CVPRW.2009.5206755.

[49] L. E. Ortiz, E. V Cabrera, and L. M. Gonçalves, "Depth data error modeling of the ZED 3D vision sensor from Stereolabs," *Electronic Letters on Computer Vision and Image Analysis*, vol. 17, no. 1, pp. 1–15, Jun. 2018, doi: 10.5565/rev/elcvia.1084.

[50] A. Jigalin and P. Gurfil, "Investigation of multiple-baseline stereovision for state estimation of unknown dynamic space targets," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 230, no. 2, pp. 207–233, Feb. 2016, doi: 10.1177/0954410015590636.

[51] I. Alhashim and P. Wonka, "High quality monocular depth estimation via transfer learning," *arXiv*, Dec. 2018.

[52] C. Q. Zhao, Q. Y. Sun, C. Z. Zhang, Y. Tang, and F. Qian, "Monocular depth estimation based on deep learning: An overview," *Science China Technological Sciences*, vol. 63, no. 9. Springer Verlag, pp. 1612–1627, Sep. 01, 2020. doi: 10.1007/s11431-020-1582-8.

[53] J. Zhu and Y. Fang, "Learning object-specific distance from a monocular image," in *Proceedings of the IEEE International Conference on Computer Vision*, Sep. 2019, vol. 2019-Octob, pp. 3838–3847. doi: 10.1109/ICCV.2019.00394.

[54] R. T. Labuguen, W. Enrico, M. Ingco, S. B. Negrete, T. Kogami, and T. Shibata, "Performance Evaluation of Markerless 3D Skeleton Pose Estimates with Pop Dance Motion Sequence", doi: 10.1101/2020.04.15.010702.

[55] T. B. Rodrigues, C. Catháin, D. Devine, K. Moran, N. E. O'Connor, and N. Murray, "An evaluation of a 3D multimodal marker-less motion analysis system," *Proceedings of the 10th ACM Multimedia Systems Conference, MMSys 2019*, pp. 213–221, Jun. 2019, doi: 10.1145/3304109.3306236.

[56] N. Nakano *et al.*, "Evaluation of 3D Markerless Motion Capture Accuracy Using OpenPose With Multiple Video Cameras," *Frontiers in Sports and Active Living*, vol. 0, p. 50, May 2020, doi: 10.3389/FSPOR.2020.00050.

[57] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017, doi: 10.1145/3065386.

[58] V. Powell, "Image kernels explained visually," 2017. https://setosa.io/ev/image-kernels/ (accessed Feb. 18, 2021).

[59] G. Fanelli, T. Weise, J. Gall, and L. Van Gool, "Real time head pose estimation from consumer depth cameras," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2011, vol. 6835 LNCS, pp. 101–110. doi: 10.1007/978-3-642-23123-0_11.

[60] K. Saleh, M. Hossny, A. Hossny, and S. Nahavandi, "Cyclist detection in LIDAR scans using faster R-CNN and synthetic depth images," in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, Mar. 2018, vol. 2018-March, pp. 1–6. doi: 10.1109/ITSC.2017.8317599.

[61] "Technology - WRNCH." https://wrnch.ai/technology/ (accessed Feb. 08, 2021).

[62] J. W. Chin, T. W. Wong, and R. H. Y. So, "Innovations in crowd management: An integration of visual closure, anthropometry, and computer vision," in *Advances in Intelligent Systems and Computing*, Aug. 2019, vol. 822, pp. 68–74. doi: 10.1007/978-3-319-96077-7_8.

[63] "tfjs-models/body-pix at master, tensorflow/tfjs-models." https://github.com/tensorflow/tfjs-models/tree/master/body-pix (accessed Feb. 08, 2021).

[64] U. Solichah, M. H. Purnomo, and E. M. Yuniarno, "Marker-less motion capture based on OpenPose model using triangulation," in *Proceedings - 2020 International Seminar on Intelligent Technology and Its Application: Humanification of Reliable Intelligent Systems, ISITIA 2020*, Jul. 2020, pp. 217–222. doi: 10.1109/ISITIA49792.2020.9163662.

[65] “hampen2929/pyvino: This is the python implementation of openvino models.” https://github.com/hampen2929/pyvino (accessed Feb. 18, 2021).

[66] G. Hidalgo *et al.*, “Single-network whole-body pose estimation,” *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2019-Octob, pp. 6981–6990, Sep. 2019.

[67] “tensorlayer/hyperpose: HyperPose: A collection of real-time human pose estimation.” https://github.com/tensorlayer/hyperpose (accessed Feb. 15, 2021).

[68] J. Cui *et al.*, “Wearable Gait Lab System providing quantitative statistical support for human balance tests,” *Smart Health*, vol. 3–4, pp. 27–38, Sep. 2017, doi: 10.1016/j.smhl.2017.05.001.

[69] “USB 3.1 multiple camera setup | FLIR systems.” https://www.flir.ca/support-center/iis/machine-vision/application-note/usb-3.1-multiple-camera-setup/ (accessed Feb. 15, 2021).

[70] S. Mittal, “A Survey on optimized implementation of deep learning models on the NVIDIA Jetson platform,” *Journal of Systems Architecture*, vol. 97. Elsevier B.V., pp. 428–442, Aug. 01, 2019. doi: 10.1016/j.sysarc.2019.01.011.

[71] R. Giubilato, S. Chiodini, M. Pertile, and S. Debei, “An evaluation of ROS-compatible stereo visual SLAM methods on a nVidia Jetson TX2,” *Measurement: Journal of the International Measurement Confederation*, vol. 140, pp. 161–170, Jul. 2019, doi: 10.1016/j.measurement.2019.03.038.

[72] A. T. Wan and S. Sankaranarayanan, “Development of a health information system in the mobile cloud environment,” in *Proceedings - 2013 IEEE International Conference on High Performance Computing and Communications, HPCC 2013 and 2013 IEEE International Conference on Embedded and Ubiquitous Computing, EUC 2013*, 2014, pp. 2187–2193. doi: 10.1109/HPCC.and.EUC.2013.313.

[73] C. Heard, “USB 3.0 empowers embedded vision: USB3 vision has a number of benefits for industrial and embedded vision applications,” *Quality*, vol. 53, no. 3, pp. 10VS-10VS, Mar. 2014.

[74] "Raspberry Pi 4 Model B specifications – Raspberry Pi." https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/ (accessed Feb. 15, 2021).

[75] "Dev Board | Coral." https://coral.ai/products/dev-board/ (accessed Feb. 15, 2021).

[76] NVIDIA, "Embedded systems developer kits & modules from NVIDIA Jetson," *NVIDIA Developer*, 2020. https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/ (accessed Feb. 15, 2021).

[77] P. Sturm and S. Ramalingam, "A generic concept for camera calibration," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3022, pp. 1–13, 2004, doi: 10.1007/978-3-540-24671-8_1.

[78] B. Poling, "A Tutorial On Camera Models," in *University of Minnesota*, 2012, pp. 1–10.

[79] J. Ponce, "What is a camera?," in *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Mar. 2010, pp. 1526–1533. doi: 10.1109/cvpr.2009.5206668.

[80] H. Louhichi, T. Fournel, J. M. Lavest, and H. Ben Aissia, "Self-calibration of Scheimpflug cameras: An easy protocol," *Measurement Science and Technology*, vol. 18, no. 8, pp. 2616–2622, Aug. 2007, doi: 10.1088/0957-0233/18/8/037.

[81] "Camera calibration toolbox for Matlab." http://www.vision.caltech.edu/bouguetj/calib_doc/ (accessed Feb. 16, 2021).

[82] J. Courbon, Y. Mezouar, L. Eck, and P. Martinet, "A generic fisheye camera model for robotic applications," in *IEEE International Conference on Intelligent Robots and Systems*, 2007, pp. 1683–1688. doi: 10.1109/IROS.2007.4399233.

[83] J. Weng, P. Coher, and M. Herniou, "Camera Calibration with Distortion Models and Accuracy Evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 10, pp. 965–980, 1992, doi: 10.1109/34.159901.

[84] J. Wang, F. Shi, J. Zhang, and Y. Liu, "A new calibration model of camera lens distortion," *Pattern Recognition*, vol. 41, no. 2, pp. 607–615, Feb. 2008, doi: 10.1016/j.patcog.2007.06.012.

[85] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000, doi: 10.1109/34.888718.

[86] B. Li, L. Heng, K. Koser, and M. Pollefeys, "A multiple-camera system calibration toolbox using a feature descriptor-based calibration pattern," in *IEEE International Conference on Intelligent Robots and Systems*, 2013, pp. 1301–1307. doi: 10.1109/IROS.2013.6696517.

[87] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, Jun. 2014, doi: 10.1016/j.patcog.2014.01.005.

[88] T. Sato, S. Ikeda, and N. Yokoya, "Extrinsic camera parameter recovery from multiple image sequences captured by an omni-directional multi-camera system," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3022, pp. 326–340, 2004, doi: 10.1007/978-3-540-24671-8_26.

[89] A. Bartoli and P. Sturm, "Structure-from-motion using lines: Representation, triangulation, and bundle adjustment," *Computer Vision and Image Understanding*, vol. 100, no. 3, pp. 416–441, Dec. 2005, doi: 10.1016/j.cviu.2005.06.001.

[90] M. Vo, S. G. Narasimhan, and Y. Sheikh, "Spatiotemporal bundle adjustment for dynamic 3D reconstruction," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Dec. 2016, vol. 2016-Decem, pp. 1710–1718. doi: 10.1109/CVPR.2016.189.

[91] K. Yang, W. Fang, Y. Zhao, and N. Deng, "Iteratively reweighted midpoint method for fast multiple view triangulation," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 708–715, Apr. 2019, doi: 10.1109/LRA.2019.2893022.

[92] O. Chum and J. Matas, "Optimal randomized RANSAC," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 8, pp. 1472–1482, Aug. 2008, doi: 10.1109/TPAMI.2007.70787.

[93] "Pose classification options | ML Kit | Google Developers." https://developers.google.com/ml-kit/vision/pose-detection (accessed May 06, 2021).

[94] M. R. Patterson *et al.*, "Does external walking environment affect gait patterns?," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2014*, Nov. 2014, pp. 2981–2984. doi: 10.1109/EMBC.2014.6944249.

[95] J. M. Goh, G. Gao, and R. Agarwal, "Evolving work routines: Adaptive routinization of information technology in healthcare," *Information Systems Research*, vol. 22, no. 3, pp. 565–585, 2011, doi: 10.1287/isre.1110.0365.

[96] J. S. Ancker, L. M. Kern, E. Abramson, and R. Kaushal, "The triangle model for evaluating the effect of health information technology on healthcare quality and safety," *Journal of the American Medical Informatics Association*, vol. 19, no. 1, pp. 61–65, Jan. 2012, doi: 10.1136/amiajnl-2011-000385.

[97] J. Colley, H. Zeeman, and E. Kendall, "'Everything happens in the hallways': Exploring user activity in the corridors at two rehabilitation units," *Health Environments Research and Design Journal*, vol. 11, no. 2, pp. 163–176, Apr. 2018, doi: 10.1177/1937586717733149.

[98] C. Hahne, A. Aggoun, V. Velisavljevic, S. Fiebig, and M. Pesch, "Baseline and triangulation geometry in a standard plenoptic camera," *International Journal of Computer Vision*, vol. 126, no. 1, pp. 21–35, Jan. 2018, doi: 10.1007/s11263-017-1036-4.

[99] "Security lenses | Fujifilm Global." https://www.fujifilm.com/products/optical_devices/cctv/ (accessed Feb. 21, 2021).

[100] G. Olague and R. Mohr, "Optimal camera placement for accurate reconstruction," *Pattern Recognition*, vol. 35, no. 4, pp. 927–944, Apr. 2002, doi: 10.1016/S0031-3203(01)00076-0.

[101] "Configuring synchronized capture with multiple cameras | FLIR Systems." https://www.flir.ca/support-center/iis/machine-vision/application-note/configuring-synchronized-capture-with-multiple-cameras/ (accessed Feb. 23, 2021).

[102] "GPIO electrical characteristics BFS-U3-16S2." http://softwareservices.flir.com/BFS-U3-16S2/latest/Family/ElectricalGPIO.htm?Highlight=BFS-U3-16S2 electrical (accessed Feb. 22, 2021).

[103] F. J. Romero-Ramirez, R. Muñoz-Salinas, and R. Medina-Carnicer, "Speeded up detection of squared fiducial markers," *Image and Vision Computing*, vol. 76, pp. 38–47, Aug. 2018, doi: 10.1016/j.imavis.2018.05.004.

[104] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and R. Medina-Carnicer, "Generation of fiducial marker dictionaries using mixed integer linear programming," *Pattern Recognition*, vol. 51, pp. 481–491, Mar. 2016, doi: 10.1016/j.patcog.2015.09.023.

[105] S. Agarwal and K. Mierle, "Ceres Solver — A large scale non-linear optimization library," 2019. http://ceres-solver.org/ (accessed May 06, 2021).

[106] "aniposelib/aniposelib at master, lambdaloop/aniposelib, GitHub." https://github.com/lambdaloop/aniposelib/tree/master/aniposelib (accessed May 09, 2021).

[107] E. H. Sinitski, E. D. Lemaire, N. Baddour, M. Besemann, N. L. Dudek, and J. S. Hebert, "Fixed and self-paced treadmill walking for able-bodied and transtibial amputees in a multi-terrain virtual environment," *Gait and Posture*, vol. 41, no. 2, pp. 568–573, Feb. 2015, doi: 10.1016/j.gaitpost.2014.12.016.

[108] "opencv_contrib/modules/aruco at master, opencv/opencv_contrib, GitHub." https://github.com/opencv/opencv_contrib/tree/master/modules/aruco (accessed May 09, 2021).

[109] "openpose/calibration.cpp at master, CMU-Perceptual-Computing-Lab/openpose, GitHub." https://github.com/CMU-Perceptual-Computing-Lab/openpose/blob/master/examples/calibration/calibration.cpp (accessed May 09, 2021).

[110] "Research Ethics Board (REB) | Office of Research Ethics and Integrity | University of Ottawa." https://research.uottawa.ca/ethics/reb (accessed May 10, 2021).

[111] J. Stenum, C. Rossi, and R. Roemmich, "Two-dimensional video-based analysis of human gait using pose estimation," *bioRxiv*, p. 2020.07.24.218776, Jul. 2020, doi: 10.1101/2020.07.24.218776.

[112] "scipy.signal.find_peaks — SciPy v1.6.3 reference guide." https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.find_peaks.html (accessed May 09, 2021).

[113] N. A. Capela, E. D. Lemaire, and N. C. Baddour, "A smartphone approach for the 2 and 6-minute walk test," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2014*, Nov. 2014, pp. 958–961. doi: 10.1109/EMBC.2014.6943751.

[114] N. A. Capela, E. D. Lemaire, and N. Baddour, "Novel algorithm for a smartphone-based 6-minute walk test application: Algorithm, application development, and evaluation," *Journal of NeuroEngineering and Rehabilitation*, vol. 12, no. 1, Jan. 2015, doi: 10.1186/s12984-015-0013-9.

[115] Microsoft, "COCO - Common objects in context," 2015. https://cocodataset.org/#home (accessed Feb. 15, 2021).

[116] M. Shi *et al.*, "MotioNet: 3D human motion reconstruction from monocular video with skeleton consistency," *ACM Transactions on Graphics*, vol. 40, no. 1, Jun. 2020, doi: 10.1145/3407659.

Appendix A: Research Ethics Board Approval Certificate

08/02/2021

Université d'Ottawa
Bureau d'éthique et d'intégrité de la recherche

University of Ottawa
Office of Research Ethics and Integrity

CERTIFICAT D'APPROBATION ÉTHIQUE | CERTIFICATE OF ETHICS APPROVAL

Numéro du dossier / Ethics File Number	H-01-21-5819
Titre du projet / Project Title	EVALUATION OF A SMART HALLWAY MULTI-CAMERA SYSTEM
Type de projet / Project Type	Thèse de maîtrise / Master's thesis
Statut du projet / Project Status	Approuvé / Approved
Date d'approbation (jj/mm/aaaa) / Approval Date (dd/mm/yyyy)	08/02/2021
Date d'expiration (jj/mm/aaaa) / Expiry Date (dd/mm/yyyy)	07/02/2022

Équipe de recherche / Research Team

Chercheur / Researcher	Affiliation	Role
Connor MCGUIRK	Département de génie mécanique / Department of Mechanical Engineering	Chercheur Principal / Principal Investigator
Natalie BADDOUR	Département de génie mécanique / Department of Mechanical Engineering	Superviseur / Supervisor
Edward LEMAIRE	Département de médecine / Department of Medicine	Co-superviseur / Co-supervisor

Conditions spéciales ou commentaires / Special conditions or comments

550, rue Cumberland, pièce 154
Ottawa (Ontario) K1N 6N5 Canada

550 Cumberland Street, Room 154
Ottawa, Ontario K1N 6N5 Canada

613-562-5387 • 613-562-5338 • ethique@uOttawa.ca / ethics@uOttawa.ca
www.recherche.uottawa.ca/deontologie | www.recherche.uottawa.ca/ethics

Appendix B: Hardware Synchronization Cable Signal Analysis

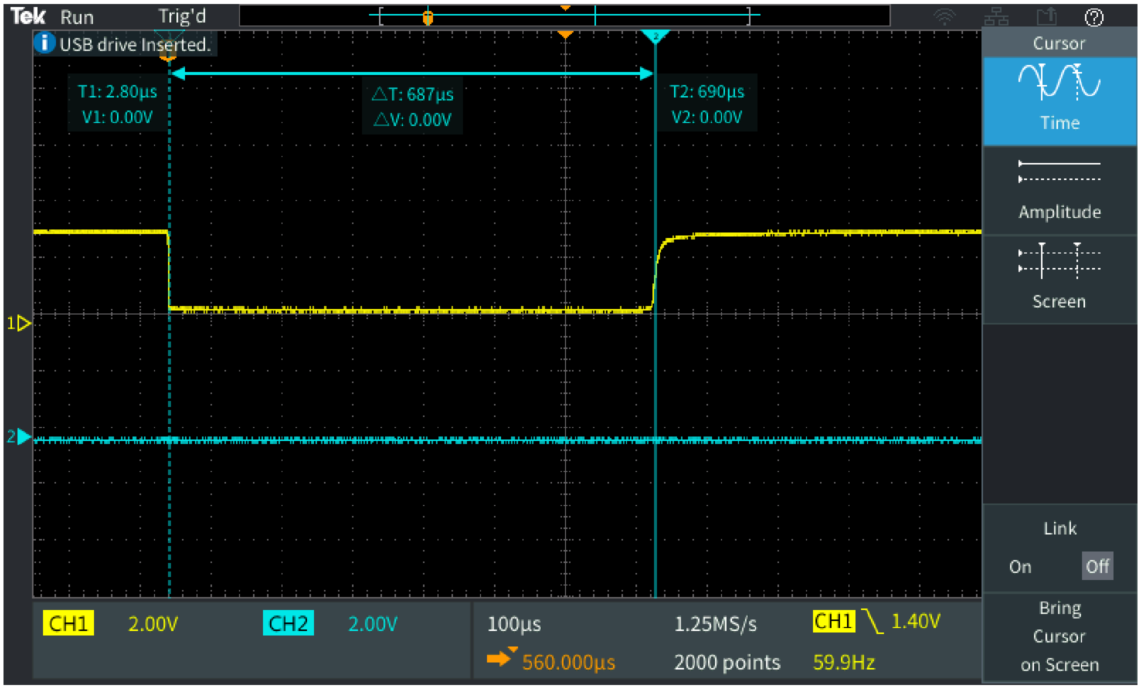


Figure 6.1 Camera trigger signal low (Yellow: Primary Camera, Blue: Secondary Camera). Cameras begin exposure at the rising edge after the signal trough.

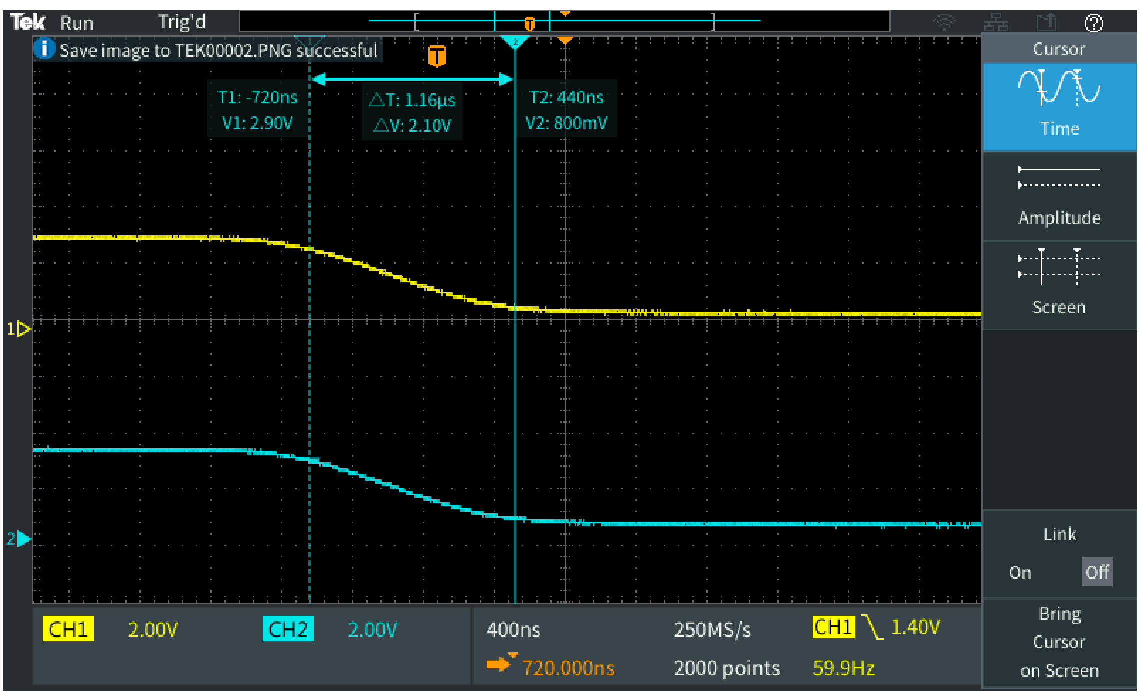


Figure 6.2 Camera trigger signal at falling edge (Yellow: Primary Camera, Blue: Secondary Camera). The trigger signal falls when the image acquisition function is called in the software.

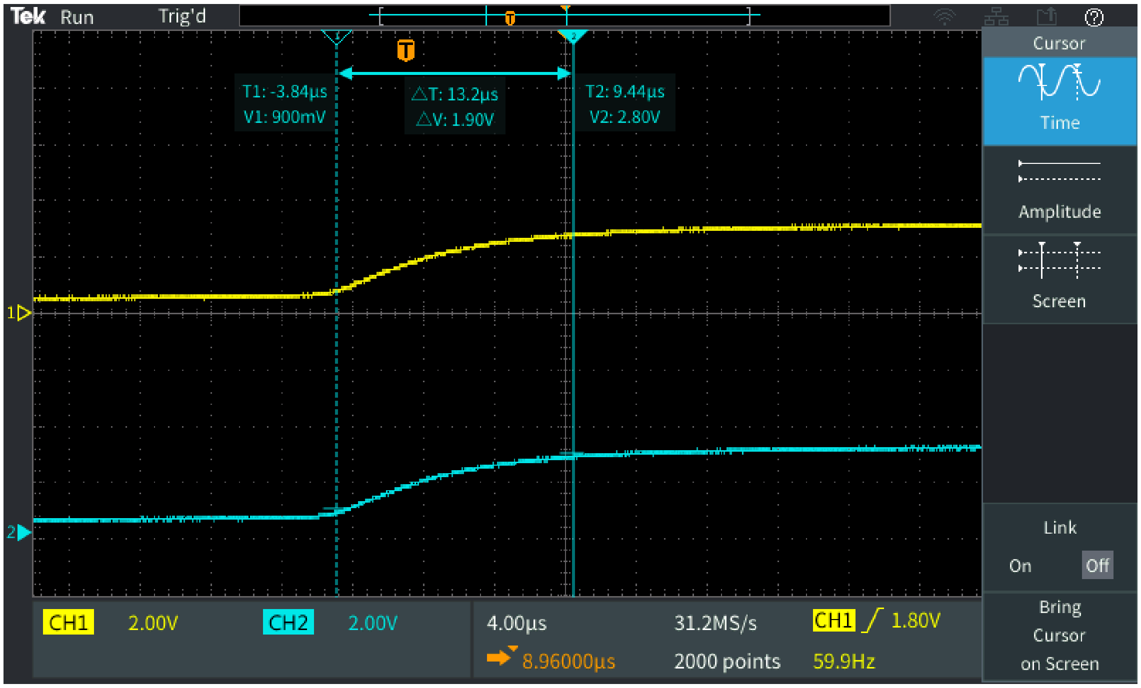


Figure 6.3 Camera trigger signal at rising edge (Yellow: Primary Camera, Blue: Secondary Camera). Once the rising edge meets the voltage required camera exposure begins for the predetermined exposure time selected by the user.

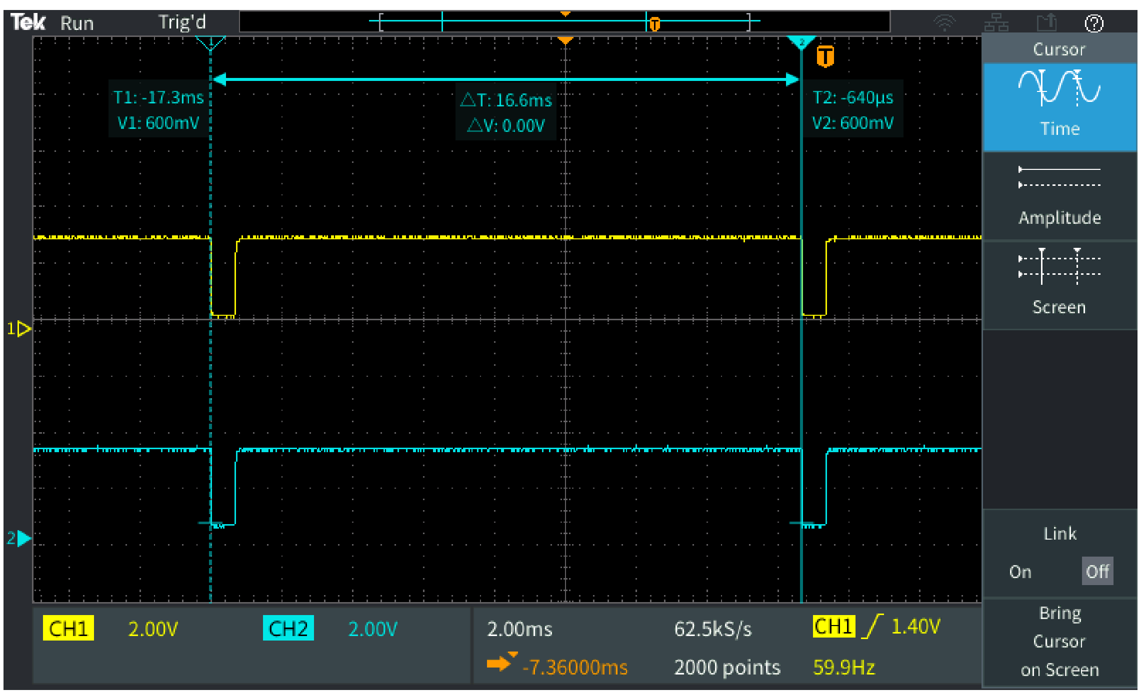


Figure 6.4 Primary Camera trigger signal (Yellow) compared to the input at Secondary Camera 2 (Blue). The period of the trigger signal is 16.667, matching the desired image capture rate.

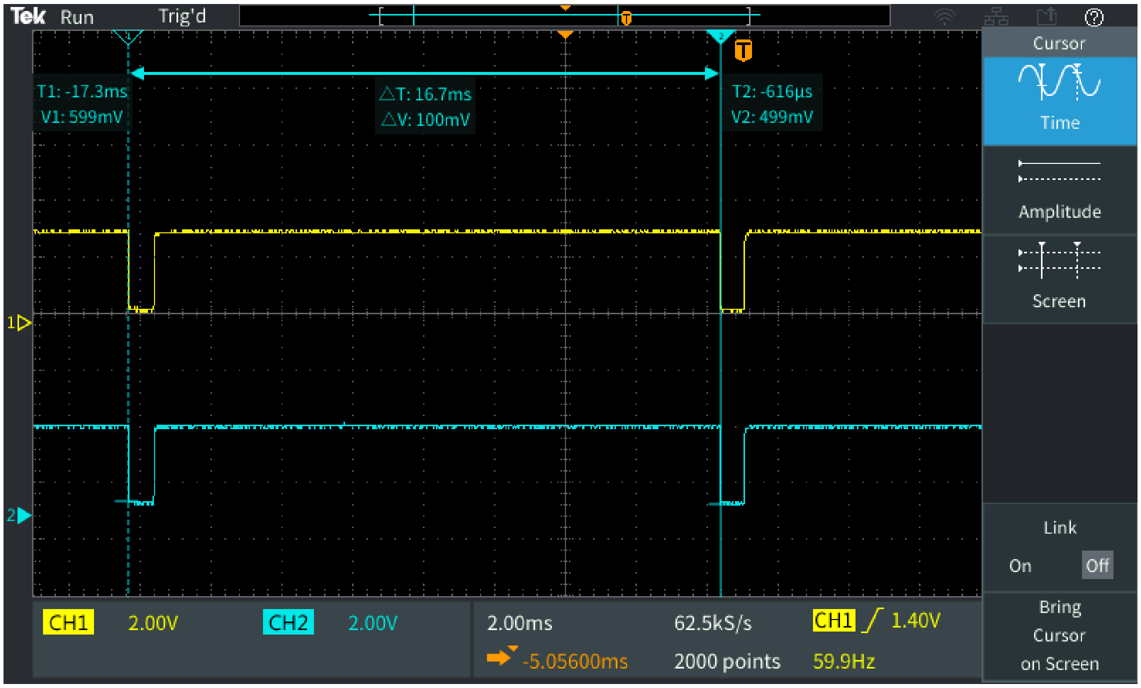


Figure 6.5 Primary Camera trigger signal (Yellow) compared to the input at Secondary Camera 3 (Blue). The period of the trigger signal is 16.667, matching the desired image capture rate.

Appendix C: Outcome Measures for other Participant(s)

Table 6-1 Body segment lengths for the walking straight, walking turn, and walker curve conditions (mean and standard deviation in brackets). Smart Hallway (SH) are calculated using the 3D reconstructed data and Delta is the difference between the Smart Hallway and ground truth segment length.

	Walking Straight		Walking Turn		Walking Curve	
Limb Segment (cm)	**SH**	**Delta**	**SH**	**Delta**	**SH**	**Delta**
Left Arm	29.27 (1.69)	1.77 (1.69)	29.19 (1.36)	1.69 (1.36)	29.10 (1.40)	1.60 (1.40)
Left Forearm	27.56 (3.22)	0.56 (3.22)	25.97 (2.88)	-1.03 (2.88)	26.29 (1.82)	-0.71 (1.82)
Right Arm	30.19 (3.80)	2.69 (3.80)	28.80 (1.62)	1.30 (1.62)	29.17 (1.20)	1.67 (1.20)
Right Forearm	26.68 (2.90)	-0.32 (2.90)	25.94 (2.84)	-1.06 (2.84)	25.54 (1.73)	-1.46 (1.73)
Left Thigh	41.54 (2.44)	-2.46 (2.44)	41.65 (2.85)	-2.35 (2.85)	40.94 (1.94)	-3.06 (1.94)
Left Shank	42.49 (3.81)	0.99 (3.81)	42.66 (3.34)	1.16 (3.34)	42.98 (2.92)	1.48 (2.92)
Right Thigh	41.87 (3.09)	-2.63 (3.09)	41.43 (2.50)	-3.07 (2.50)	41.12 (2.00)	-3.38 (2.00)
Right Shank	41.75 (4.09)	0.25 (4.09)	42.51 (3.70)	1.01 (3.70)	42.13 (3.02)	0.63 (3.02)
Left Ankle to Heel	7.67 (2.54)	-0.33 (2.54)	7.02 (2.26)	-0.98 (2.26)	7.00 (1.88)	-1.00 (1.88)
Left Ankle to Big Toe	16.06 (4.26)	-3.44 (4.26)	16.63 (4.60)	-2.87 (4.60)	17.10 (3.13)	-2.40 (3.13)
Left Ankle to Small Toe	14.38 (3.90)	-2.12 (3.90)	15.96 (4.06)	-0.54 (4.06)	16.00 (3.47)	-0.50 (3.47)
Left Toe Width	7.05 (2.46)	-1.95 (2.46)	8.00 (3.27)	-1.00 (3.27)	7.05 (2.58)	-1.95 (2.58)
Right Ankle to Heel	7.20 (3.33)	-0.80 (3.33)	7.36 (2.21)	-0.64 (2.21)	10.50 (4.47)	2.50 (4.47)
Right Ankle to Big Toe	17.09 (3.67)	-2.91 (3.67)	16.98 (4.04)	-3.02 (4.04)	17.24 (2.82)	-2.76 (2.82)
Right Ankle to Small Toe	15.68 (3.80)	-0.82 (3.80)	14.56 (3.16)	-1.94 (3.16)	15.20 (2.66)	-1.30 (2.66)
Right Toe Width	7.29 (2.58)	-1.21 (2.58)	8.90 (3.41)	0.40 (3.41)	7.76 (3.89)	-0.74 (3.89)
Shoulder Width	33.18 (1.46)	-0.82 (1.46)	32.50 (1.24)	-1.50 (1.24)	32.67 (1.61)	-1.33 (1.61)
Hip Width	20.71 (1.28)	-0.29 (1.28)	20.83 (1.46)	-0.17 (1.46)	20.93 (1.89)	-0.07 (1.89)
Chest Height	52.93 (1.34)	-3.07 (1.34)	52.32 (1.72)	-3.68 (1.72)	53.26 (1.39)	-2.74 (1.39)

Table 6-2 Body segment lengths for the walking straight, cane, and walker test conditions (mean and standard deviation in brackets). Smart Hallway (SH) are calculated using the 3D reconstructed data and Delta is the difference between the Smart Hallway and ground truth segment length.

	Walking Straight		Cane		Walker	
Limb Segment (cm)	**SH**	**Delta**	**SH**	**Delta**	**SH**	**Delta**
Left Arm	29.27 (1.69)	1.77 (1.69)	29.42 (1.72)	1.92 (1.72)	29.09 (1.60)	1.59 (1.60)
Left Forearm	27.56 (3.22)	0.56 (3.22)	26.31 (1.57)	-0.69 (1.57)	28.88 (3.90)	1.88 (3.90)
Right Arm	30.19 (3.80)	2.69 (3.80)	29.60 (1.90)	2.10 (1.90)	29.42 (1.98)	1.92 (1.98)
Right Forearm	26.68 (2.90)	-0.32 (2.90)	27.42 (3.64)	0.42 (3.64)	28.02 (3.55)	1.02 (3.55)
Left Thigh	41.54 (2.44)	-2.46 (2.44)	40.75 (2.48)	-3.25 (2.48)	41.44 (3.09)	-2.56 (3.09)
Left Shank	42.49 (3.81)	0.99 (3.81)	42.38 (2.97)	0.88 (2.97)	43.56 (3.39)	2.06 (3.39)
Right Thigh	41.87 (3.09)	-2.63 (3.09)	41.26 (3.15)	-3.24 (3.15)	40.89 (3.03)	-3.61 (3.03)
Right Shank	41.75 (4.09)	0.25 (4.09)	41.85 (3.52)	0.35 (3.52)	42.93 (3.53)	1.43 (3.53)
Left Ankle to Heel	7.67 (2.54)	-0.33 (2.54)	7.08 (2.21)	-0.92 (2.21)	6.80 (2.21)	-1.20 (2.21)
Left Ankle to Big Toe	16.06 (4.26)	-3.44 (4.26)	17.59 (4.52)	-1.91 (4.52)	15.67 (3.68)	-3.83 (3.68)
Left Ankle to Small Toe	14.38 (3.90)	-2.12 (3.90)	15.27 (4.39)	-1.23 (4.39)	13.31 (4.15)	-3.19 (4.15)
Left Toe Width	7.05 (2.46)	-1.95 (2.46)	6.96 (2.62)	-2.04 (2.62)	7.53 (2.64)	-1.47 (2.64)
Right Ankle to Heel	7.20 (3.33)	-0.80 (3.33)	6.13 (2.08)	-1.87 (2.08)	7.09 (2.44)	-0.91 (2.44)
Right Ankle to Big Toe	17.09 (3.67)	-2.91 (3.67)	16.84 (4.34)	-3.16 (4.34)	17.09 (3.86)	-2.91 (3.86)
Right Ankle to Small Toe	15.68 (3.80)	-0.82 (3.80)	14.57 (3.60)	-1.93 (3.60)	14.75 (3.59)	-1.75 (3.59)
Right Toe Width	7.29 (2.58)	-1.21 (2.58)	7.70 (2.87)	-0.80 (2.87)	7.73 (2.75)	-0.77 (2.75)
Shoulder Width	33.18 (1.46)	-0.82 (1.46)	33.62 (2.07)	-0.38 (2.07)	33.57 (1.84)	-0.43 (1.84)
Hip Width	20.71 (1.28)	-0.29 (1.28)	20.58 (1.51)	-0.42 (1.51)	20.63 (1.21)	-0.37 (1.21)
Chest Height	52.93 (1.34)	-3.07 (1.34)	52.56 (1.15)	-3.44 (1.15)	54.13 (2.43)	-1.87 (2.43)

Table 6-3 Stride parameters for the walking straight test (mean and standard deviation in brackets).

	Left			Right		
Parameters (units)	**SH**	**Ground Truth**	**Delta**	**SH**	**Ground Truth**	**Delta**
Stride length (m)	1.48 (0.15)	1.50 (0.13)	0.01 (0.07)	1.48 (0.17)	1.47 (0.15)	-0.01 (0.08)
Stride time (s)	1.13 (0.08)	1.14 (0.07)	0.01 (0.03)	1.18 (0.21)	1.17 (0.19)	-0.01 (0.05)
Stride speed (m/s)	1.31 (0.16)	1.32 (0.15)	0.00 (0.04)	1.26 (0.17)	1.26 (0.17)	0.00 (0.03)
Step length (m)	0.45 (0.15)	0.32 (0.12)	-0.14 (0.05)	0.45 (0.17)	0.35 (0.13)	-0.10 (0.06)
Step width (m)	0.07 (0.04)	0.08 (0.04)	0.01 (0.00)	0.08 (0.04)	0.09 (0.03)	0.01 (0.02)
Step time (s)	0.59 (0.11)	0.58 (0.08)	-0.01 (0.05)	0.57 (0.05)	0.57 (0.05)	0.01 (0.04)
Cadence (steps/min)	101.94 (8.36)	103.40 (5.93)	1.46 (4.14)	103.68 (13.56)	103.83 (13.47)	0.14 (4.55)
Stance time (s)	0.69 (0.05)	0.74 (0.07)	0.05 (0.06)	0.70 (0.07)	0.72 (0.04)	0.02 (0.06)
Swing time (s)	0.45 (0.06)	0.42 (0.02)	-0.03 (0.06)	0.48 (0.08)	0.42 (0.03)	-0.06 (0.07)
Stance swing ratio (NA)	1.53 (0.19)	1.75 (0.11)	0.22 (0.25)	1.47 (0.28)	1.71 (0.10)	0.24 (0.29)
Double support time (s)	0.12 (0.03)	0.16 (0.04)	0.04 (0.05)	0.13 (0.05)	0.16 (0.04)	0.03 (0.06)
Foot angle (°)	12.63 (6.90)	12.45 (6.97)	-0.18 (0.07)	21.15 (9.33)	20.33 (8.49)	-0.81 (2.02)

Table 6-4 Stride parameters for the walking turn test (mean and standard deviation in brackets).

	Left			Right		
Parameters (units)	**Measured**	**Ground Truth**	**Delta**	**Measured**	**Ground Truth**	**Delta**
Stride length (m)	1.56 (0.20)	1.54 (0.19)	-0.02 (0.08)	1.55 (0.21)	1.53 (0.24)	-0.02 (0.10)
Stride time (s)	1.05 (0.10)	1.03 (0.10)	-0.02 (0.08)	1.04 (0.07)	1.04 (0.08)	0.00 (0.05)
Stride speed (m/s)	1.45 (0.24)	1.47 (0.20)	0.02 (0.05)	1.51 (0.22)	1.49 (0.22)	-0.02 (0.05)
Step length (m)	0.40 (0.13)	0.35 (0.12)	-0.05 (0.04)	0.41 (0.13)	0.37 (0.12)	-0.04 (0.04)
Step width (m)	0.08 (0.05)	0.09 (0.05)	0.01 (0.03)	0.11 (0.05)	0.12 (0.05)	0.01 (0.02)
Step time (s)	0.54 (0.13)	0.53 (0.13)	0.00 (0.08)	0.52 (0.09)	0.51 (0.07)	-0.01 (0.07)
Cadence (steps/min)	109.03 (7.18)	109.67 (7.33)	0.64 (4.73)	112.03 (2.41)	112.92 (3.49)	0.89 (4.79)
Stance time (s)	0.62 (0.10)	0.62 (0.09)	0.00 (0.07)	0.62 (0.10)	0.62 (0.09)	0.00 (0.07)
Swing time (s)	0.42 (0.08)	0.40 (0.07)	-0.02 (0.10)	0.42 (0.08)	0.40 (0.07)	-0.02 (0.10)
Stance swing ratio (NA)	1.54 (0.53)	1.56 (0.32)	0.02 (0.57)	1.54 (0.53)	1.56 (0.32)	0.02 (0.57)
Double support time (s)	0.12 (0.13)	0.14 (0.11)	0.01 (0.05)	0.13 (0.13)	0.14 (0.11)	0.01 (0.05)
Foot angle (°)	14.21 (6.84)	13.95 (6.87)	-0.26 (1.65)	14.21 (6.84)	13.95 (6.87)	-0.26 (1.65)

Table 6-5 Stride parameters for the walking curve test (mean and standard deviation in brackets).

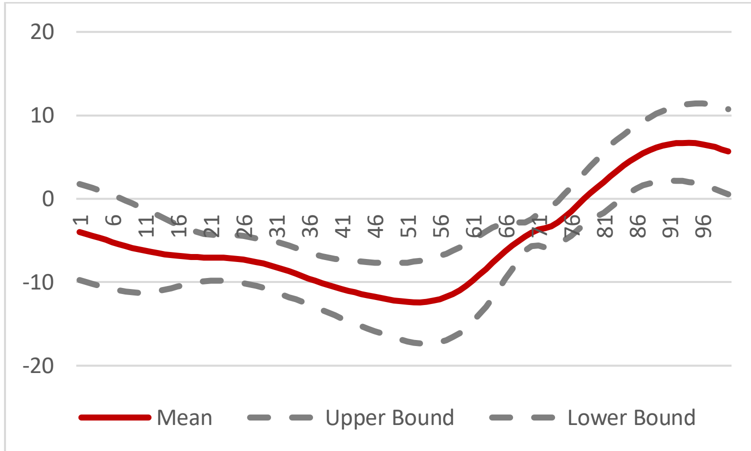
	Left			Right		
Parameters (units)	**Measured**	**Ground Truth**	**Delta**	**Measured**	**Ground Truth**	**Delta**
Stride length (m)	1.36 (0.23)	1.37 (0.16)	0.02 (0.12)	1.27 (0.20)	1.26 (0.19)	-0.01 (0.12)
Stride time (s)	1.03 (0.10)	1.04 (0.06)	0.01 (0.06)	1.06 (0.07)	1.05 (0.05)	-0.01 (0.05)
Stride speed (m/s)	1.31 (0.15)	1.32 (0.13)	0.01 (0.06)	1.22 (0.12)	1.22 (0.11)	0.00 (0.04)
Step length (m)	0.37 (0.12)	0.27 (0.10)	-0.09 (0.05)	0.35 (0.18)	0.28 (0.14)	-0.07 (0.05)
Step width (m)	0.27 (0.15)	0.22 (0.14)	-0.05 (0.07)	0.31 (0.18)	0.27 (0.15)	-0.04 (0.04)
Step time (s)	0.51 (0.10)	0.50 (0.07)	-0.01 (0.06)	0.53 (0.05)	0.54 (0.05)	0.01 (0.05)
Cadence (steps/min)	115.62 (9.07)	118.48 (7.58)	2.85 (7.42)	110.18 (8.01)	110.78 (5.57)	0.59 (5.09)
Stance time (s)	0.62 (0.08)	0.65 (0.04)	0.04 (0.06)	0.64 (0.06)	0.65 (0.05)	0.01 (0.04)
Swing time (s)	0.41 (0.08)	0.39 (0.05)	-0.03 (0.05)	0.43 (0.09)	0.41 (0.05)	-0.02 (0.06)
Stance swing ratio (NA)	1.48 (0.31)	1.64 (0.12)	0.16 (0.28)	1.46 (0.37)	1.58 (0.24)	0.12 (0.22)
Double support time (s)	0.12 (0.05)	0.13 (0.03)	0.01 (0.05)	0.11 (0.04)	0.12 (0.02)	0.01 (0.04)
Foot angle (°)	35.63 (17.17)	37.41 (17.72)	1.77 (3.14)	34.81 (18.50)	36.20 (18.90)	1.38 (3.08)

Table 6-6 Stride Parameters for the cane test (mean and standard deviation in brackets).

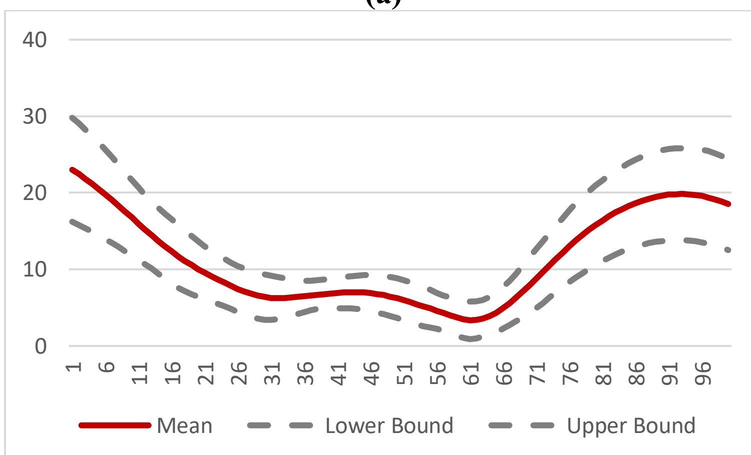
	Left			Right		
Parameters (units)	**Measured**	**Ground Truth**	**Delta**	**Measured**	**Ground Truth**	**Delta**
Stride length (m)	1.65 (0.19)	1.64 (0.20)	-0.01 (0.05)	1.68 (0.20)	1.70 (0.18)	0.02 (0.10)
Stride time (s)	1.58 (0.10)	1.56 (0.12)	-0.02 (0.06)	1.58 (0.08)	1.58 (0.09)	0.00 (0.05)
Stride speed (m/s)	1.05 (0.11)	1.05 (0.11)	0.00 (0.01)	1.09 (0.11)	1.09 (0.11)	-0.01 (0.02)
Step length (m)	0.47 (0.20)	0.46 (0.20)	-0.01 (0.03)	0.54 (0.16)	0.49 (0.15)	-0.06 (0.04)
Step width (m)	0.06 (0.03)	0.06 (0.03)	0.00 (0.01)	0.08 (0.03)	0.08 (0.03)	0.00 (0.01)
Step time (s)	0.75 (0.07)	0.79 (0.07)	0.04 (0.05)	0.82 (0.09)	0.77 (0.09)	-0.05 (0.07)
Cadence (steps/min)	80.48 (7.55)	77.32 (5.83)	-3.16 (4.63)	73.37 (5.79)	78.08 (6.17)	4.70 (5.13)
Stance time (s)	0.95 (0.07)	0.92 (0.06)	-0.03 (0.08)	0.93 (0.09)	0.96 (0.07)	0.03 (0.07)
Swing time (s)	0.63 (0.09)	0.63 (0.09)	0.00 (0.08)	0.65 (0.09)	0.59 (0.08)	-0.05 (0.07)
Stance swing ratio (NA)	1.51 (0.21)	1.44 (0.16)	-0.07 (0.19)	1.45 (0.21)	1.64 (0.21)	0.18 (0.28)
Double support time (s)	0.18 (0.07)	0.18 (0.03)	-0.01 (0.07)	0.13 (0.05)	0.14 (0.03)	0.01 (0.05)
Foot angle (°)	13.50 (7.01)	13.40 (7.01)	-0.10 (0.57)	15.22 (3.39)	14.92 (3.80)	-0.30 (1.13)

Table 6-7 Stride Parameters for the walker test (mean and standard deviation in brackets).

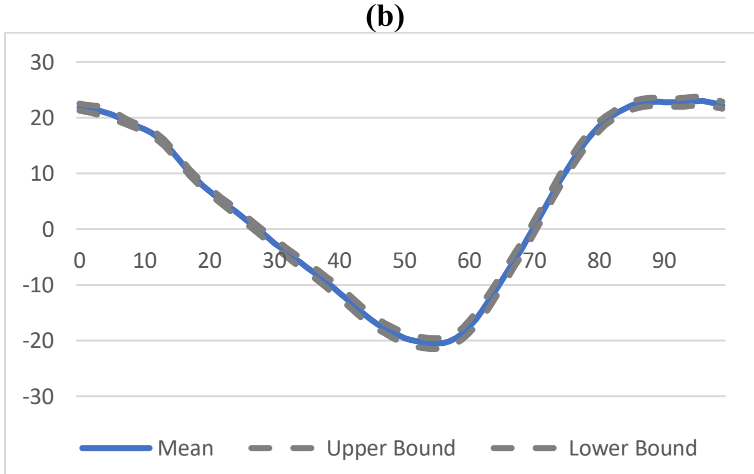
	Left			Right		
Parameters (units)	**Measured**	**Ground Truth**	**Delta**	**Measured**	**Ground Truth**	**Delta**
Stride length (m)	1.01 (0.22)	1.01 (0.23)	0.00 (0.06)	1.02 (0.20)	1.01 (0.19)	-0.01 (0.06)
Stride time (s)	1.78 (0.13)	1.79 (0.14)	0.00 (0.10)	1.81 (0.18)	1.80 (0.14)	-0.01 (0.08)
Stride speed (m/s)	0.57 (0.13)	0.57 (0.13)	0.00 (0.01)	0.57 (0.13)	0.57 (0.13)	0.00 (0.01)
Step length (m)	0.33 (0.15)	0.30 (0.16)	-0.03 (0.04)	0.30 (0.15)	0.29 (0.16)	-0.02 (0.03)
Step width (m)	0.06 (0.04)	0.07 (0.04)	0.01 (0.01)	0.11 (0.04)	0.11 (0.04)	0.00 (0.01)
Step time (s)	0.90 (0.18)	0.88 (0.16)	-0.02 (0.08)	0.91 (0.13)	0.92 (0.14)	0.01 (0.07)
Cadence (steps/min)	66.09 (8.17)	68.15 (7.68)	2.06 (2.42)	66.81 (7.72)	66.13 (7.58)	-0.68 (4.17)
Stance time (s)	1.13 (0.10)	1.20 (0.12)	0.06 (0.12)	1.16 (0.14)	1.20 (0.11)	0.04 (0.09)
Swing time (s)	0.67 (0.13)	0.59 (0.09)	-0.08 (0.12)	0.63 (0.09)	0.59 (0.07)	-0.04 (0.09)
Stance swing ratio (NA)	1.70 (0.28)	1.94 (0.24)	0.24 (0.39)	1.84 (0.31)	1.99 (0.23)	0.15 (-0.08)
Double support time (s)	0.23 (0.08)	0.29 (0.07)	0.06 (0.10)	0.25 (0.10)	0.31 (0.09)	0.06 (0.13)
Foot angle (°)	14.55 (5.73)	14.60 (5.84)	0.06 (0.78)	15.25 (4.82)	15.25 (4.76)	0.01 (2.88)



(a)

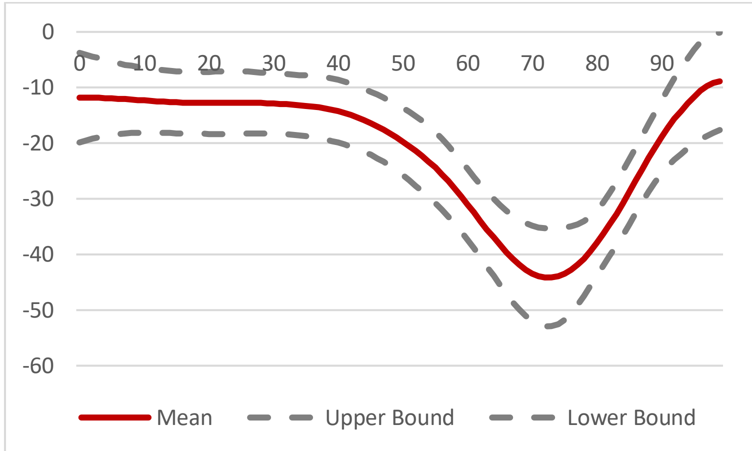


(b)

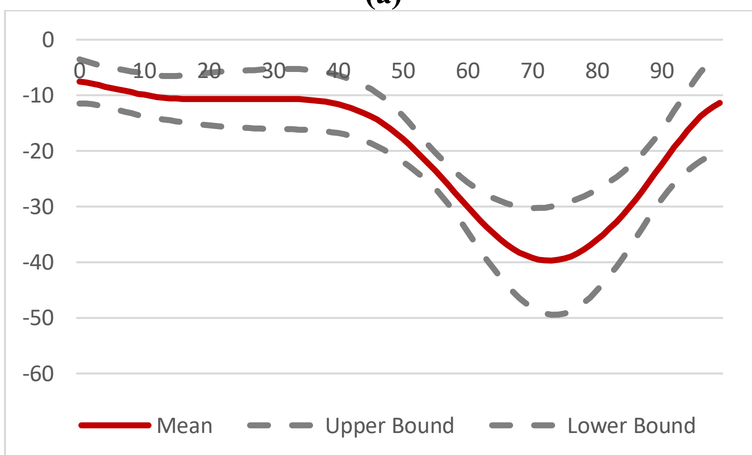


(c)

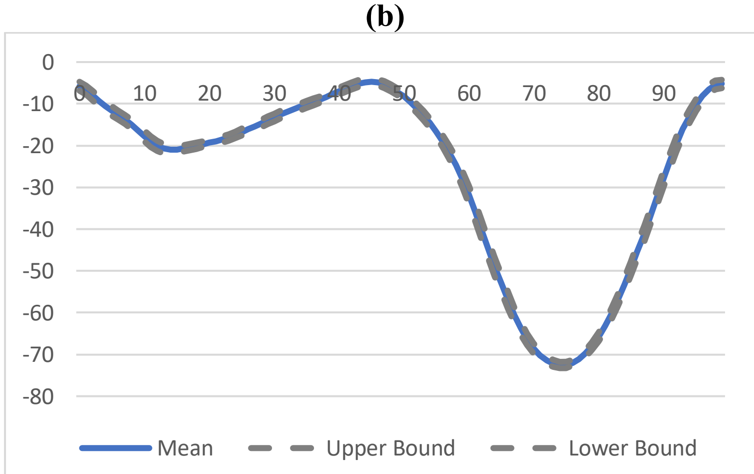
Figure 6.6 Ensemble averaged hip angles: a) Walking towards the camera array, b) Walking away from the camera array, c) Vicon normal walking comparator. The horizontal axis is the ensemble averaged time steps, and the vertical axis is the angle in degrees.



(a)

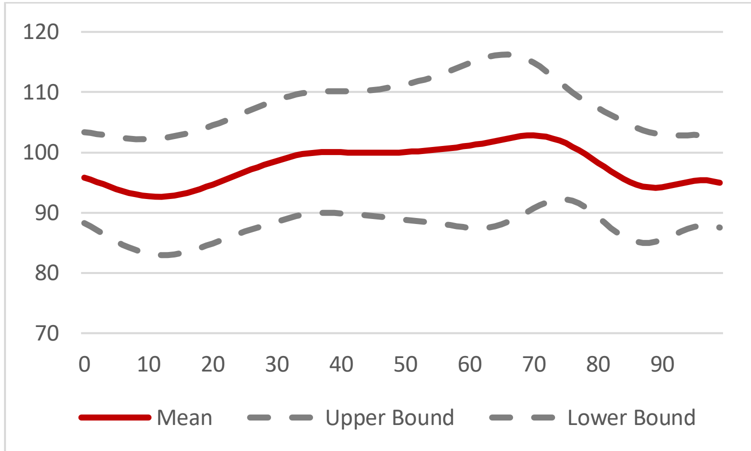


(b)

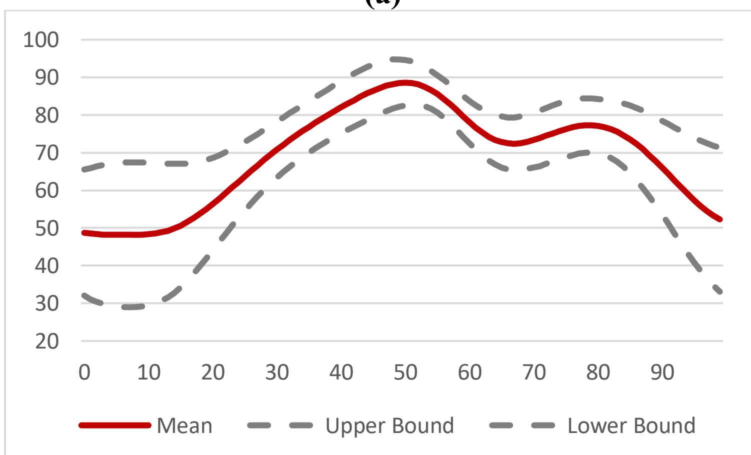


(c)

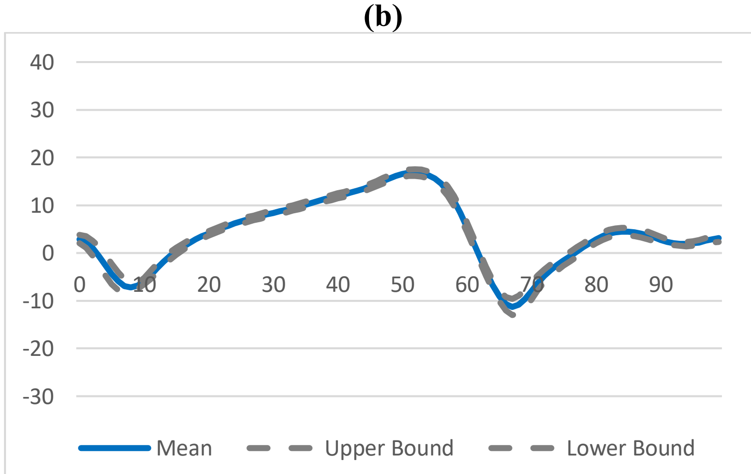
Figure 6.7 Ensemble averaged knee angles: a) Walking towards the camera array, b) Walking away from the camera array, c) Vicon normal walking comparator. The horizontal axis is the ensemble averaged time steps, and the vertical axis is the angle in degrees.



(a)



(b)



(c)

Figure 6.8 Ensemble averaged ankle angles: a) Walking towards the camera array, b) Walking away from the camera array, c) Vicon normal walking comparator. The horizontal axis is the ensemble averaged time steps, and the vertical axis is the angle in degrees.