ResCoCo: Residual ConvLSTM Network with Contrastive Learning for 3D Joint Angle Estimation

RESCOCO: RESIDUAL CONVLSTM NETWORK WITH CONTRASTIVE LEARNING FOR 3D JOINT ANGLE ESTIMATION

BY

XIJIAN LOU, M.Sc., (Computer Science)
McMaster University, Hamilton, Canada

A THESIS
SUBMITTED TO THE DEPARTMENT OF COMPUTING AND SOFTWARE
AND THE SCHOOL OF GRADUATE STUDIES
OF MCMASTER UNIVERSITY
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

Master of Science (2022)	McMaster University
(Computing and Software)	Hamilton, Ontario, Canada

TITLE:	ResCoCo: Residual ConvLSTM Network with Contrastive Learning for 3D Joint Angle Estimation
AUTHOR:	Xijian Lou M.Sc., (Computer Science) McMaster University, Hamilton, Canada
SUPERVISOR:	Dr. Rong Zheng
NUMBER OF PAGES:	xiv, 57

Abstract

3D joint angle is an important indicator in human gait analysis, and musculoskeletal disease diagnosis and treatment. To accurately estimate 3D joint angle, a Deep Learning approach, Residual ConvLSTM network with Contrastive Learning (ResCoCo), is proposed in this study. A sequence shortening layer is introduced in ResCoCo to discard part of the output sequence estimated by bi-directional LSTM layers from incomplete context, and a residual block is used for network depth increase. Contrastive learning is employed in ResCoCo to ensure robust and efficient representation extraction.

The model is validated on the WEVAL dataset for 3D knee joint angle estimation during walking. The experiment result shows that the sequence shortening layer and residual block benefit the 3D joint angle estimation accuracy, while contrastive learning increases the model resistance towards IMU-to-Segment (I2S) alignment and sensor placement variations. Furthermore, the sensor configuration for the model input is investigated. Using inertial data from two sensors as the model input is economical while effective, and leads to good model robustness towards I2S alignment and sensor placement variations, compared to using inertial data from a single sensor or six sensors as the model input. The model is also compared with a model-driven method. It is shown that ResCoCo not only provides accurate estimation accuracy

along three rotation axes, but it is also free of calibration procedures, physical constraints or predefined anatomical models, compared to model-driven approaches.

Acknowledgements

Writing the graduation dissertation is like writing down the story of my two-year research life in the Wireless System Research (WiSeR) group. When I look back on this journey, I am deeply grateful that Dr. Rong Zheng gave me the opportunity to dive into Human Pose Estimation, a prosperous research area with great application value in the medicine and the game industry. She has always been my role model as an exceptional, insightful, and candid researcher and mentor. She shows me how to be a strong female lead, an active athlete, and a kind person. The most important lesson I learned in my research life is do not rush into actions, think in depth and find the logic behind every phenomenon.

I would like to thank Yujiao Hao to be my closest comrade in the academic world. She has provided innumerable valuable suggestions for my research project. We share each other's achievements and burdens.

I want to thank all my dear friends who always provide mental support and have been great companions to me. I entrust part of my heart to the care of them, like Voldemort distributes his horcruxes. I cannot be this curious and enthusiastic without their love and care.

Most importantly, I want to thank my parents for their love and support. They are the source of my courage, strength, and resilience. I can never go through this

journey without them.

Abbreviations

I2S	IMU to Segment
RoM	Range of Motion
OMoCap	Optical Motion Capture
IMU	Inertial Measurement Unit
ISB	International Society of Biomechanics
JCS	Joint Coordinate System
LSTM	Long Short-Term Memory network
GRNN	Generalized Regression Neural Network
NARX	Nonlinear Autoregressive Network with Exogenous Inputs
DeepConvLSTM	Deep Convolutional Long Short-Term Memory Network
ANN	Artificial Neural Network
GRF	Ground Reaction Force
CoM	Center of Mass
FFNN	Fully-connected Feedforward Neural Network
DoF	Degree of Freedom
EMG	electromyography
RGB	red, green and blue

BAF	Bone-embedded Anatomical Frame
MAE	Mean Avergae Error
RMSE	Root Mean Square Error
SD	Standard Deviation
ReLU	Rectified Linear Unit
CV	Computer Vision
NLP	Natural Language Processing
HPE	Human Pose Estimation
HAR	Human Activity Recognition

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Knee joint angle estimation is an important indicator widely used in human gait analysis, athlete performance assessment, musculoskeletal disease diagnosis, treatment, progression monitoring, and rehabilitation evaluation [1, 2]. Detecting and intervening in the gait abnormality can also decrease the falling risk of elderly adults [3]. Specifically, joint angle-related indicators such as knee ranges of motion (ROM) are important for total knee arthroplasty and frequently used for athlete performance assessment [4]. Therefore, joint angle is an estimation objective of great value.

Optical Motion Capture (OMoCap) systems are the current gold standard for joint angle estimation, but they are limited to the lab environment, expensive, and complicated for daily use. Goniometers are the current norm in clinical practice to measure joint angle. Although they are inexpensive and portable, trained examiners are required under circumstances such as post-operative joint angle assessment [5]. To achieve long-term, user-friendly and inexpensive 3D joint angle estimation, inertial measurement units (IMUs) become increasingly popular. IMUs consist of accelerometer and gyroscope for linear acceleration and angular velocity signal recording, respectively. They are extensively embedded in equipment such as smart phones and smart watches, thus accessible in daily life and for long-term usage.

There are two commonly used categories of IMU-based joint angle estimation methods: model-driven and data-driven methods. Model-driven methods need predefined anatomical models, and their accuracy is influenced by calibration procedures or physical constraints. Meanwhile, data-driven methods only need inertial data and ground truth joint angles and yield an end-to-end solution for joint angle estimation. However, variations caused by subject diversity, sensor placement, and I2S alignment lead to huge inertial sensor data distribution differences. The data-driven approach performance tends to decrease with input data distribution variation [6, 7]. To face this problem and exploit the advantages of data-driven methods, ResCoCo is proposed in this study for IMU-based 3D knee joint rotation angle estimation.

1.1 Background

To describe 3D knee joint rotation, three anatomical planes of the human body need to be defined, namely, the sagittal, frontal and transverse planes. The three anatomical planes are mutually perpendicular and divide a human body into left/right, anterior/posterior, and superior/posterior parts. As recommended by the International Society of Biomechanics (ISB), the Joint Coordinate System (JCS) can be used to define the knee joint angle with three rotation axes [8], as shown in Fig. 1.1. Knee flexion and extension are rotations in the sagittal plane around the x-axis; knee abduction and adduction are rotations in the frontal plane around the y-axis; knee external and internal rotations are in the transverse plane around the z-axis. Among the three, the knee joint has the largest range of motion (RoM) in flexion/extension movements.

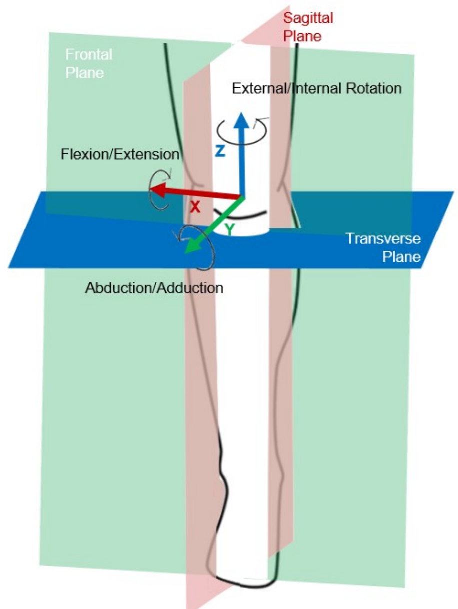


Figure 1.1: The human body anatomical planes and rotation axes to describe 3D knee joint motions.

Knee joint angle can be defined as the differential orientation between the adjacent bone-embedded anatomical frame (BAF), and can be derived from the orientation of sensors on the proximal and distal segments relative to the global coordinate frame and I2S alignment of these two sensors [8]. Specifically, to define the 3D knee joint angle, the BAFs of the femur and tibia are represented as IJK and ijk respectively, and the local coordinates of the thigh and shank sensors are UVW and uvw respectively, as shown in Fig. 1.2. The orientation of femur or tibia BAF w.r.t. the global coordinate frame ($X_gY_gZ_g$) can be described as:

$$R_i^{GS} = R_i^{GI} \otimes R_i^{IS}$$

i refers to the thigh or shank segment and R_i^{IS} represents its I2S alignment in quaternion representation. The I2S alignment is assumed to be constant. R_i^{GI} is the orientation of the sensor on segment i w.r.t. the global coordinate in quaternion representation. $\otimes$ corresponds to the quaternion multiplication operation. Knee joint angle is the differential orientation between coordinates IJK and ijk. This joint angle estimation procedure can be applied to other joints with a proximal and a distal segments.

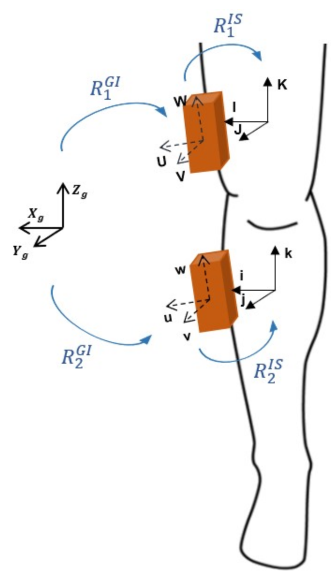


Figure 1.2: The coordinates frames used to define 3D knee joint angle, including the local sensor frames, BAFs and global frame.

1.2 Problem Statement

Is it possible to develop a data-driven model to accurately estimate the 3D knee joint angle from IMU signals? What is the optimal input sensor configuration, taking into consideration of the number of sensors and their placement? How does this data-driven approach perform on 3D knee joint angle estimation, compared to the model-driven method? How is the model generalization ability towards I2S alignment and sensor placement variations, and how to improve its generalization ability?

1.3 Contribution

An IMU-based data-driven model, namely ResCoCo, is proposed for 3D joint angle estimation in this study. The model consists of a backbone block and a residual block, and both blocks are ConvLSTM networks. It also has a sequence length shortening layer after the bidirectional LSTM layer in the backbone block. ResCoCo utilizes contrastive learning with a self-supervised training stage and a supervised training stage. The main contributions of this study include:

- A Deep Learning model, ResCoCo, is proposed with a residual block to increase network depth and a sequence length shortening layer to discard output sequence from bidirectional LSTM layers with incomplete context.

- ResCoCo can accurately estimate 3D knee joint angle from inertial data during walking without calibration procedure, physical constraints or predefined anatomical model, compared with a data-driven method. ResCoCo has the potential to be further applied to joints with complex motions, such as elbow and ankle joints.

- ResCoCo uses a contrastive learning framework, namely SimCLR, to extract robust and effective input data representations and resist sensor placement and I2S alignment variations.

- ResCoCo utilizes inertial data from a pair of IMUs, which is economical and efficient, compared to using single or six IMUs as the model input.

1.4 Organization

The remaining dissertation is organized as follows. It discusses existing works on I2S alignment, IMU-based joint angle estimation methods, and public lower-limb kinematics datasets in Chapter 2. The methodology of the proposed data-driven model ResCoCo is introduced in Chapter 3. The evaluation of the model-driven baseline method and ResCoCo on 3D knee joint angle estimation is presented in Chapter 4. Finally, the conclusion and future work are presented in Chapter 5.

Chapter 2

Related Work

The representative works in 3D knee joint angle estimation are reviewed in this chapter. Firstly, the existing I2S alignment assessment approaches are discussed in Section 2.1 for its great influence on various joint angle estimation approaches. Secondly, the existing 3D joint angle estimation approaches are reviewed in Section 2.2, including the model-driven methods and data-driven methods. Lastly, to evaluate the 3D joint angle estimation approaches, public lower-limb gait kinematics datasets are investigated and summarized in Section 2.3.

2.1 IMU to Segment Alignment

I2S alignment, also called I2S calibration, refers to the orientation of the sensor relative to the body segment that the sensor is mounted on. It is important to quantify joint motions as a component of the calibration-based model-driven joint angle estimation procedure, while its variation causes the inertial data distribution shift and degrades the data-driven method performance.

To acquire I2S alignment, static pose calibration and functional calibration approaches are commonly used. Static pose calibration commonly requires subjects to take a standing posture to determine the longitudinal axis of the segment coordinate [9, 10]. Functional calibration allows arbitrary mounting directions of IMUs, yet it requires the subject to take predefined calibration motions to determine different axes of the segment BAFs [11, 12]. For example, Favre *et al.* proposed hip abduction/adduction and two passive calibration movements for I2S alignment estimation [12]. Static calibration postures and predefined calibration movements can be used together for I2S alignment estimation with better accuracy. Favre *et al.* used static standing posture and hip abduction/adduction movement[13] to identify I2S alignment. However, typical I2S calibration procedures can be potentially problematic in their validity and reliability as investigated in [14]. To avoid calibration procedures for I2S estimation, Zimmermann *et al.* proposed a Deep Learning approach to automatically solve the I2S alignment regression task with both generated and real inertial sensor data [7].

2.2 3D Joint Angle Estimation

2.2.1 Model-driven Methods

Model-driven methods, also called kinematics-based methods, calculate joint angle as the differential orientation between BAFs of adjacent body segments based on IMU signals. They can be divided into assumed alignment methods, augmented data methods, calibration-based methods, and calibration-free methods, depending on the calibration procedure and data source [15]. Assumed alignment methods assume the local sensor frame is aligned with the anatomical frame of the body segment that the sensor is attached to, and use the sensor orientation as the orientation of BAF w.r.t. the global coordinate frame to derive joint angles [16, 17, 18, 19]. Augmented data methods use other data sources such as the OMoCap system to estimate the I2S alignment [20, 21, 22, 23].

Calibration-based methods are more accurate compared to assumed alignment methods and require fewer data sources compared to the augmented data methods. They exploit calibration procedures to estimate one or more anatomical axes of BAFs in the local sensor frame, based on IMU recordings. As introduced in [15], they have three major steps: the first step is to record the orientation of each sensor attached to body segment w.r.t. the global coordinate; then to estimate the orientation of the IMU w.r.t. the BAF of its respective body segment; lastly, to estimate the orientation of BAFs of the proximal segment relative to that of the distal segment as the joint angle. Estimation accuracy improvements are mainly made in these three major steps. Takeda *et al.* proposed to use angular velocity for transitional acceleration and gravitational acceleration estimation, and give the orientation of lower limb segments based on the acceleration measurement[9]. Favre *et al.* proposed different calibration postures and motions for better I2S alignment estimation accuracy [12, 13].

The calibration procedure relies on accurate operations, users' mobility, and the presence of experienced operators. Besides, the calibration accuracy strongly influences the joint angle estimation. In contrast, calibration-free methods avoid accurate mounting or alignment procedures, thus their popularity is growing fast [24, 25, 26]. Calibration-free methods exploit physical constraints on joint motion. For example in [27], motion parameters of the joint are derived from inertial data from IMUs on the adjacent segments. Based on the rigid body kinematics, the joint motion parameters derived from different IMUs should be congruent all the time, and the congruent constraint is used to build optimization objectives. The derived parameters including joint rotation axes and the IMU positions are used for the joint angle estimation.

2.2.2 Data-driven Methods

Model-driven methods for 3D joint angle estimation are prone to errors because of their strong dependency on the accurate predefined anatomical model, calibration procedures, anatomical parameter measurements, or physical constraints. The calibration postures and movements are also not ideal for users with mobility difficulty, and the calibration accuracy can have inter- and intra-examiner differences. Moreover, model-driven methods also cannot convey real-time estimation results.

The wide application of Deep Learning methods in Human Activity Recognition (HAR) and time series prediction makes them potential alternatives for 3D joint angle estimation [28, 29]. They do not require the definition of underlying anatomical models nor calibration procedures while yielding instant estimation results. Findlow *et al.* first applied a Generalized Regression Neural Network (GRNN) for 1D lower limb joint kinematics estimation based on inertial data recorded from IMUs on the foot and shank [6]. Luu *et al.* also used GRNN to estimate lower limb joint angle waveform for gait pattern planning in the robotic gait rehabilitation [30]. To predict dynamic lower limb data including joint kinetics and Ground Reaction Forces (GRFs), Lim *et al.* used Artificial Neural Network (ANN) with the Center of Mass (CoM) as the single IMU location [31]. Argent *et al.* explored Machine Learning algorithms including linear regression, polynomial regression, decision tree regression and random forest regression for hip and knee joint angle estimation with the input from a single IMU [5]. Dorschky *et al.* presented a method to create realistic inertial sensor data via biomechanical simulation and used Convolutional Neural Network (CNN) to estimate 1D joint angles, joint moments, and GRFs during running and walking [32]. Conte *et al.* investigated the estimation accuracy of Long Short-Term Memory (LSTM) network, GRNN and nonlinear autoregressive network with exogenous inputs (NARX) in modeling lower body joint angles in the sagittal plane with a single IMU on foot [33]. The deepest neural network used in lower limb joint angle estimation is a deep Convolutional Long Short-Term Memory network (DeepConvLSTM) proposed by Hernandez *et al.* The model receives data from 5 sensors and achieves the average of mean absolute error (MAE) of 3.6 (2.1)°across all lower limb joint angles [34].

All the aforementioned Deep Learning based joint angle estimation methods focus on knee flexion/extension angle estimation and neglect the importance of knee rotations on frontal and transverse planes. Knee abduction/adduction and internal/external rotation angles are efficient objective markers in knee osteoarthritis and running-related injury interventions [35, 36]. The reason for the research deficiency can be that knee joint angles on frontal and transverse planes are more difficult to acquire and estimate due to their smaller RoMs and they are more easily influenced by noises caused by soft tissue movement. Deep Learning based 3D knee joint angle estimation method was first proposed by Mundt *et al.* with a Fully-connected Feedforward Neural Network (FFNN) and a LSTM network. The models used simulated inertial sensor data derived from OMoCap data to estimate 3D lower limb joint angles and moments [37]. Rapp *et al.* proposed a hybrid Deep Learning and optimization approach for 3D lower limb joint angle estimation with simulated data from virtual

IMUs on shank and thigh [38]. However, the simulated data lacks real-world artifacts such as noise induced by soft tissue movement and needs further validation before being applied in the free-living scenarios. To face this problem, Mundt *et al.* collected real inertial data and simulated sensor position and orientation in their later work [37] and observed that the estimation accuracy was improved after dataset enlargement. However, their sensor orientation and position simulation targeted at position and orientation error compensation. The influence of sensor position and I2S alignment variation and model robustness towards these variation were not further investigated.

2.3 Public Datasets

To validate ResCoCo for 3D knee joint angle estimation, public datasets with lower limb gait kinematics recordings are searched and summarized in 2.1. Datasets with both inertial sensor data and 3D knee joint angle recordings are limited in number. The kinematic datasets containing both inertial sensor data and ground truth 3D knee joint angles are mostly not qualified for this study. Specifically, the inertial sensor data and 3D knee joint angle recordings do not overlap in the MoVi dataset. The inertial data in [39] is not raw inertial data. The data-driven model derived from it will require extra data pre-processing procedure and thus have lower applicability.

WEVAL dataset is chosen for evaluation in this study. It is a lower-limb joint gait kinematics dataset with both raw IMU signals and ground truth knee joint angles derived from the OMoCap system. It is collected from 15 healthy young adults (7 females) with an average age of 26 (4) years and a body mass index of 22.6 (3.0) kg/m^2. All participants provided written informed consent. The ethical approval was provided by the institutional Ethics Review Board. The gait trials were collected as barefoot, over-ground walking at each participant's self-selected speed for 2 sessions with 2-4 days in between. A standardized 22-marker Helen Hayes marker set and 12-camera optoelectronic motion capture system (Motion Analysis Corp; 120 Hz) were used to compute the criterion 3D knee joint kinematics using commercially available software (Visual3D). Additionally, measurements from six IMUs (Shimmer3; 100 Hz) on the thigh and the shank were collected simultaneously and synchronously. The inertial sensor placement is shown in Fig. 2.1. The knee kinematics measurement based on the OMoCap system was downsampled to 100 Hz and synchronized with the inertial sensor data.

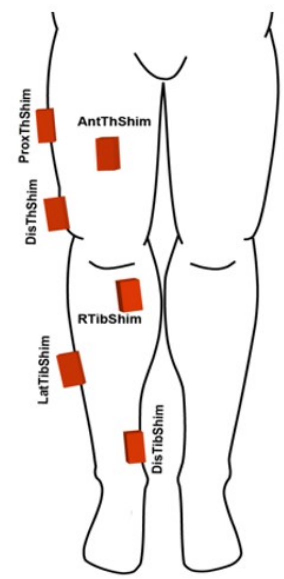


Figure 2.1: The IMU sensor placement on the lower body in WEVAL dataset. DisThShim is the abbreviation for distal thigh Shimmer IMU sensor; LatTibShim is abbreviation for lateral tibia Shimmer IMU sensor. The other names follow the same abbreviation convention.

Table 1: Review of public lower-limb kinematics datasets.

Dataset	Subject Number	Data Source	Motions	Knee Joint DoF	Lower Limb IMU Sensor Placement
MoVi [40]	90	OMoCap system, RGB video cameras, and IMUs	21 daily motions and sports movements	3D	Shank and thigh
UT Foot-Mounted Inertial Navigation Dataset [41]	5	OMoCap system and IMUs	Walking, running, crawling, and stair-climbing	None	Foot
TotalCapture [42]	5	OMoCap system and IMUs	Walking, acting, freestyle and a range of other motions	1D	Shank and thigh
LARa [43]	14	OMoCap system, RGB video cameras, and IMUs	Warehousing activities	None	Ankle
Camargo *et al.* [44]	22	OMoCap system, IMUs, EMG sensors, goniometers and forceplates	Walking and stair climbing	1D	Shank and thigh
Opportunity [45]	4	Multiple sensor systems including IMUs, pressure sensors and etc.	Daily activities and a scripted sequence of activities	None	Knee
TNT15 [46]	5	RGB video cameras and IMUs	Walking, running and three other activities	None	Shank and thigh
Virginia Tech Natural Motion Dataset [39]	17	Inertial Motion Capture System	Daily activities and warehousing activities	3D	Shank and thigh
Gait Analysis Data Base [47]	108	IMUs and EMG sensors	Walking	None	Shank and thigh
Luo *et al.* [48]	30	IMUs	Walking	None	Shank and thigh

Chapter 3

Methodology

The data-driven 3D joint angle estimation approach proposed in this study, namely ResCoCo, is generally a neural network under contrastive learning to estimate 3D knee joint angles from inertial data.

The core architecture of ResCoCo is a feature extractor network. It takes in 3D linear acceleration and angular velocity recorded by a pair of IMUs on the shank and thigh. It is connected with a projection head, trained under self-supervised training, and outputs inertial data representation. Afterwards, it is connected with a linear head, and trained with supervision to estimate 3D knee joint angle. The feature extractor network has a backbone block and a residual block. Both blocks are ConvLSTM networks. The residual block is inspired by the successful residual neural network introduced by He *et al.* [49]. Residual neural networks acquire better accuracy compared to plain networks, because it allows considerably increased network depth. It allows depth increase mainly by alleviating the gradient vanishing problem and accuracy saturation problem, using skip connection for information to flow across layers and ease the optimization process. The skip connection mechanism is shown in

Fig. 3.1. Using the ConvLSTM network in the feature extractor network is because it leverages the advantages of the convolutional layer and LSTM layer in spatial and time domains together, and it is successfully applied in Human Activity Recognition [50, 51, 52, 53].

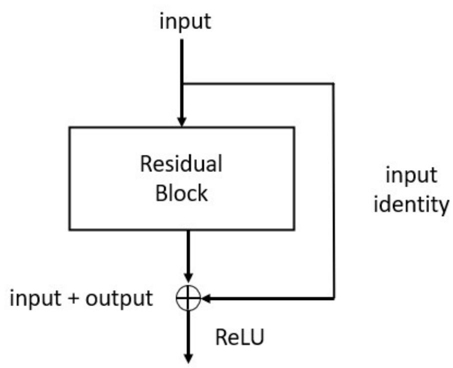


Figure 3.1: The skip connection passes the identical mapping of the residual block input. The block input is concatenated with the block output.

The reason for using contrastive learning is to explicitly capture invariant and disentangled input data representations. Contrastive learning is a class of discriminative approaches for representation learning. Representation learning the process to extract efficient representations from raw input data, and it improves the model performance on downstream tasks. Contrastive learning approaches learn representations by comparing different sample pairs. They have achieved promising performance in Natural Language Processing (NLP) and Computer Vision (CV) [54, 55]. SimCLR is one contrastive learning approach with great representation extraction efficiency proven in Human Pose Estimation (HPE) and HAR [56, 57, 58]. Spurr *et al.* used SimCLR in self-supervised hand pose estimation, encouraging equivariance and invariance in feature representation for geometric and appearance transformation, respectively [57]. This inspired us to adapt SimCLR for efficient and robust representation extraction to resist sensor placement, I2S, and subject variation in 3D joint angle estimation.

The feature extractor network architecture is first introduced in Section 3.1, and then the contrastive learning framework is presented in Section 3.2.

3.1 Feature Extractor Network

The feature extractor network architecture is shown in Fig. 3.2. The residual block extracts the input inertial data features, then the extracted features are concatenated with identity mapping of the original network input and sent to the backbone block. The residual block has four two-dimensional (2D) convolutional layers and one bidirectional LSTM layer. The backbone block has seven 2D convolutional layers and one bidirectional LSTM layer. 2D convolutional layers are utilized to extract spatial features of the inertial sensor signals, because their convolution kernels have larger receptive fields, compared to 1D convolutional layers. Thus, the interrelated features of 3D linear acceleration and angular velocity can be extracted. The bidirectional LSTM layers are used to model the temporal dynamics of the extracted features. They process the extracted features in both forward and backward directions and learn the context more thoroughly from the past and future of the signal. For both blocks, the convolutional layers have 64 kernels with the shape of 3×3. Padding is performed to maintain the feature map shape. A Rectified Linear Unit (ReLU) activation function is used after each convolutional layer. The bidirectional LSTM layers have 64 hidden units and 2 hidden layers. After the bidirectional LSTM layer in the backbone block, the feature map is flattened and inputted into a fully-connected layer, namely the sequence length shortening layer. During the self-supervised training stage, the feature extractor network is connected to a fully-connected layer, namely the projection head, to project the feature map into input inertial data representation. During the supervised training stage, the feature extractor network is connected to a fully-connected layer, namely the linear head, to project the feature map into the latent space for 3D knee joint angle estimation.

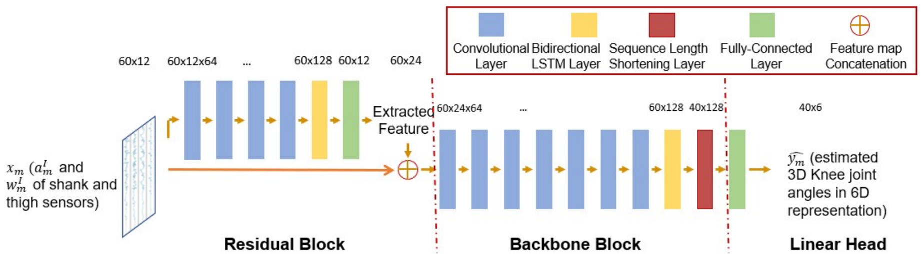


Figure 3.2: The architecture of the feature extractor network and linear head for 3D knee joint angle estimation.

The reason for using the sequence length shortening layer is to discard the output sequence estimated from the bidirectional LSTM layer based on incomplete context. When the feature extractor network without the length shortening layer is used for end-to-end IMU-based 3D knee joint angles estimation on the WEVAL dataset, the estimation error averaged across all estimation sequences is shown in Fig. 3.3. It can be observed that when the model output sequence has the same length as the input sequence, the front and end parts of the estimation sequence have higher MAE than the middle part, especially on the sagittal and transverse planes. The reason can be that the front and back parts of the input sequence have incomplete context for the bidirectional recurrent layer to learn, thus leading to lower estimation accuracy. Under such an assumption, discarding the front and back parts of the estimation sequence might be beneficial for estimation accuracy improvement. Inspired by this, the bidirectional LSTM layer output length is modified by a sequence length shortening layer and the optimal output length is investigated.

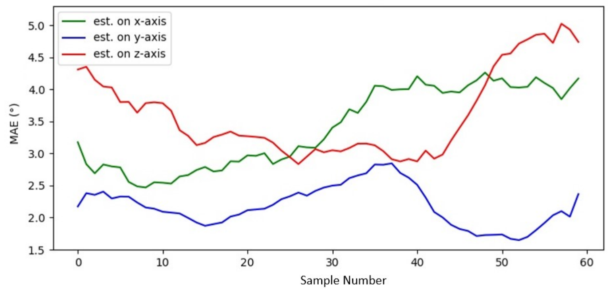


Figure 3.3: Average mean absolute error (MAE) across all 3D knee joint angle estimation sequences derived from ResCoCo without a length shortening layer, the model input and output sequence have the same length.

3.2 Contrastive Learning

The core concept of the contrastive learning approach used in ResCoCo, namely SimCLR, is to map similar pairs of data to be close in the embedding space and push unrelated data pairs apart. It has a self-supervised training stage and a supervised training stage. As shown in Fig. 3.4, for the self-supervised training stage, given a set of inertial sensor data $\{x_m\}_{m=1}^{M}$, x_m is processed with augmentation function $t(\cdot)$ from augmentation set T, $t \in T$. Each stochastic data augmentation function has two components: sensor placement randomization and I2S alignment simulation. Each sample is processed by these two components sequentially. A positive data pair $\{t(x_m), t'(x_m)\}$ is two different augmented views of the same sample x_m and a negative pair $\{t(x_m), t''(x_n)\}$ is different augmented views of different samples t_m and t_n . For the input dataset with M samples, two augmentations are applied on each sample and there are 2M augmented samples in total. For each sample, there will be one corresponding positive sample and 2(M-1) negative samples. All augmented samples are projected into embedding space Z with a feature extractor network $f(\cdot)$ for representation extraction and projection head $g(\cdot)$ to map representation to the space for contrastive loss calculation. The feature extractor network is introduced in Section 3.1. The projection head is a fully-connected layer. The contrastive objective function in use is NT-Xent. It aims to encourage positive pairs to be close and push negative pairs apart in the latent space. It maximizes the agreement between all positive pairs and minimizes the agreement between all negative pairs.

$$l_1 = -log\frac{exp(sim(z_i, z_j)/\tau)}{\sum_{k=1}^{2M} 1_{[k\neq i]}exp(sim(z_i, z_k)/\tau)} \tag{3.1}$$

τ is temperature parameter, $1_{k\neq i} \in \{0,1\}$ is an indicator function that equals to 1 iff $k \neq i$. $sim(u,v) = u^T v/||u||||v||$ is the cosine similarity between data pairs.

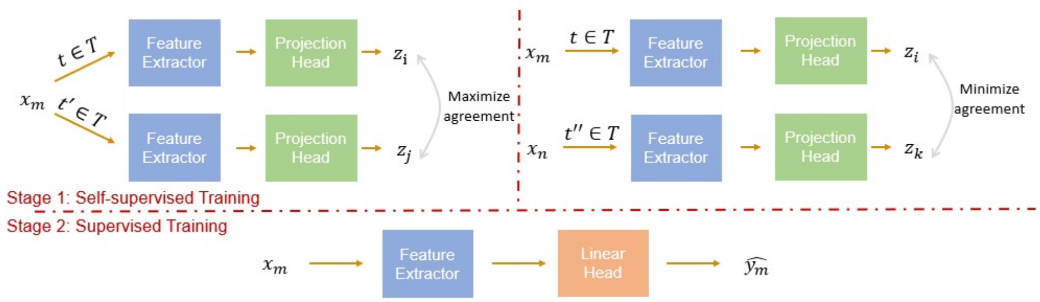


Figure 3.4: The architecture of ResCoCo. In the self-supervised training stage, each input sample is augmented and processed by the feature extractor network and project head. Agreement of positive pairs are maximized and agreement of negative pairs are minimized. In the supervised training stage, the model input is processed with the feature extractor network and linear head.

After the self-supervised training, the feature extractor network is trained under a supervised training scheme. Both raw inertial sensor data and ground truth 3D knee joint angles are used as the model input and label. The model input is inertial sensor signals from a pair of IMU sensors, one on shank and one on thigh, represented by 2D matrices x_m for M windows, $x_m \in R^{L\times(2\times C)}(m = 1, 2, \ldots, M)$. The window length is $L = 60$ for sensor signals of 1 second. The sensor signals include 3D angular velocity a_i^I and linear acceleration w_i^I, thus $C = 6$. x_m is mapped to the model output $y_m \in R^{l \times 6}(m = 1, 2, \ldots, M)$, which is the knee joint angles on three rotation planes in 6D rotation representation for length l . The loss function in use is mean squared error (MSE), calculating the distance between estimated and ground truth 3D knee joint angles.

$$l_2 = \frac{1}{M} \sum_{m=1}^{M} (y_m - \hat{y_m})^2 \tag{3.2}$$

6D rotation representation is used for 3D knee joint angle representation for its continuity. Generally, 3D, 4D or 9D rotation representations (i.e., Euler angles, quaternion, and rotation matrix) can be used to represent 3D rotation. However, it is proven that 3D rotation in four or fewer dimension representations is discontinuous, while continuous rotation representations potentially lead to better neural network performance in practice [59]. To acquire continuous rotation representation efficiently and avoid orthogonalization in calculation, Zhou *et al.* introduced 6D rotation representation in [59]. 6D rotation representation is applied in ResCoCo to represent 3D knee joint angles in the model output.

Two augmentation approaches used in contrastive learning are sensor placement randomization and I2S alignment simulation. Specifically, each model input consists of inertial data from a pair of sensors, one from the shank and the other from the thigh. In the sensor placement randomization, the sensor placement is randomly drawn and their inertial data is added to the training dataset. Note that the shank sensor is randomly chosen from the three shank sensor placements and the thigh sensor is randomly chosen from the three thigh sensor placements, as shown in Fig. 2.1. For the I2S alignment simulation, random rotation $R_m^{II'}$ is applied to each sample x_m to simulate random I2S alignment, as introduced in Section 3.3, and the resulting rotated inertial data is added into the training dataset.

$$
\begin{aligned}
a_m^{I\,\prime} &= a_m^I \cdot R_m^{II\,\prime} \\
w_m^{I\,\prime} &= w_m^I \cdot R_m^{II\,\prime}
\end{aligned}
\tag{3.3}
$$

Chapter 4

Evaluation

To validate the data-driven method proposed in the last chapter, firstly the implementation of a baseline model-driven calibration-free method and the implementation of ResCoCo are introduced in Section 4.1. These two models are evaluated on the WEVAL dataset for knee joint angle estimation and the estimation result is presented in Section 4.2. Firstly, the estimation result of the model-driven approach is presented in Section 4.2.1. In Section 4.2.2, the overall performance of ResCoCo is introduced, and an ablation study is conducted on each component of ResCoCo. The feature extractor network and its component are investigated, and the influence of contrastive learning and the augmentation approaches are investigated. Lastly, the optimal input sensor configuration, including sensor number and sensor placement, is studied.

4.1 Implementation

4.1.1 Model-Driven Method Implementation

The knee joint angle estimation approach proposed in [27] is utilized as the baseline model-driven approach in this study. The knee joint is assumed to be a hinge joint with one Degree of Freedom (DoF) on the sagittal plane. 3D linear acceleration and angular velocity are utilized for angle estimation to avoid magnetic field measurement error caused by ferromagnetic materials and recorded by magnetometer. Generally, the knee flexion/extension angle is estimated by fusing the knee joint angles derived from linear acceleration and angular velocity. The estimation procedure is detailed in Appendix A.1.

WEVAL dataset is used to validate the model-driven calibration-free method. IMUs placed on proximal thigh and lateral tibia in Fig. 2.1 are utilized. To solve the optimization problems in Equation A.4 and Equation A.9, Levenberg–Marquardt algorithm is adopted. The optimization algorithm is repeated 100 times with different random initial points. Experiment results with the smallest loss are saved. The weight coefficient in Equation A.1 is set as 0.01.

4.1.2 ResCoCo Implementation

To evaluate the 3D knee joint angle estimation ability of ResCoCo, the WEVAL dataset is divided into three parts: subject 1 to 12 for training, subject 13 for validation, and subject 14 and 15 for testing. Every experiment in Section 4.2 is repeated three times and the experiment results are averaged. The temperature parameter τ

in Equation 3.1 is set to 0.5. Adam optimizer is used for both self-supervised and supervised training stages, and the learning rate is set to 10^{-3}. The augmented dataset size is empirically chosen and set to five times larger than the original dataset. The rotation $R_m^{II'}$ in Equation 3.3 is represented in extrinsic Euler angle. The range of rotation on the three rotation axes is empirically chosen and set to $[-60°, 60°]$.

To divide the input and label signals for the model with or without a sequence length shortening layer, two sequence segmentation schemes are introduced and shown in Fig. 4.1. In the first segmentation scheme, the input and label samples have the same length L. In the second segmentation scheme, the input length L is no shorter than the label length l. Under both segmentation schemes, the ground truth knee joint angle sequences are segmented in a sliding-window scheme and have the same total length. For example, if $L = 60, l = 40$, and both IMU and ground truth knee joint angle sequence start from t_0. The first input and label sample should be between t_{10} and t_{70}, under the first segmentation scheme. The first input sample should be between t_0 and t_{60}, and the first label sample should be between t_{10} and t_{50}, under the second segmentation scheme. Therefore, the label sequences have the same total length under two segmentation schemes, and the experiment result of models with or without a sequence length shortening layer are comparable. The dataset is segmented with scheme 1 to be used by models without a length shortening layer. The dataset is segmented with scheme 2 to be used by models with a length shortening layer.

Both estimated and ground truth 3D knee joint angles are presented in 6D rotation representation. The estimation results are transformed into Euler angle representation as 3D knee joint angles.

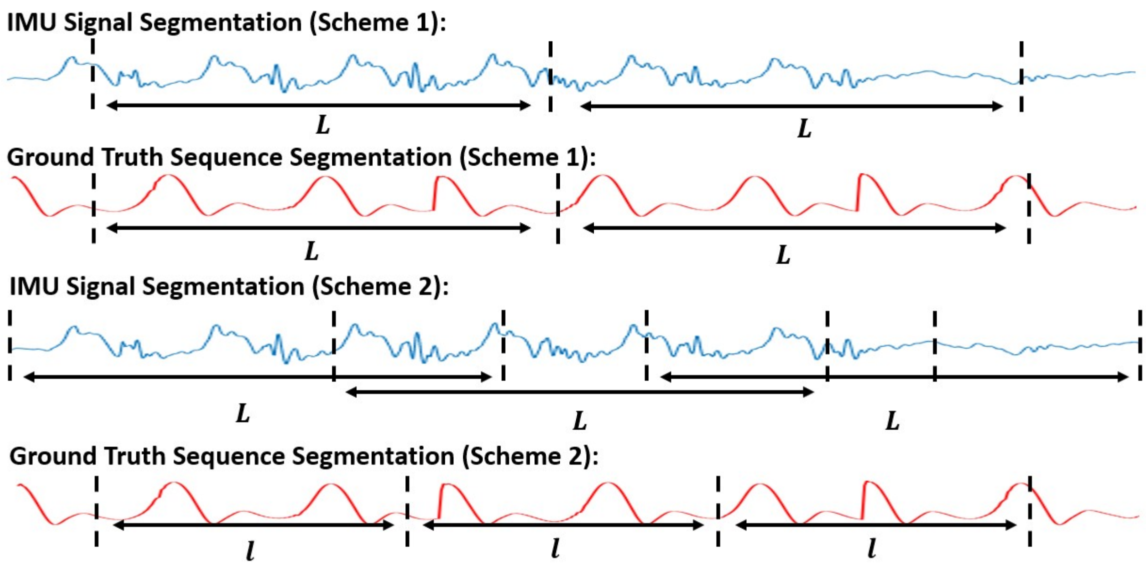


Figure 4.1: Two schemes to segment the input and label signals.

4.2 Estimation Result

4.2.1 Model-Driven Method Performance

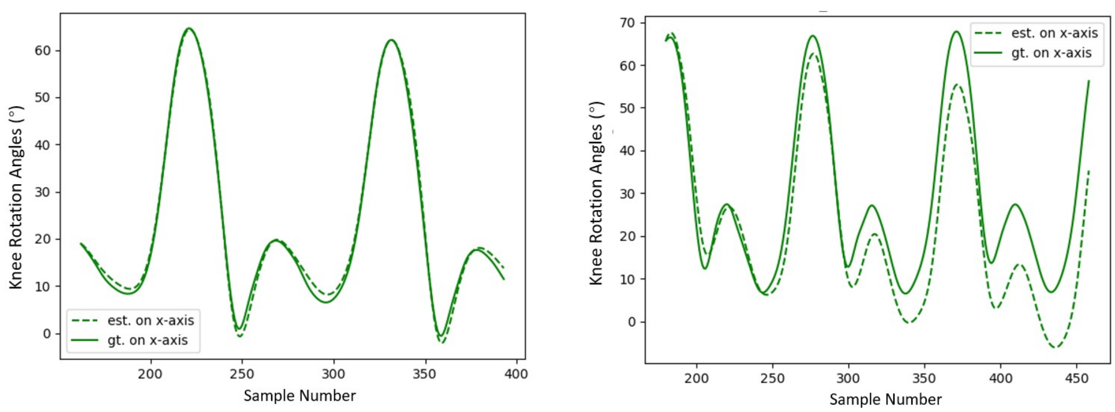
The model-driven method acquires 1D knee joint angle estimation MAE and root mean square error (RMSE) of 3.93° and 5.24°, respectively. The standard deviation for MAE across all subjects is 1.60° and the standard deviation for RMSE across all subjects is 1.97°. The MAE, RMSE of all subjects are reported in Table 4.1. The estimated and ground truth knee joint angles are visualized in Fig. 4.2a.

It can be observed that the model-driven method has inter-subject performance differences. It performs best on subject 15 and worst on subject 11. This can be due to drift issues, noises caused by soft tissue movements, or the hinge joint assumption of the knee joint. The drifting issue caused by gyroscope recordings leads to large errors, as shown in Fig. 4.2b and Table 4.1.

The inaccuracy of the model-driven approach estimation results can be explained

Table 4.1: Performance of the model-based calibration-free method on each subject.

Subject ID	1	2	3	4	5	6	7	8
MAE(°)	4.38	3.27	2.11	4.79	2.48	2.84	3.18	3.40
RMSE(°)	5.42	4.27	2.60	5.83	3.10	3.57	4.11	4.14
Subject ID	9	10	11	12	13	14	15	
MAE(°)	4.51	4.34	7.79	6.78	4.40	2.49	2.01	
RMSE(°)	5.71	5.21	9.87	8.17	5.32	3.10	2.56	



(a) Estimated and ground truth knee joint angle of Subject 15.

(b) Drift issue in the knee joint angle estimation on Subject 11.

Figure 4.2: 1D knee joint angle estimation results of the model-driven calibration-free method.

by that even though the knee joint angle estimated based on linear acceleration compensates for the drift issue, the error caused by drift cannot be fully eliminated. Besides, the estimation accuracy tends to be low if the position vectors are estimated solely based on gait data, because the homogeneous walking pattern has small variation, thus the optimization results tend to fall into the local minimum, leading to less accurate knee joint angle estimation result.

4.2.2 ResCoCo Performance

Overall ResCoCo performance Using the complete ResCoCo framework with contrastive learning can yield accurate 3D knee joint angle estimation results with high robustness towards I2S alignment and sensor placement variations. The 3D knee joint angle estimation accuracy (MAE(°)) of ResCoCo is 4.71°, 3.14° and 5.10° on sagittal, frontal, and transverse planes, using randomly placed sensor pair, and is 3.18°, 3.08° and 4.12° using the same sensor pair as in the supervised training stage. The 1D knee joint angle estimation accuracy (MAE(°)) of the baseline model-driven approach is 3.94° on the sagittal plane. Since ResCoCo does not need calibration procedures, physical constraints or a predefined anatomical model and can accurately estimate knee joint angle on three rotation planes, it is a better solution compared to the model-driven approach. ResCoCo uses inertial data from one pair of IMUs as the model input, which yields an economical and efficient input sensor configuration compared to using single or six sensors as the model input.

Optimal feature extractor network architecture To determine the best-performing architecture of the feature extractor network, it is combined with linear head, trained and tested directly with raw inertial data input and raw ground truth knee joint angle. The raw inertial data is recorded from DisThShim and RTibShim sensors, as shown in Fig. 2.1. This training and testing scheme is similar to other data-driven methods [34, 37, 60].

As induced in Section 3.2, the sequence length modifying layer discards part of the output sequence estimated by bidirectional LSTM layers from incomplete context and leaves the highly accurate part. To acquire the best output sequence length, the sequence length modifying layer is investigated and the resulting 3D knee joint angle

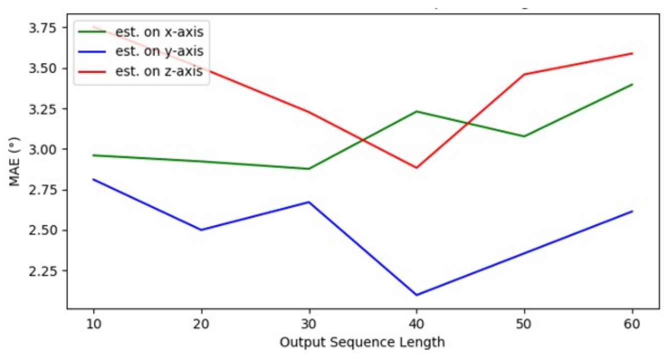


Figure 4.3: 3D knee joint angle MAE(°) of the proposed neural network with different estimation sequence length.

estimation accuracy is shown in Fig. 4.3. It can be observed that decreasing the estimation sequence length moderately from 60 frames to 40 frames can efficiently lift the estimation accuracy, while keeping the total estimation sequence length unchanged. The average MAEs across all output sequences is illustrated in Fig. 4.4. The average MAEs across all estimation sequences are flatter, compared to those in Fig. 3.3. It shows that using the length shortening layer is beneficial for balancing estimation accuracy across the whole output sequence. Therefore, a length shortening layer is used in ResCoCo to modify the feature map length from 60 frames to 40 frames per window.

To validate each component of the feature extractor network, an ablation study is conducted, and the resulting 3D knee joint angle estimation accuracy is listed in Table 4.2. Firstly the length shortening layer and residual block are removed from the feature extractor network, leading to a plain ConvLSTM network. Then the residual block is added to this baseline model and the resulting model is evaluated. It can be observed that residual block can improve the knee joint angle estimation accuracy, especially along the x-axis and y-axis. Lastly, the feature extractor network with

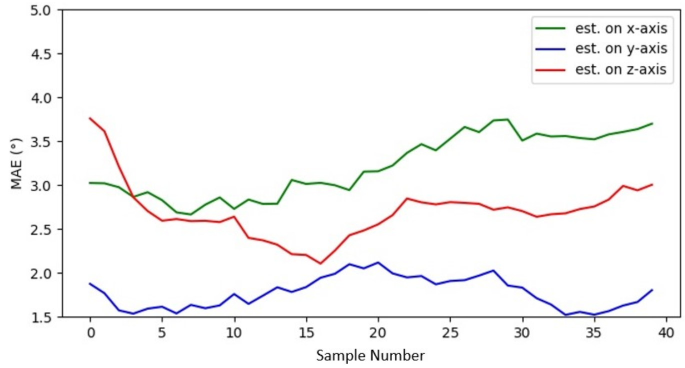


Figure 4.4: Average MAE across 3D knee joint angle estimation sequences from the feature extractor network and linear head.

both the residual block and length shortening layer is evaluated. It can be observed that the length shortening layer leads to a large performance improvement, especially on the transverse plane. The estimated and ground truth 3D knee joint angles are visualized in Fig. 4.5.

Table 4.2: Influence of each component of ResCoCo on the 3D knee joint angle estimation accuracy (MAE(°)). The estimation accuracy of each axis is normalized with the knee motion range on that axis and recorded in parenthesis.

	x-axis	y-axis	z-axis
Plain ConvLSTM Network	3.54 (0.05)	2.32 (0.13)	3.60 (0.13)
+ Residual Block	3.42 (0.05)	2.28 (0.13)	3.90 (0.14)
+ Residual Block + Length Shortening Layer	**3.23 (0.05)**	**2.10 (0.12)**	**2.88 (0.10)**

Influence of contrastive learning To investigate the influence of contrastive learning and the model robustness towards variation, the complete ResCoCo is tested. This model is first trained under self-supervised training and then fine-tuned under supervision, with raw inertial data recorded by DisThShim and RTibShim IMUs

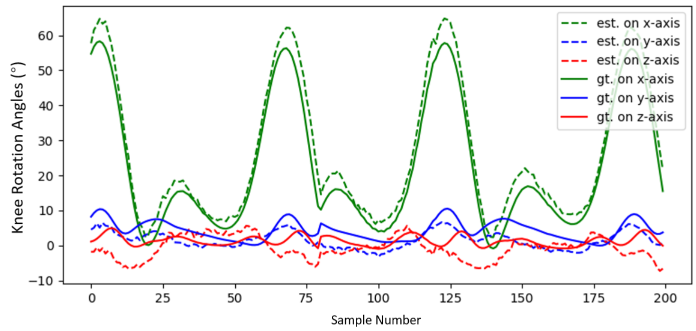


Figure 4.5: Ground truth and 3D knee joint angles estimated by ResCoCo without contrastive learning.

as in Fig. 2.1. To evaluate the model robustness towards sensor placement and I2S alignment variation, models are tested with inertial data from randomly placed sensor pairs with random or real I2S alignment.

As proven in [56], the batch size in self-supervised training greatly influences the model performance, because more data pairs are produced for comparison in each batch when the batch size is larger. Different batch sizes in self-supervised training are investigated and the resulting 3D knee joint angle estimation accuracy of ResCoCo is shown in Fig. 4.6. It can be observed that the large batch size can lift the estimation accuracy and the accuracy does not further increase after reaching the batch size of 256. Thus the batch size is set to 256.

The ResCoCo with or without contrastive learning is tested with randomly placed sensor pairs and simulated/real I2S alignment. The result is listed in Table 4.3. It can be observed that training the neural network without contrastive learning makes the network overfitted to one specific sensor placement and have drastically performance drop when the sensor placement is randomized. The 3D knee joint angle estimation accuracy increases on three rotation axes after applying contrastive learning in ResCoCo. Training the model with contrastive learning greatly improves

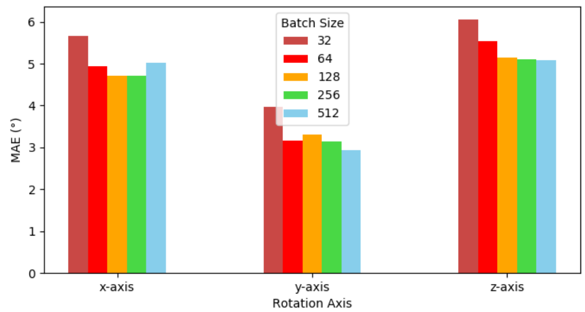


Figure 4.6: 3D knee joint angle MAE (°) of ResCoCo with different batch size in self-supervised training. Models are tested with inertial data from randomly placed sensor pair and real I2S alignment.

the model resistance towards sensor placement and I2S alignment variation.

To further analyze the influence of augmentation approaches on the robustness of ResCoCo, two augmentation approaches are split and used separately in the self-supervised training stage. The remaining training procedure and augmented dataset size stay the same. The experiment results are listed in Table 4.3. Using sensor placement randomization as the only augmentation method in the self-supervised training stage leads to similar estimation accuracy as only using I2S alignment simulation as the augmentation approach. Combining sensor placement randomization and I2S alignment simulation leads to the best estimation accuracy and model robustness, especially on the sagittal plane. Therefore, both augmentation approaches force the model to learn robust representations and resist different kinds of variations. Combining these two augmentation approaches leads to the best model performance and robustness.

One of the augmentation components in contrastive learning, I2S alignment simulation is to apply a random rotation onto the input acceleration and angular velocity.

Table 4.3: Influence of augmentation methods in the contrastive learning on the 3D knee joint angle estimation accuracy (MAE(°)) of ResCoCo. The estimation accuracy of each axis is normalized with the knee motion range on that axis and recorded in the parenthesis.

	On Random Sensor Placement and Random I2S Alignment			On Random Sensor Placement and Real I2S Alignment		
	x-axis	y-axis	z-axis	x-axis	y-axis	z-axis
ResCoCo w/o CL	10.12 (0.15)	3.90 (0.21)	6.22 (0.23)	7.01 (0.10)	3.28 (0.18)	5.19 (0.19)
ResCoCo with Sensor Placement Randomization in CL	8.06 (0.12)	3.55 (0.20)	5.93 (0.22)	5.37 (0.08)	**3.14 (0.17)**	5.22 (0.19)
ResCoCo with I2S Alignment Simulation in CL	8.73 (0.13)	3.38 (0.19)	5.88 (0.21)	6.81 (0.10)	3.17 (0.18)	**4.61 (0.17)**
ResCoCo with Two Augmentation Methods in CL	**5.35 (0.08)**	**3.26 (0.18)**	**5.36 (0.20)**	**4.71 (0.07)**	**3.14 (0.17)**	5.10 (0.19)

This random rotation is represented by extrinsic Euler angles, and its range is empirically chosen between $-60°$ and $60°$ on three rotation axes. The rotation range of simulated I2S alignment has been modified and the estimation accuracy of resulting models is as shown in 4.4. The resulting model estimation accuracy and generalization ability do not have obvious improvement. The I2S alignment can be simulated more accurately in future work than applying random rotation with the same range on three rotation axes.

Rotation range of the random rotation applied onto I2S alignment is changed and performance of resulting ResCoCo is listed in Table 4.4.

ResCoCo utilized two augmentation approaches, sensor placement randomization

Table 4.4: 3D knee joint angle estimation accuracy (MAE(°)) of ResCoCo with different rotation range of the random rotation in I2S alignment simulation. They are tested on inertial data from random placed sensor pair and real I2S alignment.

Rotation Range	x-axis	y-axis	z-axis
$[-60°, 60°]$	4.71 (0.07)	**3.14 (0.17)**	**5.10 (0.19)**
$[-120°, 120°]$	4.52 (0.07)	3.62 (0.20)	5.43 (0.20)
$[-180°, 180°]$	**4.11 (0.06)**	3.28 (0.18)	5.58 (0.20)

and I2S alignment simulation, in the self-supervised training stage. There exists another way of using these two augmentation approaches, which is to directly add the augmented data to the training dataset and train the neural network under supervision. The estimation accuracy of the resulting model is listed in Table 4.5. Directly augmenting the training dataset leads to better estimation accuracy and model robustness towards variations. However, it is less data-efficient, because it requires ground truth 3D knee joint angle label. The ground truth 3D knee joint angle is derived from OMoCap system recordings, which is cumbersome to collect and process, and requires expensive lab environment and expertise. ResCoCo only requires IMU recordings from different placements, which is easier and cheaper to acquire.

It can be observed that generally the knee joint angle estimation accuracy of all models is lower on the transverse and frontal planes compared to the estimation accuracy on the sagittal plane. This may be caused by a variety of factors. First, the ground truth knee joint angles in the transverse and frontal planes may themselves

Table 4.5: Comparison of 3D knee joint angle estimation accuracy (MAE(°)) of the feature extractor network trained with augmented training dataset, and ResCoCo with augmentation approaches embedded in the CL framework.

	On Random Sensor Placement and Random I2S Alignment			On Random Sensor Placement and Real I2S Alignment		
	x-axis	y-axis	z-axis	x-axis	y-axis	z-axis
Trained with Augmented Training Dataset	**3.86 (0.06)**	**2.75 (0.15)**	**4.87 (0.18)**	**2.39 (0.04)**	**2.58 (0.14)**	**4.20 (0.15)**
Augmentation Method Embedded in CL	5.35 (0.08)	3.26 (0.18)	5.36 (0.20)	4.71 (0.07)	3.14 (0.17)	5.10 (0.19)

have potentially lower accuracy than knee joint angles in the sagittal plane. For example, even small misplacement of joint markers on the transverse plane (e.g., markers placed anterior or posterior to the joint centre) can have large effects on transverse plane angles (e.g., 10mm misplacement can result in nearly 8° of transverse plane error) [61, 62]. Therefore, knee marker misplacement may be minor and undetectable, but still causes noticeable differences in knee joint angle errors, making it difficult for data-driven models to estimate these angles. Additionally, knee joint rotation patterns in the transverse and frontal planes have more inter-subject variations than those on the sagittal plane, making them more difficult to model.

The 3D knee joint angle estimation accuracy of ResCoCo is different across subjects. The estimation accuracy on the two test subjects is listed separately in Table 4.6. It can be observed that that inter-subject differences can greatly influence the model estimation accuracy and robustness towards subject variation needs to be further improved.

Table 4.6: 3D knee joint angle estimation accuracy (MAE(°)) of ResCoCo on different subjects.

	On Random Sensor Placement and Random I2S Alignment			On Random Sensor Placement and Real I2S Alignment		
	x-axis	y-axis	z-axis	x-axis	y-axis	z-axis
Subject 14	**4.99 (0.07)**	3.41 (0.19)	**4.65 (0.17)**	**4.42 (0.07)**	3.69 (0.20)	**5.05 (0.18)**
Subject 15	5.58 (0.08)	**3.12 (0.17)**	7.33 (0.27)	5.05 (0.07)	**2.63 (0.15)**	6.03 (0.22)

Sensor Configuration In the experiments above, the input of ResCoCo in the supervised training stage is inertial data recorded by a pair of IMUs placed on the right tibia and distal thigh. The sensor number can be reduced or increased, and the sensor pair can have a different placements. To investigate the most efficient input sensor configuration, the input sensor number and placement are investigated.

Firstly, the two-sensor setting is investigated. The model input is inertial data recorded by one pair of sensor, one on the shank and the other on the thigh. The self-supervised training stage stays the same. For the supervised training stage, the model input is inertial data recorded by one certain pair of sensor. There are three shank sensors and three thigh sensors, leading to nine sensor placement combinations in total. The 3D knee joint angle estimation accuracy of models with different sensor pair placements is listed in Table 4.7. All two-sensor models obtain similarly good estimation results. Using IMUs on the right tibia and the distal thigh as the model input in the supervised training stage leads to the best 3D knee joint angle estimation accuracy with a MAE of 4.71°, 3.14°, 5.10° in the sagittal, frontal and transverse planes, respectively.

Table 4.7: Influence of sensor placement on the 3D knee joint angle estimation accuracy (MAE(°)) with inertial data from two sensors as the model input. The estimation accuracy of each axis is normalized with the knee motion range on that axis and recorded in the parenthesis.

Sensor Placement	On Random Sensor Placement and Random I2S Alignment			On Random Sensor Placement and Real I2S Alignment		
	x-axis	y-axis	z-axis	x-axis	y-axis	z-axis
DisTh RTib	5.35 (0.08)	**3.26 (0.18)**	**5.36 (0.20)**	4.71 (0.07)	**3.14 (0.17)**	**5.10 (0.19)**
AntTh RTib	**5.20 (0.08)**	3.27 (0.18)	5.77 (0.21)	4.89 (0.07)	3.37 (0.19)	5.89 (0.22)
ProxTh RTib	5.71 (0.08)	3.63 (0.20)	5.86 (0.21)	5.58 (0.08)	3.39 (0.19)	5.78 (0.21)
DisTh DisTib	5.26 (0.08)	3.31 (0.18)	5.37 (0.20)	3.88 (0.06)	3.37 (0.19)	5.18 (0.19)
AntTh DisTib	5.33 (0.08)	3.39 (0.19)	5.75 (0.21)	5.01 (0.07)	3.34 (0.19)	6.21 (0.23)
ProxTh DisTib	5.75 (0.09)	3.57 (0.20)	5.55 (0.20)	**4.60 (0.07)**	3.80 (0.21)	5.99 (0.22)
DisTh LatTib	6.35 (0.09)	3.86 (0.21)	6.10 (0.22)	5.60 (0.08)	3.85 (0.21)	5.72 (0.21)
AntTh LatTib	5.94 (0.09)	3.87 (0.21)	6.11 (0.22)	5.82 (0.08)	3.57 (0.20)	5.96 (0.22)
ProxTh LatTib	6.86 (0.10)	3.47 (0.19)	5.89 (0.21)	6.17 (0.09)	3.55 (0.20)	5.82 (0.21)
SD	0.53	0.22	0.26	0.67	0.21	0.35

Secondly, the one-sensor setting is investigated. The model input is inertial data recorded by one sensor either on the shank or thigh. During the self-supervised training stage, the I2S alignment simulation augmentation method remains the same. For the sensor placement randomization part, the sensor is randomly drawn from the six sensors and its recording is added to the training dataset for augmentation. The supervised training stage stays the same, except inertial data from only one certain sensor is used as the model input. The 3D knee joint angle estimation accuracy of the resulting models is listed in 4.8. Using IMU on the distal thigh as the model input in the supervised training stage gives the best estimation accuracy with a MAE of 8.11°, 3.73°, 5.73° in the sagittal, frontal, and transverse planes, respectively.

The standard deviation of 3D knee joint angle estimation error across all two-sensor models is lower than the standard deviation of estimation error across all one-sensor models. It can be concluded that sensor placement modification does not lead to large performance change when the model takes in inertial data from two sensors. The two-sensor models have higher estimation accuracy and robustness towards input sensor placement variation. The reason can be that a pair of sensors bring comparatively more abundant and efficient information than in the one-sensor situation.

Lastly, the six-sensor setting is investigated. During the self-supervised training stage, the I2S alignment simulation procedure stays the same. Three thigh sensors and three shank sensors are selected in sensor placement randomization, each thigh sensor is randomly drawn from the three thigh sensors and each shank sensor is randomly drawn from the three shank sensors. The resulting inertial data is added to the training dataset for augmentation. The supervised training stage stays the same, except inertial data from all six sensors is used as the model input. The 3D knee joint angle estimation accuracy of the resulting model is listed in Table 4.9.

Table 4.8: Influence of sensor placement on the 3D knee joint angle estimation accuracy (MAE(°)) with inertial data from one sensor as the model input. The estimation accuracy of each axis is normalized with the knee motion range on that axis and recorded in the parenthesis.

Sensor Placement	On Random Sensor Placement and Random I2S Alignment			On Random Sensor Placement and Real I2S Alignment		
	x-axis	y-axis	z-axis	x-axis	y-axis	z-axis
RTib	10.53 (0.16)	4.22 (0.23)	6.13 (0.22)	10.47 (0.16)	4.31 (0.24)	6.10 (0.22)
DisTib	10.74 (0.16)	4.81 (0.27)	6.71 (0.25)	10.45 (0.15)	4.83 (0.27)	6.69 (0.24)
LatTib	**8.49 (0.13)**	4.15 (0.23)	6.36 (0.23)	8.79 (0.13)	4.13 (0.23)	6.43 (0.24)
DisTh	9.53 (0.14)	3.89 (0.22)	6.21 (0.23)	**8.11 (0.12)**	3.73 (0.21)	**5.73 (0.21)**
AntTh	12.22 (0.18)	**3.79 (0.21)**	6.84 (0.25)	9.82 (0.15)	**3.61 (0.20)**	6.52 (0.24)
ProxTh	10.39 (0.15)	3.90 (0.22)	**6.05 (0.22)**	8.65 (0.13)	3.75 (0.21)	5.83 (0.21)
SD	1.14	0.34	0.29	0.91	0.42	0.36

The best-performing one-sensor model, two-sensor model and six-sensor model are selected and their estimation accuracy is illustrated in Fig. 4.7. Using inertial data from six IMUs yields the best knee joint angle estimation accuracy along three axes with random sensor placement and real/random I2S alignment, while using one IMU yields the worst estimation accuracy. To evaluate the statistical significance of the superiority of six-sensor or two-sensor models, we further obtain the MAEs of the one-sensor, two-sensor and six-sensor models through leave-of-one-subject experiments of every subject in the dataset. The resulting MAEs are then used to estimate p-values

Table 4.9: Influence of sensor placement on the 3D knee joint angle estimation accuracy (MAE(°)) with inertial data from six sensors as the model input. The estimation accuracy of each axis is normalized with the knee motion range on that axis and recorded in the parenthesis.

	x-axis	y-axis	z-axis
On Random Sensor Placement and Real I2S Alignment	4.37 (0.06)	2.99 (0.17)	4.96 (0.18)
On Random Sensor Placement and Random I2S Alignment	4.72 (0.07)	3.09 (0.17)	5.22 (0.19)

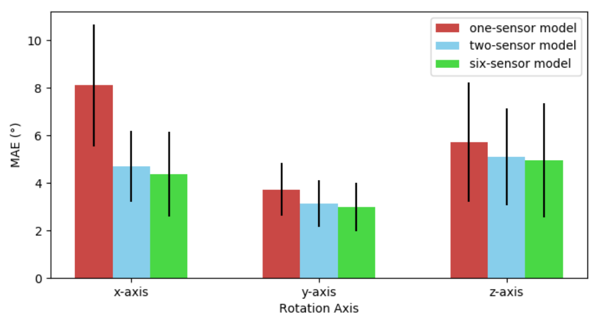


Figure 4.7: 3D knee joint angle MAE (°) of ResCoCo with different input sensor number. Models are tested with real I2S alignment and randomized sensor pair placement. The error bars represent standard deviation.

from t-test. The result is listed in Table 4.10. It can be observed that the one-sensor model is statistically superior than the two-sensor model in joint angle estimation in the sagittal plane. In contrast, the differences between the one-sensor and two-sensor models in frontal and transverse planes and those between the two-sensor and six-sensor models in all planes are statistically insignificant.

The reason can be that models with one sensor or two sensors as the input tend to have deficient information for 3D knee joint angle estimation and learn the homogeneity of gait patterns across healthy individuals and have worse resistance towards

Table 4.10: P-values of t-test on subject-wise MAEs of different models.

	x-axis	y-axis	z-axis
One-Sensor Model and Two-Sensor Model	4.25×10^{-6}	0.40	0.28
Two-Sensor Model and Six-Sensor Model	0.74	0.30	0.96

variation. However, the best two-sensor model has close performance compared to the six-sensor model, which makes it an efficient and economical alternative. Another phenomenon worth noticing is that models with different input sensor configurations generally yield better estimation results using inertial data with real I2S alignment, compared to using inertial signals with simulated I2S alignment. It shows that the models perform better estimating inertial data with the familiar I2S alignment pattern as in the training dataset, and potentially need better generalization ability towards I2S alignment pattern shift.

Chapter 5

Conclusion and Future Work

An IMU-based data-driven 3D joint angle estimation approach, namely ResCoCo, is proposed in this study. It uses a ConvLSTM network as the backbone model, utilizes residual block for network depth increase; estimation sequence length shortening layer to discard output sequence estimated from incomplete context; contrastive learning to increase the robustness of extracted representation. Two augmentation approaches are proposed in contrastive learning, sensor placement randomization and I2S alignment simulation. ResCoCo is validated for 3D knee joint angle estimation with the kinematics dataset WEVAL. It has been proven that the sequence shortening layer and residual block result in model estimation accuracy increase. Contrastive learning is beneficial for model robustness improvement towards sensor placement and I2S alignment variations. ResCoCo can be applied to other complex joints such as the elbow and ankle and further applied to full-body human pose estimation.

Furthermore, the sensor configuration for the model input is investigated. It has been found that using inertial data recorded by six sensors as the data source leads to the best estimation result and model generalization ability, compared to using single or two IMUs. Using a pair of IMUs as the model input is an economic alternative with good estimation accuracy and model robustness.

ResCoCo does not require predefined anatomical model, calibration procedure, physical constraint, or data source other than IMU, and yields accurate 3D knee joint angle estimation results, compared to its model-driven counterpart with the same IMU configuration. Therefore, ResCoCo is an efficient and economical 3D knee joint angle estimation approach than data-driven methods.

ResCoCo has only been validated on walking data of healthy subjects. Its good performance might benefit from stereotypical gait movement. It potentially needs further improvement in its generalization ability on different motions. ResCoCo has only been validated on two subjects in most of the experiments. Further investigation of estimation accuracy and generalization ability of ResCoCo across different subjects is needed in the future.

Appendix A

Appendix

A.1 Model-Driven Method

A.1.1 Methodology Details

To estimate the knee joint flexion/extension angle, one IMU placed on thigh and another placed on shank are used and noted as p, q respectively. The inertial data collected by IMUs includes 3D linear acceleration a_p, a_q and 3D angular velocity g_p, g_q. To avoid the magnetic disturbances caused by ferromagnetic materials, magnetometer recordings are not used. The inertial data is recorded in sensor local coordinate frames. Knee flexion/extension angle $\alpha(t)$ at time t is estimated by fusing the angles estimated by accelerometers α_{acc} and gyroscopes α_{gyro}.

$$\alpha(t) = w_1\alpha_{acc}(t) + (1 - w_1)\alpha_{gyro}(t) \tag{A.1}$$

where w_1 is the weight coefficient, $w_1 \in [0, 1]$. The gyroscope provides angular velocity estimation with potential error caused by drift, while accelerometer does not have drift error cased by integration, yet its readings are less accurate. To compensate the aforementioned errors, fusing the measurements from accelerometer and gyroscope theoretically provides better accuracy. It has been found that fusing the measurements through weight yields good results.

Knee flexion/extension angle α_{gyro} derived from gyroscope is estimated by the integration of relative angular velocities of thigh and shank around the knee flexion/extension axis X. Since the detected angular velocities are in the sensor frames, they need to be projected to the axis x first.

$$\alpha_{gyro}(t) = \int_0^t (g_p(\Delta t)X_p - g_q(\Delta t)X_q)d\Delta t \tag{A.2}$$

g_s represents the angular velocity recorded by the shank IMU or thigh IMU, $s \in \{p, q\}$. The axis vector X_p and X_q in the frames of the corresponding sensor are estimated based on the hinge joint assumption. That is, the knee is assumed to be a hinge joint in daily movement since the movements of adduction/abduction and internal/external rotation are relatively small compared to flexion/extension. Therefore, the projections of angular velocities to the joint planes should be congruent at any time. The joint planes are vertical to the knee joint axis.

$$||g_p(t) \times X_p|| = ||g_q(t) \times X_q||, \forall t \tag{A.3}$$

X_p and X_q are optimized with the objective shown in Equation A.4.

$$\min_{X_p, X_q} \sum_t (||g_p(t) \times X_q|| - ||g_p(t) \times X_q||)^2 \tag{A.4}$$

The direction of the axes could be determined based on prior knowledge of sensor mounting. Based on the axes X_p and X_q, two local joint planes f_p and f_q could be derived.

$$\begin{aligned} f_s &= \{x_s, y_s\}, s \in \{p, q\} \\ x_s &= X_s \times V \\ y_s &= X_s \times x_s \end{aligned} \tag{A.5}$$

where V could be any 3×1 vector that is not paralleled with X_p and X_q. f_p and f_q show the orientations of the knee in the sensor frame.

Knee flexion/extension angle derived from accelerometer α_{acc} is estimated by the relative angles between accelerations a_p^P and a_q^P of the knee joints in the local plane frames f_p and f_q.

$$\alpha_{acc}(t) = \arccos \frac{a_p^{PT} \cdot a_q^P}{||a_p^p|| \cdot ||a_q^p||} \tag{A.6}$$

To estimate a_p^P and a_q^P, the acceleration vectors of the knee joint is estimated first. Then, they are projected to the joint planes in each local frame.

Sensor acceleration could be taken as the sum of the knee joint center acceleration and acceleration due to the rotation of one sensor around the knee joint center. Therefore, the knee joint acceleration could be estimated based on Equation A.7.

$$a_s^k(t) = a_s(t) - g_s(t) \times (g_s(t) \times r_s) - \dot{g}_s(t) \times r_s, s \in \{p, q\} \tag{A.7}$$

where $a_s^k(t)$ denotes the knee joint acceleration in sensor frame p or q. $\dot{g}_s$ is the first order derivative of the angular velocity. r_s is the vector pointing to the sensor from the knee joint center. r_s is estimated through optimization. Then $a_s^k(t)$ is projected to the local joint plane as shown in Equation A.8.

$$a_s^P(t) = [x_s^T \cdot a_s^k(t), y_s^T \cdot a_s^k(t)]^T \tag{A.8}$$

$a_s^P(t)$ is the knee joint acceleration w.r.t f_p and f_q. Thus the differential orientation between $a_p^P(t)$ and $a_q^P(t)$ in Equation A.6 represents the angle between the thigh and the shank at time t, given there is no relative movement between sensors and the body segments.

To estimate r_i in Equation A.7, the optimization objective is formulated as follow:

$$\min_{r_p, r_q} \sum_t (||a_p^k(t)|| - ||a_q^k(t)||)^2 \tag{A.9}$$

Bibliography

[1] R. Baker, A. Esquenazi, M. G. Benedetti, and K. Desloovere, "Gait analysis: clinical facts," *Eur. J. Phys. Rehabil. Med*, vol. 52, no. 4, pp. 560–574, 2016.

[2] M. Gholami, A. Rezaei, T. J. Cuthbert, C. Napier, and C. Menon, "Lower body kinematics monitoring in running using fabric-based wearable sensors and deep convolutional neural networks," *Sensors*, vol. 19, no. 23, p. 5325, 2019.

[3] D. Hamacher, N. Singh, J. H. Van Dieën, M. Heller, and W. R. Taylor, "Kinematic measures for assessing gait stability in elderly individuals: a systematic review," *Journal of The Royal Society Interface*, vol. 8, no. 65, pp. 1682–1698, 2011.

[4] M. A. Ritter and E. D. Campbell, "Effect of range of motion on the success of a total knee arthroplasty," *The Journal of arthroplasty*, vol. 2, no. 2, pp. 95–97, 1987.

[5] R. Argent, S. Drummond, A. Remus, M. O'Reilly, and B. Caulfield, "Evaluating the use of machine learning in the assessment of joint angle using a single inertial sensor," *Journal of Rehabilitation and Assistive Technologies Engineering*, vol. 6, p. 2055668319868544, 2019.

[6] A. Findlow, J. Goulermas, C. Nester, D. Howard, and L. Kenney, "Predicting lower limb joint kinematics using wearable motion sensors," *Gait & posture*, vol. 28, no. 1, pp. 120–126, 2008.

[7] T. Zimmermann, B. Taetz, and G. Bleser, "Imu-to-segment assignment and orientation alignment for the lower body using deep learning," *Sensors*, vol. 18, no. 1, p. 302, 2018.

[8] E. S. Grood and W. J. Suntay, "A joint coordinate system for the clinical description of three-dimensional motions: application to the knee," *Journal of biomechanical engineering*, vol. 105, no. 2, pp. 136–144, 1983.

[9] R. Takeda, S. Tadano, A. Natorigawa, M. Todoh, and S. Yoshinari, "Gait posture estimation using wearable acceleration and gyro sensors," *Journal of biomechanics*, vol. 42, no. 15, pp. 2486–2494, 2009.

[10] L. S. Vargas-Valencia, A. Elias, E. Rocon, T. Bastos-Filho, and A. Frizera, "An imu-to-body alignment method applied to human gait analysis," *Sensors*, vol. 16, no. 12, p. 2090, 2016.

[11] K. J. O'Donovan, R. Kamnik, D. T. O'Keeffe, and G. M. Lyons, "An inertial and magnetic sensor based technique for joint angle measurement," *Journal of biomechanics*, vol. 40, no. 12, pp. 2604–2611, 2007.

[12] J. Favre, R. Aissaoui, B. M. Jolles, J. A. de Guise, and K. Aminian, "Functional calibration procedure for 3d knee joint angle description using inertial sensors," *Journal of biomechanics*, vol. 42, no. 14, pp. 2330–2335, 2009.

[13] J. Favre, B. Jolles, R. Aissaoui, and K. Aminian, "Ambulatory measurement of 3d knee joint angle," *Journal of biomechanics*, vol. 41, no. 5, pp. 1029–1035, 2008.

[14] B. Bouvier, S. Duprey, L. Claudon, R. Dumas, and A. Savescu, "Upper limb kinematics using inertial and magnetic sensors: Comparison of sensor-to-segment calibrations," *Sensors*, vol. 15, no. 8, pp. 18 813–18 833, 2015.

[15] R. V. Vitali and N. C. Perkins, "Determining anatomical frames via inertial motion capture: A survey of methods," *Journal of Biomechanics*, vol. 106, p. 109832, 2020.

[16] Y. Ohtaki, K. Sagawa, and H. Inooka, "A method for gait analysis in a daily living environment by body-mounted instruments," *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, vol. 44, no. 4, pp. 1125–1132, 2001.

[17] J. F. Lin and D. Kulić, "Human pose recovery using wireless inertial measurement units," *Physiological measurement*, vol. 33, no. 12, p. 2099, 2012.

[18] M. Makikawa, S. Kurata, Y. Higa, Y. Araki, and R. Tokue, "Ambulatory monitoring of behavior in daily life by accelerometers set at both-near-sides of the joint," in *MEDINFO 2001.* IOS Press, 2001, pp. 840–843.

[19] K. Liu, Y. Inoue, and K. Shibata, "Physical-sensor and virtual-sensor based method for estimation of lower limb gait posture using accelerometers and gyroscopes," *Journal of Biomechanical Science and Engineering*, vol. 5, no. 4, pp. 472–483, 2010.

[20] G. Cooper, I. Sheret, L. McMillian, K. Siliverdis, N. Sha, D. Hodgins, L. Kenney, and D. Howard, "Inertial sensor-based knee flexion/extension angle estimation," *Journal of biomechanics*, vol. 42, no. 16, pp. 2678–2685, 2009.

[21] L. Gastaldi1a, V. Agostini, R. Takeda, S. Pastorelli, and S. Tadano, "Evaluation of the performances of two wearable systems for gait analysis: a pilot study," *International Journal of Applied Engineering Research*, vol. 11, no. 16, pp. 8820–8827, 2016.

[22] P. Picerno, A. Cereatti, and A. Cappozzo, "Joint kinematics estimate using wearable inertial and magnetic sensing modules," *Gait & posture*, vol. 28, no. 4, pp. 588–595, 2008.

[23] W. Teufl, M. Miezal, B. Taetz, M. Fröhlich, and G. Bleser, "Validity, test-retest reliability and long-term stability of magnetometer free inertial sensor based 3d joint kinematics," *Sensors*, vol. 18, no. 7, p. 1980, 2018.

[24] P. Müller, M.-A. Bégin, T. Schauer, and T. Seel, "Alignment-free, self-calibrating elbow angles measurement using inertial sensors," *IEEE journal of biomedical and health informatics*, vol. 21, no. 2, pp. 312–319, 2016.

[25] E. Allseits, K. J. Kim, C. Bennett, R. Gailey, I. Gaunaurd, and V. Agrawal, "A novel method for estimating knee angle using two leg-mounted gyroscopes for continuous monitoring with mobile health devices," *Sensors*, vol. 18, no. 9, p. 2759, 2018.

[26] G. Bleser, B. Taetz, M. Miezal, C. A. Christmann, D. Steffen, and K. Regenspurger, “Development of an inertial motion capture system for clinical application,” *i-com*, vol. 16, no. 2, pp. 113–129, 2017.

[27] T. Seel, J. Raisch, and T. Schauer, “Imu-based joint angle measurement for gait analysis,” *Sensors*, vol. 14, no. 4, pp. 6891–6909, 2014.

[28] M. Vrigkas, C. Nikou, and I. A. Kakadiaris, “A review of human activity recognition methods,” *Frontiers in Robotics and AI*, vol. 2, p. 28, 2015.

[29] Z. Han, J. Zhao, H. Leung, K. F. Ma, and W. Wang, “A review of deep learning models for time series prediction,” *IEEE Sensors Journal*, vol. 21, no. 6, pp. 7833–7848, 2019.

[30] T. P. Luu, K. Low, X. Qu, H. Lim, and K. Hoon, “An individual-specific gait pattern prediction model based on generalized regression neural networks,” *Gait & posture*, vol. 39, no. 1, pp. 443–448, 2014.

[31] H. Lim, B. Kim, and S. Park, “Prediction of lower limb kinetics and kinematics during walking by a single imu on the lower back using machine learning,” *Sensors*, vol. 20, no. 1, p. 130, 2019.

[32] E. Dorschky, M. Nitschke, C. F. Martindale, A. J. Van den Bogert, A. D. Koelewijn, and B. M. Eskofier, “Cnn-based estimation of sagittal plane walking and running biomechanics from measured and simulated inertial sensor data,” *Frontiers in bioengineering and biotechnology*, vol. 8, p. 604, 2020.

[33] J. Conte Alcaraz, S. Moghaddamnia, and J. Peissig, "Efficiency of deep neural networks for joint angle modeling in digital gait assessment," *EURASIP Journal on Advances in Signal Processing*, vol. 2021, no. 1, pp. 1–20, 2021.

[34] V. Hernandez, D. Dadkhah, V. Babakeshizadeh, and D. Kulić, "Lower body kinematics estimation from wearable sensors for walking and running: A deep learning approach," *Gait & Posture*, vol. 83, pp. 185–193, 2021.

[35] I. Wouters, T. Almonroeder, B. DeJarlais, A. Laack, J. D. Willson, and T. W. Kernozek, "Effects of a movement training program on hip and knee joint frontal plane running mechanics," *International journal of sports physical therapy*, vol. 7, no. 6, p. 637, 2012.

[36] M. Adouni and A. Shirazi-Adl, "Partitioning of knee joint internal forces in gait is dictated by the knee adduction angle and not by the knee adduction moment," *Journal of biomechanics*, vol. 47, no. 7, pp. 1696–1703, 2014.

[37] M. Mundt, W. Thomsen, T. Witter, A. Koeppe, S. David, F. Bamer, W. Potthast, and B. Markert, "Prediction of lower limb joint angles and moments during gait using artificial neural networks," *Medical & biological engineering & computing*, vol. 58, no. 1, pp. 211–225, 2020.

[38] E. Rapp, S. Shin, W. Thomsen, R. Ferber, and E. Halilaj, "Estimation of kinematics from inertial measurement units using a combined deep learning and optimization framework," *Journal of Biomechanics*, vol. 116, p. 110229, 2021.

[39] J. H. Geissinger and A. T. Asbeck, "Motion inference using sparse inertial sensors, self-supervised learning, and a new dataset of unscripted human motion," *Sensors*, vol. 20, no. 21, p. 6330, 2020.

[40] S. Ghorbani, K. Mahdaviani, A. Thaler, K. Kording, D. J. Cook, G. Blohm, and N. F. Troje, "Movi: A large multi-purpose human motion and video dataset," *Plos one*, vol. 16, no. 6, p. e0253157, 2021.

[41] B. Wagstaff, V. Peretroukhin, and J. Kelly, "University of toronto foot-mounted inertial navigation dataset," 2021. [Online]. Available: https://dx.doi.org/10.21227/v1z6-9z84

[42] M. Trumble, A. Gilbert, C. Malleson, A. Hilton, and J. Collomosse, "Total capture: 3d human pose estimation fusing video and inertial sensors," in *Proceedings of 28th British Machine Vision Conference*, 2017, pp. 1–13.

[43] F. Niemann, C. Reining, F. Moya Rueda, N. R. Nair, J. A. Steffens, G. A. Fink, and M. Ten Hompel, "Lara: Creating a dataset for human activity recognition in logistics using semantic attributes," *Sensors*, vol. 20, no. 15, p. 4083, 2020.

[44] J. Camargo, A. Ramanathan, W. Flanagan, and A. Young, "A comprehensive, open-source dataset of lower limb biomechanics in multiple conditions of stairs, ramps, and level-ground ambulation and transitions," *Journal of Biomechanics*, vol. 119, p. 110320, 2021.

[45] D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. Förster, G. Tröster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha *et al.*, "Collecting complex activity datasets in highly rich networked sensor environments," in *2010 Seventh*

international conference on networked sensing systems (INSS). IEEE, 2010, pp. 233–240.

[46] T. Marcard, G. Pons-Moll, and B. Rosenhahn, "Multimodal motion capture dataset tnt15," *Leibniz Univ. Hannover, Hanover, Germany, and Max Planck for Intelligent Systems, Tübingen, Germany, Tech. Rep*, 2016.

[47] H. Loose and J. L. Bolmgren, "Gaitanalysisdatabase–short overview," *Tech. Hochsch. Brandenbg*, pp. 1–6, 2019.

[48] Y. Luo, S. M. Coppola, P. C. Dixon, S. Li, J. T. Dennerlein, and B. Hu, "A database of human gait performance on irregular and uneven surfaces collected by wearable sensors," *Scientific data*, vol. 7, no. 1, pp. 1–9, 2020.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[50] F. J. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.

[51] X. Y. Lim, K. B. Gan, and N. A. Abd Aziz, "Deep convlstm network with dataset resampling for upper body activity recognition using minimal number of imu sensors," *Applied Sciences*, vol. 11, no. 8, p. 3543, 2021.

[52] M. Lv, W. Xu, and T. Chen, "A hybrid deep convolutional and recurrent neural network for complex activity recognition using multimodal sensors," *Neurocomputing*, vol. 362, pp. 33–40, 2019.

[53] S. S. Saha, S. S. Sandha, and M. Srivastava, "Deep convolutional bidirectional lstm for complex activity recognition with missing data," in *Human Activity Recognition Challenge*. Springer, 2021, pp. 39–53.

[54] J. Giorgi, O. Nitski, B. Wang, and G. Bader, "Declutr: Deep contrastive learning for unsupervised textual representations," *arXiv preprint arXiv:2006.03659*, 2020.

[55] B. Dai and D. Lin, "Contrastive learning for image captioning," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[56] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[57] A. Spurr, A. Dahiya, X. Wang, X. Zhang, and O. Hilliges, "Self-supervised 3d hand pose estimation from monocular rgb via contrastive learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 230–11 239.

[58] P. Kumar and S. Chauhan, "Study on temperature $(\tau)(\tau)$ variation for simclr-based activity recognition," *Signal, Image and Video Processing*, pp. 1–6, 2022.

[59] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5745–5753.

[60] M. Mundt, A. Koeppe, S. David, T. Witter, F. Bamer, W. Potthast, and B. Markert, "Estimation of gait mechanics based on simulated and measured

imu data using an artificial neural network," *Frontiers in bioengineering and biotechnology*, vol. 8, p. 41, 2020.

[61] S. T. Osis, B. A. Hettinga, S. Macdonald, and R. Ferber, "Effects of simulated marker placement deviations on running kinematics and evaluation of a morphometric-based placement feedback method," *PloS one*, vol. 11, no. 1, p. e0147111, 2016.

[62] S. P. Nair, S. Gibbs, G. Arnold, R. Abboud, and W. Wang, "A method to calculate the centre of the ankle joint: A comparison with the vicon® plug-in-gait model," *Clinical Biomechanics*, vol. 25, no. 6, pp. 582–587, 2010.