Predicting Lower Extremity Joint Kinematics Using Multi-Modal Data in the Lab and Outdoor Environment

Md Sanzid Bin Hossain
University of Central Florida
md543636@ucf.edu

Zhishan Guo
NC State University
zguo32@ncsu.edu

Ning Sui
NC State University
nsui@ncsu.edu

Hwan Choi
University of Central Florida
hwan.choi@ucf.edu

Abstract

Predicting future walking joint kinematics is crucial for assistive device control, especially in variable walking environments. Traditional optical motion capture systems provide kinematics data but require laborious post-processing, whereas IMU based systems provide direct calculations but add delays due to data collection and algorithmic processes. Predicting future kinematics helps to compensate for these delays, enabling the system real-time. Furthermore, these predicted kinematics could serve as target trajectories for assistive devices such as exoskeletal robots and lower limb prostheses. However, given the complexity of human mobility and environmental factors, this prediction remains to be challenging. To address this challenge, we propose the Dual-ED-Attention-FAM-Net, a deep learning model utilizing two encoders, two decoders, a temporal attention module, and a feature attention module. Our model outperforms the state-of-the-art LSTM model. Specifically, for Dataset A, using IMUs and a combination of IMUs and videos, RMSE values decrease from 4.45° to 4.22° and from 4.52° to 4.15°, respectively. For Dataset B, IMUs and IMUs combined with pressure insoles result in RMSE reductions from 7.09° to 6.66° and from 7.20° to 6.77°, respectively. Additionally, incorporating other modalities alongside IMUs helps improve the performance of the model.

Keywords: Future Kinematics Prediction, Deep learning, Multi-modal Fusion, Wearable Sensors

1. Introduction

Gait motion analysis is a quantitative method for assessing a person's kinesiological health. It can assist in determining the severity, prognosis, and course of an illness or injury. Additionally, it is possible to infer appropriate lower limb assistive device control by predicting the motion of a person's gait based on previous movement data. Therefore, the ability to evaluate gait motion facilitates patient monitoring, hastens rehabilitation, and also allows for prosthesis control.

In traditional measuring techniques for gait analysis, trajectories of reflective markers are measured by 3D infrared motion capture cameras. To compute joint angles, the obtained data is then processed using dynamic analysis software (OpenSimDelp et al., 2007, Visual3D (C-Motion, USA), Vicon Nexus (Oxford, UK)]. Despite the fact that this method is regarded as the gold standard for measuring the dynamic function of human motion (gait), it requires manual marker trajectory processing with expertise, and the data collection is limited to a specific setup. Thus, this method limits the application where kinematics has to be estimated or predicted outside the lab environment in real-time. To address these issues, a single in-depth camera or two or more conventional video cameras detect joint centers with an algorithm (Z. Zhang, 2012, Cao et al., 2021) to measure joint motions. However, they still require a specific capture volume, and since the level of accuracy is heavily influenced by the type of fabric and the participant's BMI, the outcomes are not as accurate as a 3D infrared motion capture camera.

Some of the drawbacks of the current approaches for evaluating the dynamic function of human motion may be solved by the use of Inertial Measurement Unit (IMU) sensors. Researchers can measure human movements outside of the laboratory by substituting wearable IMU sensors for conventional motion capture cameras. Potential long-term use and integrated daily monitoring applications are also made possible by these techniques. Although IMU-based motion estimation systems such as Xsens (Schepers et al., 2018) are commercially available, kinematics computation from IMU can lead to time delay due to data collection and algorithm process (X. Zhang et al., 2023), limiting use for assistive device control. As a result, a deep learning approach can be a viable option to anticipate future joint kinematics, which can compensate for the delay for real-time assistive device control.

It is essential to anticipate future motion when controlling the lower limb assistive devices. For example, the majority of prosthetic device controls consider three-level hierarchical architecture (Tucker

URI: https://hdl.handle.net/10125/106770
978-0-9981331-7-1

et al., 2015), with the highest level being activity mode such as stair, flat, or ground, the mid-level being a controller, which progresses through a gait phase within a mode, and the low level being a controller to supply torque or position information to the hardware for achieving desired state. Numerous studies are currently attempting to achieve continuous control by translating predicted kinematics to appropriate prosthesis control, without the use of separate controllers for various gait phases (Eslamy et al., 2020) or activities (Rai and Rombokas, 2019).

Currently, there are some ongoing works for the future prediction of joint kinematics. In (Sharma and Rombokas, 2022), egocentric vision data and joint angle data from IMUs are used to predict future joint angles using the Long Short Term Memory (LSTM) based model. In (Rai and Rombokas, 2019), joint angle data from Xsens IMU-based motion data is used to predict ankle joint angle using the LSTM layer-based deep learning model. In (Sharma and Rombokas, 2022), an out-of-the-lab dataset is used to train the model, while in (Rai and Rombokas, 2019), Comprehensive High-Level Activity Mobility Predictor (CHAMP) and an obstacle course are introduced in the trial. Although both of the approaches showed promising directions for kinematics prediction in highly variable gait patterns, the use of a generic LSTM-based deep learning model may limit the prediction performance. Moreover, their algorithm is validated only on a single dataset, how their approach will perform on a new dataset with different walking conditions, remains unknown. To address these limitations, we are proposing a novel deep-learning model Dual-ED-Attention-FAM-Net to predict future joint angles in two different datasets, and our proposed model is performing better than LSTM or other generic approaches for both datasets.

Contribution. To predict future kinematics of sagittal plane hip, knee, and ankle angle for both legs in lab environments and outside of the lab, this study proposes a novel encoder-decoder-based deep learning model Dual-Encoder Decoder-attention-Feature Attention Module-Net (Dual-ED-Attention-FAM-Net). We use two datasets (Dataset A (self-collected), and Dataset B (Losing and Hasenjäger, 2022)) to implement the model. Dataset A contains different activities in the lab environment, while Dataset B contains walking trials outside the lab in three separate walking courses. We use additional modalities such as video or pressure data from insoles in addition to IMUs to improve future joint kinematics prediction. The main contributions of this study are: (i) We propose a novel deep-learning model Dual-ED-Attention-FAM-Net to predict future joint kinematics; In Dual-ED-Attention-FAM-Net, we utilize both Bi-directional LSTM and Gated Recurrent Unit (GRU) layer to enhance the performance of the model. We also add a temporal attention module to improve the prediction. Finally, we propose a feature attention module to weigh the input of decoder architecture to further improve the kinematics prediction performance; (ii) we demonstrate the better generalization capability of our model for both datasets; (iii) we propose multi-modal data fusion to improve the prediction performance compared to using only IMUs; (iv) we present comprehensive model ablation to prove that our model outperforms generic deep learning models for both datasets.

2. Proposed Approach

In this section, we will outline the problem statement, present our proposed model, and provide a description of other models which will be used for performance comparison.

2.1. Problem Statement

This paper predicts 0.1 seconds of sagittal plane hip, knee, and ankle joint angles of the left and right leg in different walking scenarios encountered in daily living. To implement the algorithm, we utilize two different datasets (Dataset A (self-collected), and Dataset B (Losing and Hasenjäger (2022)). As the datasets are collected with different sensor configurations and modalities, we use the available modalities for the respective dataset to train the model.

For Dataset A, we use IMUs and processed features from video data to predict joint kinematics whereas in Dataset B, we use IMU and pressure data from footwear pressure insoles. Specifically, when we have accelerometer and gyroscope data of $\mathbb{I}^{acc}_{\Delta T^{acc}_{imu}}$=[$I^{acc}_{1}$, ... ,$I^{acc}_{\Delta T^{acc}_{imu}}$] $\in \mathbb{R}^{\Delta T^{acc}_{imu} \times D^{acc}_{imu} \cdot M^{acc}}$ and $\mathbb{I}^{gyr}_{\Delta T_{imu}}$=[$I^{gyr}_{1}$, ... ,$I^{gyr}_{\Delta T^{gyr}_{imu}}$] $\in \mathbb{R}^{\Delta T^{gyr}_{imu} \times D^{gyr}_{imu} \cdot M^{gyr}}$ for a specific window length of ΔT^{acc}_{imu}, ΔT^{gyr}_{imu}, and processed features from other modalities (video features/insole pressure data) $\mathbb{O}^{vid/ins}_{\Delta T^{vid/ins}_{other}}$=[$O^{vid/ins}_{1}$,.....,$O^{vid/ins}_{\Delta T^{vid/ins}_{other}}$] $\in \mathbb{R}^{\Delta T^{vid/ins}_{other} \times D^{vid/ins}_{other} \cdot N}$ for a window length $\Delta T^{vid/ins}_{other}$, then joint kinematics prediction for window length of $\Delta T_{prediction}$, $\mathbb{K}_{\Delta T_{prediction}}$=[$K_1$,.....,$K_{\Delta T_{prediction}}$] $\in \mathbb{R}^{\Delta T_{prediction} \times D_K}$ can be realized by Equation 1.

$$(\mathbb{I}^{acc}_{\Delta T^{acc}_{imu}}, \mathbb{I}^{gyr}_{\Delta T^{gyr}_{imu}}, \mathbb{O}^{vid/ins}_{\Delta T^{vid/ins}_{other}}) \rightarrow \mathbb{K}_{\Delta T_{prediction}} \quad (1)$$

Here, D^{acc}_{imu} and D^{gyr}_{imu} represent three-axis accelerometer and gyroscope data, $O^{vid/ins}_{other}$ represents

video/foot pressure features for each leg, M^{acc} and M^{gyr} is the number of IMU sensors, and N is the total number of the video camera/pressure sensor insoles. For Dataset A, ΔT^{acc}_{imu} = ΔT^{gyr}_{imu} = 45, $\Delta T^{vid/ins}_{other}$ = 15, $\Delta T_{prediction}$ = 9, D^{acc}_{imu} = D^{gyr}_{imu} = 3, $D^{vid/ins}_{other}$ = 18, M^{acc} = M^{gyr} = 8, and N = 2. For Dataset B, ΔT^{acc}_{imu}=ΔT^{gyr}_{imu} = 30, $\Delta T^{vid/ins}_{other}$ = 30, $\Delta T_{prediction}$ = 6, D^{acc}_{imu}=D^{gyr}_{imu} = 3, $D^{vid/ins}_{other}$ = 25, M^{acc} = M^{gyr} = 17, and N = 2.

2.2. Proposed Model

Our proposed Dual-ED-Attention-FAM-Net (Figure 1) builds with Bi-LSTM and Bi-GRU layer-based dual encoders and dual decoders, a Temporal Attention Module (TAM), a Feature Attention Module (FAM), and a Fully Connected (FC) layer. Each component is described in the following subsection.

2.2.1. Bi-LSTM Encoder: In the Bi-LSTM encoder, two bi-directional LSTM layers are stacked together to encode multiple input modalities separately. A dropout layer is added after each Bi-LSTM layer to tackle the problem of overfitting. A batch normalization layer is applied to the input of each modality separately to perform an operation similar to the standard normalization of the input (Li et al., 2020). The cell and hidden state of the three encoders are concatenated together to get a single vector of the cell and hidden state. These cells and hidden state vectors are passed to the Bi-LSTM layer of the decoder stage. If the batch normalized features from accelerometer and gyroscope of IMU are $\mathbb{I}^{acc}_{\Delta T^{acc}_{imu}} \in \mathbb{R}^{\Delta T^{acc}_{imu} \times D^{acc}_{imu} \cdot M^{acc}}$, $\mathbb{I}^{gyr}_{\Delta T^{gyr}_{imu}} \in \mathbb{R}^{\Delta T^{gyr}_{imu} \times D^{gyr}_{imu} \cdot M^{gyr}}$, and other modality (video features/insole pressure data) is $\mathrm{O}^{vid/ins}_{\Delta T^{vid/ins}_{other}} \in \mathbb{R}^{\Delta T^{vid/ins}_{other} \times D^{vid/ins}_{other} \cdot N}$ respectively, then output from the Bi-LSTM encoder can be derived using Equation 2, 3, and 4.

$$X^{acc,bi-lstm}_{\Delta T^{acc}_{imu}}, (c^{acc,bi-lstm}_{T}, h^{acc,bi-lstm}_{T}) = Bi - LSTM(\mathbb{I}^{acc}_{\Delta T^{acc}_{imu}}) \quad (2)$$

$$X^{gyr,bi-lstm}_{\Delta T^{gyr}_{imu}}, (c^{gyr,bi-lstm}_{T}, h^{gyr,bi-lstm}_{T}) = Bi - LSTM(\mathbb{I}^{gyr}_{\Delta T^{gyr}_{imu}}) \quad (3)$$

$$X^{vid/ins,bi-lstm}_{\Delta T^{vid/ins}_{other}}, (c^{vid/ins,bi-lstm}_{T}, h^{vid/ins,bi-lstm}_{T}) = Bi - LSTM(\mathbb{O}^{vid/ins}_{\Delta T^{vid/ins}_{other}}) \quad (4)$$

where, $X^{acc,bi-lstm}_{\Delta T^{acc}_{imu}}$, $X^{gyr,bi-lstm}_{\Delta T^{gyr}_{imu}}$ are the encoded accelerometer and gyroscope features for a window length of ΔT^{acc}_{imu} and ΔT^{gyr}_{imu}, respectively. $X^{vid/ins,bi-lstm}_{\Delta T^{vid/ins}_{other}}$ is the encoded features of video and insole data for a window length of $\Delta T^{vid/ins}_{other}$. $c^{acc,bi-lstm}_{T}$, $c^{gyr,bi-lstm}_{T}$, and $c^{vid/ins,bi-lstm}_{T}$ are the final cell state and $h^{acc,bi-lstm}_{T}$, $h^{gyr,bi-lstm}_{T}$, and $h^{vid/ins,bi-lstm}_{T}$ are final hidden states from three encoders.

2.2.2. Bi-GRU Encoder: In the Bi-GRU encoder, two bidirectional GRU are stacked together and a dropout layer is used after each layer to avoid overfitting. Batch normalization is applied to each input modality separately before inputting to the encoder. The hidden state from each encoder is concatenated together to form a single tensor and used as the hidden state for the decoder Bi-GRU layer. Output features and hidden state from the Bi-GRU Encoder can be represented by the Equation 5, 6, and 7.

$$X^{acc,bi-gru}_{\Delta T^{acc}_{imu}}, h^{acc,bi-gru}_{T} = Bi - GRU(\mathbb{I}^{acc}_{\Delta T^{acc}_{imu}}) \quad (5)$$

$$X^{gyr,bi-gru}_{\Delta T^{gyr}_{imu}}, h^{gyr,bi-gru}_{T} = Bi - GRU(\mathbb{I}^{gyr}_{\Delta T^{gyr}_{imu}}) \quad (6)$$

$$X^{vid/ins,bi-gru}_{\Delta T^{video/insoels}_{other}}, h^{vid/ins,bi-gru}_{T} = Bi - GRU(\mathbb{O}^{vid/ins}_{\Delta T^{vid/ins}_{other}}) \quad (7)$$

where, $X^{acc,bi-gru}_{\Delta T^{acc}_{imu}}$, $X^{gyr,bi-gru}_{\Delta T^{gyr}_{imu}}$ are the encoded accelerometer and gyroscope features for a window length of ΔT^{acc}_{imu} and ΔT^{gyr}_{imu} respectively. $X^{vid/ins,bi-gru}_{\Delta T^{vid/ins}_{other}}$ is the encoder features of vid/ins data for a window length of $\Delta T^{vid/ins}_{other}$. $h^{acc,bi-gru}_{T}$, $h^{gyr,bi-gru}_{T}$, and $h^{vid/ins,bi-gru}_{T}$ are final hidden states from three encoders.

2.2.3. Bi-LSTM Bi-GRU Decoder: We utilize two decoders to predict future joint kinematics. Future joint kinematics prediction can be represented by the Equation 8.

$$Y_t = FC([X^{decoder,bi-lstm}_{t}, X^{decoder,bi-gru}_{t}]), \quad t = 1, 2, ..., \Delta T_{prediction} \quad (8)$$

where $X^{decoder,bi-lstm}_{t}$, $X^{decoder,bi-gru}_{t}$ are the output features from Bi-LSTM and Bi-GRU decoder respectively. The output features can be derived with Equation 9 and 10.

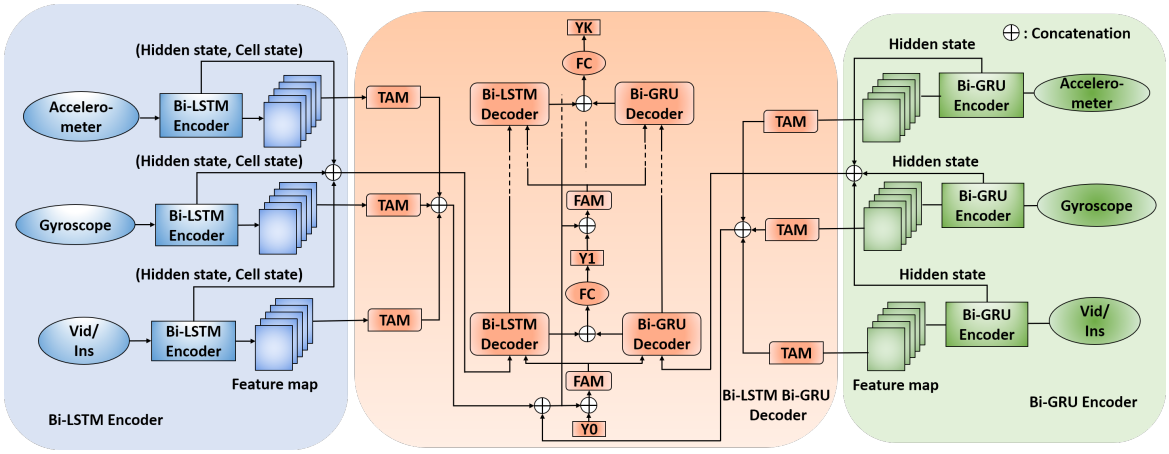

Figure 1. Entire structure of Dual-ED-Attention-FAM-Net. It has a Temporal Attention Module (TAM), Feature Attention Module (FAM), and Fully Connected (FC) layer. For Dataset A, we use video data from legs, where in Dataset B, we utilize pressure data from foot insoles.

$$X_t^{decoder,bi-lstm}, (c_t^{decoder,bi-lstm}, h_t^{decoder,bi-lstm}) = Bi-LSTM(\mathbb{C}_{weighted,t-1}^{features}, (c_{t-1}^{concat,bi-lstm}, h_{t-1}^{concat,bi-lstm})), t = 1, 2, ..., \Delta T_{prediction} \quad (9)$$

$$X_t^{decoder,bi-gru}, h_t^{decoder,bi-gru} = Bi-GRU(\mathbb{C}_{weighted,t-1}^{features}, h_{t-1}^{concat,bi-gru}), t = 1, 2, ..., \Delta T_{prediction} \quad (10)$$

where $\mathbb{C}_{weighted,t-1}^{features}$ is the weighted concatenated features of the TAM module and prediction of a single time step. Specifically, encoded output features from each modality are passed to TAM to generate a context vector separately. Context vectors from each of the modalities from Bi-LSTM and Bi-GRU encoders and the prediction from the single time step are then concatenated together to build initial features (Equation 11).

$$\mathbb{C}_{concat,t-1}^{features} = [context_{bi-lstm}^{concat}, context_{bi-gru}^{concat}, K_{t-1}], t = 1, 2, ..., \Delta T_{prediction} \quad (11)$$

These initially concatenated features are weighted by passing through FAM to create $\mathbb{C}_{weighted,t-1}^{features}$. Then the weighted features are passed to both Bi-LSTM and Bi-GRU decoders. To get the prediction of the first time step, we initialize $K_0 = 0$.

In Equation 12, $c_t^{decoder,bi-lstm}, h_t^{decoder,bi-lstm}$ is current cell and $c_{t-1}^{decoder,bi-lstm}, h_{t-1}^{decoder,bi-lstm}$ is previous cell. Both cells are hidden state of Bi-LSTM decoder. On the other hand, in Equation 13, $h_t^{decoder,bi-gru}$ and $h_{t-1}^{decoder,bi-gru}$ are the current and previous hidden state of Bi-GRU decoder. For the first time step, concatenated cell state, hidden state from multiple Bi-LSTM encoders, and concatenated hidden state from multiple Bi-GRU encoders are used as the initial cell, hidden state for Bi-LSTM decoder and initial hidden state for Bi-GRU decoder, respectively. We can derive the initial cell, the hidden state of the Bi-LSTM decoder, and the hidden state of the Bi-GRU decoder utilizing the following equations.

$$(c_T^{concat,bi-lstm}, h_T^{concat,bi-lstm}) = [(c_T^{acc,bi-lstm}, h_T^{acc,bi-lstm}), (c_T^{gyr,bi-lstm}, h_T^{gyr,bi-lstm}), (c_T^{vid/ins,bi-lstm}, h_T^{vid/ins,bi-lstm})] \quad (12)$$

$$h_T^{concat,bi-gru} = [h_T^{acc,bi-gru}, h_T^{gyr,bi-gru}, h_T^{vid/ins,bi-gru}] \quad (13)$$

$$c_0^{concat,bi-lstm}, h_0^{concat,bi-lstm} = c_T^{concat,bi-lstm}, h_T^{concat,bi-lstm} \quad (14)$$

$$h_0^{concat,bi-gru} = h_T^{concat,bi-gru} \quad (15)$$

Temporal Attention Module (TAM): Output features of each modality from the encoders are passed through TAM separately to create separate context vectors. At first, output features are passed to two fully connected

layers. A hyperbolic tangent activation is used after the first fully connected layer. After these two fully connected layers, the softmax activation function is applied to calculate the attention score for each time step (Equation 16, 17, and 18).

$$Attn^{acc}_{bi-lstm} = Softmax(fc(X^{acc,bi-lstm}_{\Delta T^{acc}_{imu}})) \quad (16)$$

$$Attn^{gyr}_{bi-lstm} = Softmax(fc(X^{gyr,bi-lstm}_{\Delta T^{gyr}_{imu}})) \quad (17)$$

$$Attn^{vid/ins}_{bi-lstm} = Softmax(fc(X^{vid/ins,bi-lstm}_{\Delta T^{vid/ins}_{other}})) \quad (18)$$

Weighted summation is taken over the sequence length to generate a context vector, which is later used as the features of input of the decoder (Equation 19, 20, and 21).

$$context^{acc}_{bi-lstm} = \sum_{i=1}^{\Delta T^{acc}_{imu}} Attn^{acc}_{bi-lstm,i} \odot X^{acc,bi-lstm}_{i} \quad (19)$$

$$context^{gyr}_{bi-lstm} = \sum_{i=1}^{\Delta T^{gyr}_{imu}} Attn^{gyr}_{bi-lstm,i} \odot X^{gyr,bi-lstm}_{i} \quad (20)$$

$$context^{vid/ins}_{bi-lstm} = \sum_{i=1}^{\Delta T^{vid/ins}_{other}} Attn^{vid/ins}_{bi-lstm,i} \odot X^{vid/ins,bi-lstm}_{i} \quad (21)$$

Contexts from all the modalities are concatenated together to build a single tensor and passed to the input of the decoder (Equation 22).

$$context^{concat}_{bi-lstm} = [context^{acc}_{bi-lstm}, context^{gyr}_{bi-lstm}, context^{vid/ins}_{bi-lstm}] \quad (22)$$

Similarly, we can apply the same method to Bi-GRU Encoder to obtain the overall context vector from the encoder and later use it as the input features for the decoder (Equation 23).

$$context^{concat}_{bi-gru} = [context^{acc}_{bi-gru}, context^{gyr}_{bi-gru}, context^{vid/ins}_{bi-gru}] \quad (23)$$

Feature Attention Module (FAM): Concatenated features of TAM modules for Bi-LSTM and Bi-GRU encoders are concatenated with the kinematics prediction of a single time step to create input features for Bi-LSTM and Bi-GRU decoders. Concatenated features are passed to the FAM module to put weight on each features ensuring its proper importance. A fully connected layer with a sigmoid function is used to calculate attention weights. The calculated weights are multiplied with the concatenated features to produce attention-weighted features. If concatenated features from TAM modules and the prediction layer are $\mathbb{C}^{features}_{concat,t} \in \mathbb{R}^{1\times f}$ and the attention score vector is $\mathbb{W}^{features}_{t} \in [0,1]^{1\times f}$ where f is the total number of concatenated features, then attention weighted features can be realized by Equation 24.

$$\mathbb{C}^{features}_{weighted,t} = \mathbb{W}^{features}_{t} \odot \mathbb{C}^{features}_{concat,t} \quad (24)$$

$\mathbb{C}^{features}_{weighted,t}$ is used as the input for both the Bi-LSTM and Bi-GRU layers of the decoder.

2.3. Models for Comparison

We categorize all of our models into three groups for comparison: the state-of-the-art model, other baseline models, and intermediate models. The state-of-the-art model has been previously employed in related studies, other baseline models are simple models similar to the state-of-the-art ones, and the intermediate models serve as the basis for our final proposed model.

2.3.1. State-of-the-art Model

• **LSTM-Net (Sharma and Rombokas, 2022, Rai and Rombokas, 2019):** Two LSTM layers are stacked together to build an encoder, which features from the accelerometer, gyroscope, and video/pressure insoles data separately. A dropout layer is added after each LSTM layer to avoid overfitting. Then, features from each of the encoders are concatenated together and connected to a fully connected layer with linear activation to predict future kinematics.

2.3.2. Other Baseline Models

• **GRU-Net:** Two GRU layers are stacked together to extract features from three input modalities and concatenated features from these modalities are used for future kinematics prediction.

• **Bi-LSTM-Net:** LSTM layer of LSTM-Net is replaced with bi-directional LSTM to create Bi-LSTM-Net.

• **Bi-GRU-Net:** GRU layer of GRU-Net is replaced with bi-directional GRU to create Bi-GRU-Net.

2.3.3. Intermediate Models

• **LSTM-LSTM-ED-Net:** We use Encoder-Decoder (ED) architecture in LSTM-LSTM-ED-Net to predict future joint kinematics. Two LSTM layers are stacked together to build the encoder. A dropout layer is used

after each LSTM layer. The cell and hidden state from the encoder are used as the initial cell and hidden state of the decoder. Six joint angles (sagittal plane hip, knee, and ankle angle for both legs) initialized to zero are fed as the input to the decoder for the first time step. Then, the prediction of the LSTM layer from the previous step is used as the input to the current LSTM layer. The cell state and hidden state of the decoder from the previous time step are passed to the current time step. To prevent overfitting, a dropout layer is used after the LSTM layer.

• **GRU-GRU-ED-Net:** In GRU-GRU-ED-Net, we replace the LSTM layer of LSTM-LSTM-ED-Net with the GRU layer and only the hidden state is passed to the next step as no cell state is available for the GRU layer.

• **Bi-LSTM-Bi-LSTM-ED-Net:** We replace LSTM layer of LSTM-LSTM-ED-Net with bi-directional LSTM to create Bi-LSTM-Bi-LSTM-ED-Net.

• **Bi-GRU-Bi-GRU-ED-Net:** GRU layer of GRU-GRU-ED-Net is replaced with bi-directional GRU to create Bi-GRU-Bi-GRU-ED-Net.

• **Bi-LSTM-Bi-LSTM-ED-Attention-Net:** We adapt Dual-ED-Attention-Net to Bi-LSTM-Bi-LSTM-ED-Attention-Net by removing FAM and the Bi-GRU component from the ED.

• **Bi-GRU-Bi-GRU-ED-Attention-Net:** We remove FAM and Bi-LSTM components from ED from Dual-ED-Attention-Net to create Bi-GRU-Bi-GRU-ED-Attention-Net.

• **Dual-ED-Attention-Net:** We remove FAM from Dual-ED-Attention-FAM-Net to create Dual-ED-Attention-Net.

Table 1. Hyperparameters of the different models. Values in the first parenthesis is for Dataset B

Model	Encoder	Decoder	FC
LSTM-Net	Unit:128, 64, Drop.: 0.0, 0.0	NA	Out_feat: 54 (36)
GRU-Net	Unit:128, 64, Drop.: 0.0, 0.0	NA	Out_feat: 54 (36)
Bi-LSTM-Net	Unit:128, 64, Drop.: 0.0, 0.0	NA	Out_feat: 54 (36)
Bi-GRU-Net	Unit:128, 64, Drop.: 0.0, 0.0	NA	Out_feat: 54 (36)
LSTM-LSTM-ED-Net	Unit:128, 64, Drop.: 0.0, 0.0	Unit: 192, Drop.: 0.0	Out_feat: 6 (6)
GRU-GRU-ED-Net	Unit:128, 64, Drop.: 0.0, 0.0	Unit: 192, Drop.: 0.0	Out_feat: 6 (6)
Bi-LSTM-Bi-LSTM-ED-Net	Unit:128, 64, Drop.: 0.0, 0.0	Unit: 192, Drop.: 0.05	Out_feat: 6 (6)
Bi-GRU-Bi-GRU-ED-Net	Unit:128, 64, Drop.: 0.0, 0.0	Unit: 192, Drop.: 0.05	Out_feat: 6 (6)
Bi-LSTM-Bi-LSTM-ED-Attention-Net	Unit:128, 64, Drop.: 0.05, 0.05	Unit: 128, Drop.: 0.05	Out_feat: 6 (6)
Bi-GRU-Bi-GRU-ED-Attention-Net	Unit:128, 64, Drop.: 0.0, 0.0	Unit: 128, Drop.: 0.05	Out_feat: 6 (6)
Dual-ED-Attention-Net	Unit:128, 64, Drop.: 0.05, 0.05	Unit: 128, Drop.: 0.05	Out_feat: 6 (6)
	Unit:128, 64, Drop.: 0.05, 0.05	Unit: 128, Drop.: 0.05	Out_feat: 6 (6)
Dual-ED-Attention-FAM-Net	Unit:128, 64, Drop.: 0.05, 0.05	Unit: 128, Drop.: 0.05	Out_feat: 6 (6)
	Unit:128, 64, Drop.: 0.05, 0.05	Unit: 128, Drop.: 0.05	Out_feat: 6 (6)

3. Experiment

3.1. Dataset description

In this paper, we utilize two different datasets. A brief description of the datasets is discussed in this subsection.

Dataset A: Ten subjects (6 males, and 4 females, age: 23.9±2.91 years, height: 1.65±0.06 m, weight: 63.41±6.81 kg) are collected where participants wear 8 IMUs sensors (Avanti wireless EMG/IMU, Delsys, Boston, MA) on the whole body and a single GoPro camera (Hero7 black, GoPro Inc., San Mateo, California) in the middle of the tibia on the longitudinal direction of each leg. All participants completed a total of 14 trials. Each participant walks on the treadmill and overground with slow, normal, fast, and very fast walking speeds. Two trials each on stairs and slope (up and down) conditions were also performed. Furthermore, two more trials, where participants walked in a round path, and a path with two obstacles for avoidance were also collected. Joint kinematics and IMU data are collected with a sampling rate of 100 Hz and ˜148 Hz respectively. Video is collected at 30 fps. Histograms of optical flow features are calculated from these wearable video data. To synchronize more efficiently, the IMU and joint kinematics data are down-sampled to 90 Hz so that one sample from the histogram of optical flow features will be equivalent to three samples of IMU and joint angle. We use 0.50 seconds of input data, consisting of 45 IMU samples and 15 video feature samples, to forecast the subsequent kinematics of 0.1 seconds, equivalent to 9 samples. We then shift this input window by 9 samples, obtaining input features from samples 10 to 54, and predict future kinematics for the subsequent 0.1 seconds from samples 55 to 63. We apply the same technique to pre-process the whole dataset.

Dataset B (Losing and Hasenjäger (2022)): Twenty healthy participants (5 females, 15 males, age: 18-69 years, height: 178.5±7.64 cm, weight: 72.9±8.7 kg) wore 17 IMU sensors of Xsens motion capture suit (Schepers et al., 2018) consisting of the MVN-Link BIOMECH full-body system and the MVN Link lycra suit, insoles with eight pressure sensing cells per foot. Participants completed three different walking courses, which includes different common walking environments such as level areas, stairs, flat ramps, steep ramps, etc. that are encountered in daily life. Pressure, normalized pressure, and raw pressure value of eight force sensors for each leg are used as the input features of the deep learning model. All the modalities in this dataset are

Table 2. Mean and standard deviation of RMSE and PCC for all the joint angles for all subjects for Dataset A when different modality is used as the input to the deep learning model.

Model	Modality-A (Acc-Gyr)		Modality-B (Acc-Gyr-Videos)	
	RMSE (°)	PCC	RMSE (°)	PCC
LSTM-Net	4.45 ± 0.46	0.964 ± 0.005	4.52 ± 0.40	0.963 ± 0.004
GRU-Net	4.38 ± 0.45	0.965 ± 0.003	4.34 ± 0.40	0.966 ± 0.004
Bi-LSTM-Net	4.33 ± 0.40	0.967 ± 0.004	4.31 ± 0.44	0.967 ± 0.005
Bi-GRU -Net	4.34 ± 0.50	0.967 ± 0.005	4.33 ± 0.46	0.967 ± 0.004
LSTM-LSTM-ED-Net	4.56 ± 0.44	0.964 ± 0.004	4.45 ± 0.43	0.964 ± 0.004
GRU-GRU-ED-Net	4.36 ± 0.51	0.965 ± 0.004	4.37 ± 0.50	0.966 ± 0.004
Bi-LSTM-Bi-LSTM-ED-Net	4.35 ± 0.50	0.967 ± 0.005	4.29 ± 0.48	0.968 ± 0.004
Bi-GRU-Bi-GRU-ED-Net	4.26 ± 0.47	0.968 ± 0.004	4.19 ± 0.50	0.969 ± 0.004
Bi-LSTM-Bi-LSTM-ED-Attention-Net	4.32 ± 0.48	0.967 ± 0.005	4.22 ± 0.48	0.968 ± 0.004
Bi-GRU-Bi-GRU-ED-Attention-Net	4.22 ± 0.48	0.968 ± 0.004	4.17 ± 0.49	0.969 ± 0.004
Dual-ED-Attention-Net	4.27 ± 0.54	0.969 ± 0.004	4.18 ± 0.47	0.969 ± 0.004
Dual-ED-Attention-FAM-Net	**4.22 ± 0.46**	**0.969 ± 0.004**	**4.15 ± 0.47**	**0.969 ± 0.003**

Table 3. Mean and standard deviation of RMSE and PCC for all the joint angles for all subjects for Dataset B when different modality is used as the input to the deep learning model.

Model	Modality-A (Acc-Gyr)		Modality-B (Acc-Gyr-Insoles)	
	RMSE (°)	PCC	RMSE (°)	PCC
LSTM-Net	7.09 ± 1.88	0.931 ± 0.022	7.20 ± 1.72	0.933 ± 0.016
GRU-Net	7.16 ± 1.84	0.928 ± 0.022	7.28 ± 1.67	0.929 ± 0.016
Bi-LSTM-Net	6.85 ± 1.83	0.936 ± 0.020	7.06 ± 1.75	0.939 ± 0.016
Bi-GRU-Net	6.91 ± 1.85	0.933 ± 0.023	7.20 ± 1.83	0.936 ± 0.017
LSTM-LSTM-ED-Net	7.13 ± 1.90	0.929 ± 0.023	7.12 ± 1.74	0.938 ± 0.015
GRU-GRU-ED-Net	7.12 ± 1.87	0.928 ± 0.021	7.09 ± 1.77	0.937 ± 0.015
Bi-LSTM-Bi-LSTM-ED-Net	6.81 ± 1.78	0.936 ± 0.021	6.91 ± 1.70	0.941 ± 0.015
Bi-GRU-Bi-GRU-ED-Net	6.93 ± 1.83	0.934 ± 0.021	7.00 ± 1.76	0.941 ± 0.015
Bi-LSTM-Bi-LSTM-ED-Attention-Net	6.81 ± 1.86	0.937 ± 0.021	6.86 ± 1.68	0.943 ± 0.014
Bi-GRU-Bi-GRU-ED-Attention-Net	6.91 ± 1.82	0.934 ± 0.022	6.99 ± 1.65	0.941 ± 0.016
Dual-ED-Attention-Net	6.73 ± 1.84	0.940 ± 0.020	6.88 ± 1.69	0.945 ± 0.015
Dual-ED-Attention-FAM-Net	**6.66 ± 1.80**	**0.940 ± 0.021**	**6.77 ± 1.71**	**0.946 ± 0.015**

synchronized with a sampling frequency of 60 Hz. More detailed information on the dataset can be found in (Losing and Hasenjäger, 2022). We utilize 0.50 seconds/30 samples of IMU and pressure sensor data as input and forecast subsequent 0.1 seconds/6 samples of kinematics.

3.2. Implementation Details

We train all of our models in Pytorch with a TITAN Xp GPU (NVIDIA, CA). The Dual-ED-Attention-FAM-Net is trained with a run time of approximately 25 minutes per subject for Dataset A and 51 minutes per subject for Dataset B. Adam (Kingma and Ba, 2014) is used as the optimizer. All the models are run for 12 epochs with a batch size of 64. We utilize Root Mean Square Error (RMSE) as a loss function to train all the models.

3.3. Evaluation Procedures

We perform a leave-one-subject-out cross-validation method, in which test subject data is excluded from the training set. Exclusion of the test subject data from training helps to tackle the problem of overfitting of the model and ensures the model's proper generalization capability. To measure the performance of the model, we utilize two performance metrics: RMSE and Pearson Correlation Coefficient (PCC). RMSE calculates the overall offset between the ground truth and the estimation from the model, while PCC ensures the correlation between these two.

4. Results

Model Ablation: In Table 2 and 3, we present RMSE and PCC values of different models when different modalities are used as input. Although we have modest improvement compared to intermediate models, larger improvements are achieved compared to state-of-the-art and other simple baseline models.

Dual Encoder Decoder: we utilize both Bi-LSTM and Bi-GRU layers to create dual encoder-decoder architecture, which helps to improve the performance compared to single encoder-decoder models (Table 2 and 3).

Feature Attention Module: We utilize a feature attention module to put weight on the input of the decoder, which helps to improve the performance in both datasets.

Temporal Attention Module: Although it makes a

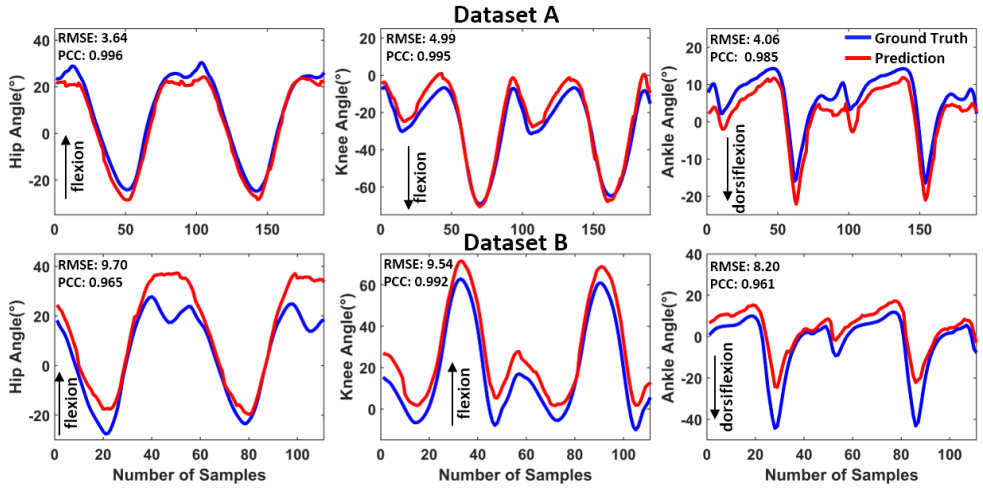

Figure 2. A sample plot of two gait cycles of hip, knee, and ankle joint angles for Dataset A and Dataset B. Knee angle is different due to different conventions of ground truth provided in the datasets.

small difference, the temporal attention module does not substantially improve performance.

Multi-modal effect: In Table 4 and 5, we provide the analysis when different modalities are used as the input to Dual-ED-Attention-FAM-Net for both datasets. We can see a performance improvement when additional modalities such as video features or pressure data from insoles are used as input. Although for Dataset A, both RMSE and PCC are improved, we see degraded performance for RMSE in Dataset B. However, we achieve higher PCC when insoles pressure data is used as an additional modality.

We provide a plot of two gait cycles of ground truth and prediction from our model in Figure 2 for qualitative demonstration.

Table 4. Mean and standard deviation of RMSE and PCC of hip, knee, and ankle joint angles for all the subjects for Dataset A when different modalities are used as input .

Joint Angle	Modality	RMSE (°)	PCC
Hip	Modality-A (Acc-Gyr)	4.60 ± 0.78	0.975 ± 0.008
	Modality-B (Acc-Gyr-Videos)	4.56 ± 0.81	0.977 ± 0.007
Knee	Modality-A (Acc-Gyr)	4.38 ± 0.45	0.986 ± 0.002
	Modality-B (Acc-Gyr-Videos)	4.30 ± 0.48	0.987 ± 0.002
Ankle	Modality-A (Acc-Gyr)	3.68 ± 0.50	0.945 ± 0.008
	Modality-B (Acc-Gyr-Videos)	3.58 ± 0.44	0.945 ± 0.008

5. Discussion

In this study, we propose a novel deep learning model Dual-ED-Attention-FAM-Net to predict future joint kinematics of the lower extremities across two different datasets. We also show that merging different data types from different measurement modalities can help to improve the prediction performance further compared to only IMU-based prediction. In addition to performance improvement, these modalities can be used for additional purposes. For example, in Dataset A, we utilize egocentric vision data from the leg. This vision data can be used for sensing the environment in the high-level controller of a prosthesis, where activity modes such as ground, flat, or stair, etc. are required(Tschiedel et al., 2020). On the other hand, foot pressure sensor data in Dataset B can be used to optimize a wearable assistive device (Jacobson et al., 2022). Moreover, our model can be modified with IMUs, cameras, and pressure sensor input modalities to capture all the aforementioned advantages of the respective sensor.

Table 5. Mean and standard deviation of RMSE and PCC of hip, knee, and ankle joint angles for all the subjects for Dataset B when different modalities are used as input.

Joint Angle	Modality	RMSE (°)	PCC
Hip	Modality-A (Acc-Gyr)	6.36 ± 2.50	0.968 ± 0.015
	Modality-B (Acc-Gyr-Insoles)	6.94 ± 2.67	0.967 ± 0.021
Knee	Modality-A (Acc-Gyr)	7.01 ± 2.03	0.970 ± 0.014
	Modality-B (Acc-Gyr-Insoles)	7.00 ± 2.04	0.973 ± 0.010
Ankle	Modality-A (Acc-Gyr)	6.60 ± 3.37	0.882 ± 0.047
	Modality-B (Acc-Gyr-Insoles)	6.38 ± 3.07	0.898 ± 0.031

Our proposed model Dual-ED-Attention-FAM-Net outperforms all the models for comparison. We

carefully design Dual-ED-Attention-FAM-Net with different modules. We add temporal attention to assign weights to each time step to put more emphasis on critical temporal features for the prediction. This strategy helps to improve the results as additional contexts are given to the input of decoders in addition to the kinematics data from the previous time step. We apply FAM to assign weight to the input features of the decoder. This helps to put more weight on the important features and further improves the predictive performance of the model. In table 2, 5, Bi-LSTM-Bi-LSTM-ED-Attention-Net provides better performance than Bi-GRU-Bi-GRU-ED-Attention-Net in Dataset B, while the opposite performance is seen for Dataset A. This causes generalization problems across different datasets. To Tackle this problem, we include both Bi-LSTM and Bi-GRU layers to construct our encoder and decoder, which in turn helps to improve the overall performance for both datasets.

We test our model on a GeForce GTX 1080 Ti GPU, and it takes approximately 0.2 milliseconds to predict 0.1 seconds of kinematics into the future using our proposed model. While data transmission and processing may add some extra time, it is generally feasible to handle this time delay with a GPU-based approach. However, if other low-cost device is used in the assistive device, which lacks the high computing power of a GPU, potentially causes delays in prediction. In the future, it would be worth validating our algorithm with larger time windows than 0.1 sec for compensating the delay caused by the prediction of low cost device.

From Table 2 and 3, we find that Dataset A is providing more accurate results compared to Dataset B. As Dataset A is collected in a lab environment with strictly classified walking conditions, it is easier to predict the kinematics as each walking condition is repeatable. Dataset B consists of walking trials from the outdoors, which is substantially more complex compared to walking in a lab environment. Transitions from one walking condition to another, such as from level-ground to a stair or from a stair to level-ground, require participants to adapt to various walking techniques for smooth transitions. These multiple factors contribute to less performance of future kinematics prediction in Dataset B. Although using additional input features from video in Dataset A helps to improve the overall performance of all joint angles, adding pressure data from the insoles does not reduce RMSE and increase PCC for all the joint angles. Specifically, RMSE of hip flexion angle increases when pressure insoles data is used. Although for the knee angle, we achieve marginal improvement, there are noticeable improvements in the ankle angle. A possible reason behind this is that as pressure insoles data provides pressures of different regions of the feet, it provides some essential information about the ankle angle, which in return improves the prediction accuracy.

In Figure 2, it is evident that most of the error accumulates due to the shift of the prediction graph. This is due to the reason that test subject's neutral limb alignment may not well represent in the training data (Rapp et al., 2021). In future, we can apply passive pseudo-calibration (Rapp et al., 2021), which may fix the shift and make the performance better. Another limitation of our proposed study is the use of a large number of wearable sensors as input. The number of sensors should be reduced further for practical implementation and user comfort as implemented in these studies for human movement evaluation (Hossain, Dranetz, et al., 2022, Hossain, Choi, et al., 2022, Hossain, Guo, et al., 2022). Another limitation of our work is the use of wearable IMU sensor-based joint angle estimation as ground truth for Dataset B. Although optical motion capture is considered the gold-standard method for motion capture, IMU-based motion capture exhibits deviation from the optical motion-capture-based motion estimation system (Robert-Lachaine et al., 2020). This different ground truth in Dataset B may impact the prediction performance.

Our proposed model is made publicly available at (https://github.com/Md-Sanzid-Bin-Hossain/Kinematics-Prediction-Using-Dual-ED-Attention-FAM-Net). Researcher or clinicians utilize our model directly or modify it according to their protocol. For Dataset A, the users need to follow the protocol of this paper. However, as Dataset B is collected with commercially available Xsens and foot pressure sensors, the setup should be the same.

6. Conclusion

In this study, we describe a method to improve future kinematics prediction by incorporating additional modalities, such as video data or pressure sensor data from the feet. These additional modalities aid in providing more information for prostheses control. We introduce a novel encoder-decoder-based deep learning model that outperforms different baseline models for both datasets and demonstrates the generalizability of our model across diverse datasets with various modalities.

Acknowledgments

This work is partially supported by the National Science Foundation Awards FRR-2246671, 2246672, and startup funding from North Carolina State University.

References

Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., & Sheikh, Y. (2021). Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE transactions on pattern analysis and machine intelligence*, *43*(1), 172–186.

Delp, S. L., Anderson, F. C., Arnold, A. S., Loan, P., Habib, A., John, C. T., Guendelman, E., & Thelen, D. G. (2007). Opensim: Open-source software to create and analyze dynamic simulations of movement. *IEEE Transactions on Biomedical Engineering*, *54*(11), 1940–1950. https : / / doi . org / 10 . 1109 / TBME.2007.901024

Eslamy, M., Oswald, F., & Schilling, A. F. (2020). Estimation of knee angles based on thigh motion: A functional approach and implications for high-level controlling of active prosthetic knees. *IEEE Control Systems Magazine*, *40*(3), 49–61.

Hossain, M. S. B., Choi, H., & Guo, Z. (2022). Estimating lower extremity joint angles during gait using reduced number of sensors count via deep learning. *Fourteenth International Conference on Digital Image Processing (ICDIP 2022)*, *12342*, 1116–1123.

Hossain, M. S. B., Dranetz, J., Choi, H., & Guo, Z. (2022). Deepbbwae-net: A cnn-rnn based deep superlearner for estimating lower extremity sagittal plane joint kinematics using shoe-mounted imu sensors in daily living. *IEEE Journal of Biomedical and Health Informatics*, *26*(8), 3906–3917.

Hossain, M. S. B., Guo, Z., & Choi, H. (2022). Estimation of hip, knee, and ankle joint moment using a single imu sensor on foot via deep learning. *2022 IEEE/ACM Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*, 25–33.

Jacobson, M., Kantharaju, P., Jeong, H., Ryu, J.-K., Park, J.-J., Chung, H.-J., & Kim, M. (2022). Foot contact forces can be used to personalize a wearable robot during human walking. *Scientific Reports*, *12*(1), 10947.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Li, F., Shirahama, K., Nisar, M. A., Huang, X., & Grzegorzek, M. (2020). Deep transfer learning for time series data based on sensor modality classification. *Sensors*, *20*(15), 4271.

Losing, V., & Hasenjäger, M. (2022). A multi-modal gait database of natural everyday-walk in an urban environment. *Scientific data*, *9*(1), 473.

Rai, V., & Rombokas, E. (2019). A framework for mode-free prosthetic control for unstructured terrains. *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, 796–802.

Rapp, E., Shin, S., Thomsen, W., Ferber, R., & Halilaj, E. (2021). Estimation of kinematics from inertial measurement units using a combined deep learning and optimization framework. *Journal of Biomechanics*, *116*, 110229.

Robert-Lachaine, X., Parent, G., Fuentes, A., Hagemeister, N., & Aissaoui, R. (2020). Inertial motion capture validation of 3d knee kinematics at various gait speed on the treadmill with a double-pose calibration. *Gait & Posture*, *77*, 132–137.

Schepers, M., Giuberti, M., Bellusci, G., et al. (2018). Xsens mvn: Consistent tracking of human motion using inertial sensing. *Xsens Technol*, *1*(8).

Sharma, A., & Rombokas, E. (2022). Improving imu-based prediction of lower limb kinematics in natural environments using egocentric optical flow. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *30*, 699–708.

Tschiedel, M., Russold, M. F., & Kaniusas, E. (2020). Relying on more sense for enhancing lower limb prostheses control: A review. *Journal of neuroengineering and rehabilitation*, *17*(1), 1–13.

Tucker, M. R., Olivier, J., Pagel, A., Bleuler, H., Bouri, M., Lambercy, O., Millán, J. d. R., Riener, R., Vallery, H., & Gassert, R. (2015). Control strategies for active lower extremity prosthetics and orthotics: A review. *Journal of neuroengineering and rehabilitation*, *12*(1), 1–30.

Zhang, X., Zhang, H., Hu, J., Deng, J., & Wang, Y. (2023). Motion forecasting network (mofcnet): Imu-based human motion forecasting for hip assistive exoskeleton. *IEEE Robotics and Automation Letters*.

Zhang, Z. (2012). Microsoft kinect sensor and its effect. *IEEE MultiMedia*, *19*(2), 4–10. https://doi.org/10.1109/MMUL.2012.24