

UC San Diego

UC San Diego Electronic Theses and Dissertations

Title

Evaluation of Full Body Pose Estimation Implementations Utilizing Sparse Sensors

Permalink

<https://escholarship.org/uc/item/1r08p9wm>

Author

Vo, Danny Minh

Publication Date

2024

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA SAN DIEGO

Evaluation of Full Body Pose Estimation Implementations Utilizing Sparse Sensors

A thesis submitted in partial satisfaction of the
requirements for the degree
Master of Science

in

Electrical Engineering (Computer Engineering)

by

Danny Minh Vo

Committee in charge:

Professor Dinesh Bharadia, Chair
Professor Erik Viirre
Professor Edward Wang

2024

Copyright
Danny Minh Vo, 2024
All rights reserved.

The thesis of Danny Minh Vo is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2024

TABLE OF CONTENTS

Thesis Approval Page	iii
Table of Contents	iv
List of Figures	v
Acknowledgements	vi
Abstract of the Thesis	vii
Chapter 1 Introduction	1
Chapter 2 Prior Works	4
2.1 IMUPoser	4
2.2 QuestSim	5
2.3 SlimeVR	6
2.4 Physical Inertial Poser	7
Chapter 3 Evaluation Method	8
3.0.1 Development of Test Suite	8
3.0.2 Experimentation Protocol	9
3.1 Experimentation Results	14
3.1.1 Evaluating Accuracy of Single Trial	14
3.1.2 Evaluating Consistency of Motions Across Trials	16
3.2 Limitations of Experimentation Method	16
Chapter 4 Future Work	22
4.0.1 Potential Improvements Using Intermediate Distances	22
Chapter 5 Conclusion	25
Appendix A Recording Videos and PIP Output	26
Bibliography	27

LIST OF FIGURES

Figure 3.1:	The categorized list of motions used to test for robust and accurate body pose estimations.	8
Figure 3.2:	The Vive tracker mount combined with the SlimeVR IMU module. It can also mount a Tottag, which utilizes ultra wide-band to determine distances between Tottags.	10
Figure 3.3:	A visualization showing how PIP's skeleton does not match the position of Vive trackers. Yellow lines indicate the offset vector between PIP's t-pose and the recorded Vive tracker t-pose.	12
Figure 3.4:	A reference t-pose and the coordinate system that recorded motion data needs to match.	13
Figure 3.5:	3D plot of the hip motion. This visually shows the disparity between ground truth and PIP's output pose.	15
Figure 3.6:	3D plot of the left leg motion. Simple motions are more accurate in comparison to Figure 3.5, but also shows PIP's preference to lower body pose accuracy.	15
Figure 3.7:	Plots of 5 tests of individual body segment movement. Each column represents the target body segment.	16
Figure 3.8:	Plots of the tracker errors for head, arm, and leg motions.	18
Figure 3.9:	Plots of the tracker errors for multiple body segment motions.	19
Figure 3.10:	Plots of the tracker errors with long motion recorded data for PIP.	19
Figure 3.11:	Plots of the tracker errors for single body segment movements across multiple trials.	20
Figure 3.12:	Plots of the tracker errors for simultaneous body segment movements across multiple trials.	20
Figure 3.13:	Plots of the tracker errors for typical motions and the cumulative motion recording across multiple trials.	21
Figure 4.1:	The characterization of Tottag to see if the angle between two tags correlate to the bias error. This is for one orientation between the two tags.	23
Figure 4.2:	Another characterization of Tottag similar to figure 4.1. This is for another orientation between the two tags.	23

ACKNOWLEDGEMENTS

I would like to express my gratitude to my mentors Professor Dinesh Bharadia and Dr. Aditya Arun. Professor Dinesh Bharadia has allowed me to understand the crux of research and ask questions regarding better testing and analysis method. Also, Dr. Aditya Arun has helped guide me in developing improvements to evaluating full body tracking implementations, as well as support on my journey to Masters.

Also I would like to thank Michael Shao for introducing me to the topic of full body pose estimation methods.

ABSTRACT OF THE THESIS

Evaluation of Full Body Pose Estimation Implementations Utilizing Sparse Sensors

by

Danny Minh Vo

Master of Science in Electrical Engineering (Computer Engineering)

University of California San Diego, 2024

Professor Dinesh Bharadia, Chair

Current methods of full body tracking have a steep correlation between hardware infrastructure and accuracy of the estimated full body pose. While implementations exist to allow for high accuracy body poses, they often require high setup overhead involving high accuracy sensors placed across all body ligaments or marker based solutions. To this end, there are existing implementations to approximate the accuracy of conventional full body tracking by utilizing sparse sensors, allowing for lower setup overhead. In this thesis, we evaluate the adeptness of Physical Inertial Poser (PIP) in accurately reproducing body poses through a mixture of single ligament and typical movements used for full body tracking. We determine that current body pose estimation solutions have a tendency to provide inaccurate results when compared against

corresponding Vive tracker data. These inaccuracies are a result of faults in using accelerometers for body pose estimation, which involves sensor accuracy, limited information to solve the body pose system, and accumulation of error over time. We also propose that current body pose estimation methods could be made more accurate by measuring intermediate distances between each tracker, specifically with ultra wide-band distance tracking. This would allow the body pose estimation to take into account the correlation between each tracker and body poses, especially with regards to the root origin of the body pose estimate.

Chapter 1

Introduction

The need for deriving a virtual full body pose from a physical human allows for a wide array of applications. Its use is widely seen in various applications such as robotics, virtual reality (VR), animation, and more. For robotics, full body tracking can be used to drive humanoid robots to match motions that humans make, such as the work presented by Spanlang et. al. in mapping body motions to robots [SNN⁺13]. On the other hand, there is an increasing interest in utilizing full body tracking towards animating virtual humanoid figures in VR in which there are numerous papers that delve into full body pose controlled avatars [CGAG20][JSQ⁺22]. However, few papers exist in examining the robustness of body pose estimation algorithms, especially in the context of its user toward VR applications.

There have been prior papers that have investigated the prevalence of technologies used to evaluate a full body pose. Caserman et. al. found that over the span of numerous publications, there were three different methods of which a full body pose was determined: markerless, marker-based, and inertial measurement unit (IMU) implementations. Markerless implementations use cameras and computer vision in order to estimate a person's body pose. Such implementations are limited when only one camera is used; in this case, Caserman et. al. mentions that papers often used additional sensors such as the Magic Leap or multiple cameras to help refine the resulting

body pose. Marker-based solutions are similar to markerless methods; however, infrared (IR) reflective material is placed around a body suit. Lastly, the researchers identified the use of IMUs in order to determine a body pose. In this method, high accuracy IMUs are used on parts of the body in order to determine the full body pose. In evaluating these methods, Caserman et. al. notes that at the time, many methods of full body poses were determined using camera based solutions rather than IMUs [CGAK⁺19].

The use of methods to determine full body pose estimations was not the only metric investigated by Caserman et. al. among other topics mentioned in the paper. The researchers also identified the faults of all three systems. In particular, camera based systems typically used for marker solutions tend to suffer from occlusion. To combat the possibility of line of sight loss for marker based setups, more cameras are needed. On the other hand, IMU solutions do not face occlusion problems but instead are influenced by the inaccuracy of IMU sensors, which results in the resulting body pose drifting from ground truth [CGAK⁺19].

In addition to these considerations, we must also consider the setup overhead that body pose estimation methods have. Marker based solutions tend to have the highest setup overhead because of the need for more cameras to avoid occlusion. On the other hand, marker-less methods have the least setup overhead due to the need for only one camera at minimum. However, the accuracy of marker-less body pose estimations are correlated to the camera's specifications such as resolution and the number of frames per second that it can see. This means that pose accuracy may be compromised if the camera does not have the right specifications for the required body tracking goals. Finally, IMU based setups have relatively high setup overhead as well due to the need for IMU sensors on each body segment. As a result, there are implementations that exist to reduce the setup overhead that IMUs typically have by reducing the number of sensors needed for body pose estimation.

In this paper, we will evaluate the efficacy of current solutions to full body pose estimation using fewer sensors by analyzing the mechanics in which these solutions derive body pose

estimates, specifically for Physical Inertial Poser (PIP). Following this, we will then investigate the accuracy and robustness of both implementations' body pose estimations, as well as propose a method to improve the output of full body pose estimations for IMU implementations.

Chapter 2

Prior Works

In order to more effectively evaluate current methods of inertial measurement units (IMU) full body pose estimations, we examine existing models that have tried to solve for full body pose estimation utilizing sparse sensors.

In explaining the current implementations of full body pose estimation, it is important to note the use of Skinned Multi-Person Linear Models (SMPL) in prior works. SMPL is an implementation to create realistic three-dimensional body models from skeletons provided as an input. This output can then be parsed into graphical programs or libraries that can parse the output mesh from SMPL. The prominent use of SMPL can be attributed to the variety of programs the implementation can output to such as Unity, Blender, and Maya [LMR⁺15]. The prior works mentioned utilize SMPL as their output for full body estimations.

2.1 IMUPoser

One such implementation is IMUPoser by Molyn et. al. IMUPoser uses IMUs typically used in Apple devices such as their headphones, phones, and watches. To achieve a full body pose estimation, they train a two-layer Bidirectional Long Short-term Memory (LSTM) neural network model. In this case, Molyn et. al. uses possible locations of the Apple devices, which

happens to be 5 locations, and transforms the 5 IMU datasets into a reduced linear unit (ReLU). This then gets passed into the Bidirectional LSTM model, which outputs a skinned multi-person linear model (SMPL). In order to train this model, Molyn et. al. utilized the Archive of Motion Capture As Surface Shapes (AMASS) as the input for training the neural network model.

Using this model, the researchers then tested the accuracy of their model with the different configurations available for placing the user's watch, phone, and earphone. They found that the accuracy of the output varied depending on the number of devices the user wore. In addition, the accuracy is comparable with other algorithms such as Physical Inertial Poser (PIP) while using fewer number of sensors.

Despite the closeness in accuracies between IMUPoser and other implementations, there are limitations that prevent IMUPoser from making accurate full body pose estimations. First, the number of sensors used for determining a body pose result in the pose being underconstrained. Because there are segments on the body without a sensor, the model has to guess the motion occurring on the other segments. This results in inaccuracies when body poses outside of its trained data are presented to the model, which is recognized by the paper's limitations. In addition, the paper also notes the low accuracy and polling rate of the IMUs used for the model. It may be possible to use higher accuracy IMUs for this model; nevertheless, this can introduce errors due to the model being trained on less accurate IMUs [MAG⁺23].

2.2 QuestSim

Another implementation to attain a full body pose is QuestSim. QuestSim by Winkler et. al. implements a full body pose estimation by using the position of the Quest 2 VR headset and its two controllers and then outputs a body pose estimate using a "reinforcement learning" model. This model takes into account the output pose and the state of the headset and controller positions in order to determine the next output pose. The model is able to adapt to different heights of the

person and the skeleton, which allows this model more flexibility on its use in full body tracking applications.

The results of QuestSim demonstrate that it is also comparable to PIP despite not having sensors on the legs. QuestSim determined the accuracy of their pose by utilizing PIP's method of evaluating pose accuracy, which involves analyzing the mean error of the poses's joints as well as the smoothness of the output pose.

However, despite achieving close results, there are significant issues with the model's robustness and accuracy. The inherent nature of this model requires the estimation of lower body poses. While QuestSim has relevant information regarding upper body poses, there are no sensors to get data to determine a lower body pose. In addition, the motions tested for QuestSim are largely tested on movement motions such as walking and running. This limits the versatility of the body pose estimation and results in a less robust body pose system. [WWY22]. The researchers also note that upper body poses can also have different poses from the same data, which also reduces the accuracy of the pose.

2.3 SlimeVR

SlimeVR is an open source implementation that determines a full body pose estimation utilizing IMUs. They are often used in VR social applications where only lower body locomotion is needed, but can still output a full body pose. The method in which SlimeVR determines a body pose is reminiscent to how normal IMU based solutions on body tracking is done due to the need for IMU sensors on each segment of the body. However, SlimeVR is intended to be used in VRChat, which only lacks lower body pose information. Thus, it only requires 5 sensors; however, it is possible to do full body tracking by using more sensors to match a typical IMU body pose tracking setup.

Despite its similarity to IMU based body pose tracking, it also inherits the method's

problems. SlimeVR utilizes BNO085 IMUs, which have the capability to achieve accurate full body tracking for short periods of time. However, SlimeVR's body pose in long periods of motion tracking degrades, requiring the user to reset the body pose after some time period. This is an indicator that while it can provide accurate body poses, SlimeVR does not prove to be robust due to the body pose accuracy degradation.

2.4 Physical Inertial Poser

Physical Inertial Poser (PIP) by Yi et. al. allows for deriving a full body pose estimate by using 6 sensors placed on the head, hips, wrists, and knees. The researchers propose that this method allows for determining accurate output body poses while allowing for low setup overhead. This is made possible through their use of two different models. The first is a recurrent neural network, which takes the input from the 6 IMUs and determines a body pose. This is then passed into a physics model that adjusts and optimizes the body poses with regards to the collision between the ground and the feet. The two networks thus allows for body poses that are accurate and smooth, as well as being able to determine the distance that the full body pose has travelled.

Nevertheless, there are issues that arise from utilizing few sensors in this manner. In this case, the model can be impacted by IMUs that are not as accurate compared to the ones used for the paper. This may result in the body pose being inaccurate. In addition, [YZH⁺22]

Chapter 3

Evaluation Method

In order to effectively analyze the accuracy and robustness of PIP, we need to develop a set of benchmark tests that examines the characteristics of PIP’s output pose estimation. In order to do so, we divide this requirement into four different categories. Through evaluating an ideal robust and accurate body pose estimation system, we develop 14 tests in total. We also need to test for whether the data is consistent across different trials.

3.0.1 Development of Test Suite

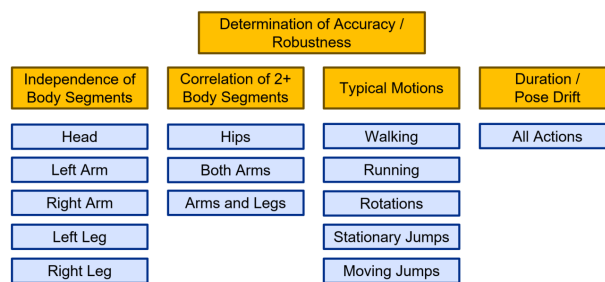


Figure 3.1: The categorized list of motions used to test for robust and accurate body pose estimations.

First, we need to be able to examine how PIP handles with correlations between body segment motions. The best case scenario is when the model is able to discern between when

a body segment moves by itself and when the body segment moves alongside other segments. This set of tests will require testing the movement of the arms, legs, and head. This thus results in testing the independence of body poses for the head, arms, and legs, as well as a second set of motions to analyze correlated motions involving the arms, legs, and hips. These comprise of 8 different tests where 5 of them test for body segment independence and 3 of them test for correlation of movement between body segments.

In addition to this, we need to also consider whether it is accurate with typical motions. This includes walking, running, rotations of the body, and jumping. For jumps, there is the possibility that a person can jump across to a different location. Thus, we need to include a different scenario to account for jumps that involve a translational motion. This category comprises of 5 motions in the list of test scenarios.

Lastly, in order to account for the robustness of PIP, we also need to consider whether it is able to accurate output poses over a long period of time. For this test, the previous motions listed from before are all done in one test. This approximates to 8 minutes of motion data for PIP to process.

While there may be more possible motions that can also effectively measure the accuracy and robustness of PIP, these 14 tests are enough to characterize PIP's output pose. In order to ensure that these tests are accurate, we also undergo these motions multiple times, resulting in a total of four trials in which these 14 motions are done. This allows us to test for the consistency of the output pose error.

3.0.2 Experimentation Protocol

Using the 14 tests and multiple trials, we now consider the collection method for obtaining motion recording data to pass into PIP and analyze the error of the output pose. In order to evaluate the accuracy and robustness of PIP, we need relevant ground truth to compare against as well as an IMU to use for PIP. For this, we utilize Vive Trackers and SlimeVR Trackers. Both are

attached to each other using a custom 3D printed mount.

Data Collection Method

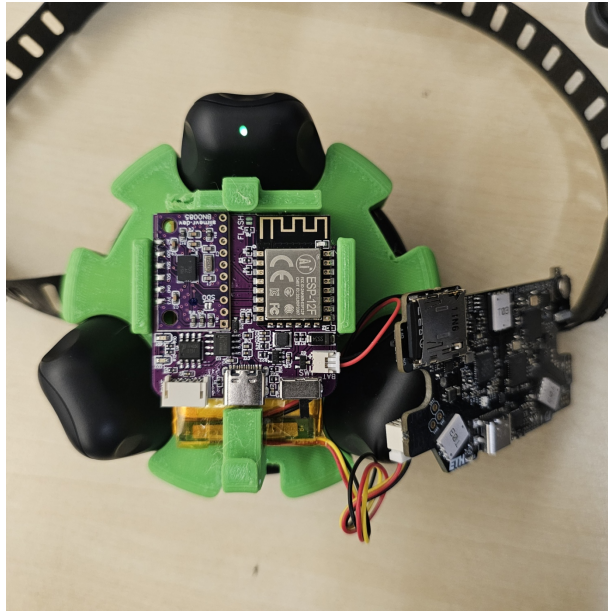


Figure 3.2: The Vive tracker mount combined with the SlimeVR IMU module. It can also mount a Tottag, which utilizes ultra wide-band to determine distances between Tottags.

Vive trackers are similar to marker based methods in that there are "lighthouses" that scan the tracked area in order to determine the position of a Vive Tracker. Vive trackers are also commonly used for accurate localization of points in the tracked area, which allows us to compare the body segment location with PIP's output pose.

In addition, we utilize SlimeVR IMU modules because they use the BNO085 IMU sensor from CEVA. These IMUs are typically used for SlimeVR due to their accuracy and low IMU noise, though it is not as accurate as IMUs found on a dedicated IMU full body tracking setup. However, these trackers do not communicate their accelerometer or rotation data outside of SlimeVR. Therefore, custom firmware needed to be developed in order to extract the relevant IMU data needed for PIP.

PIP requires data from 6 trackers placed on the head, hips, arms, and knees. In order

to obtain tracking information from both the Vive tracker and SlimeVR module for all of these positions on the body, we needed to create a system that is able to record this data at 60 frames per second. This is because the input of PIP is expected to be 60 frames per second. Therefore, we took advantage of the Robot Operating System (ROS) to synchronize the data collection of the Vive tracker and SlimeVR module. This allowed us to streamline the data collection while ensuring that PIP receives the correct number of frames per second. We also need to ensure that motion recordings with this setup start with a t-pose. PIP notes that the identity rotation matrix corresponds to t-pose, so calculations must be done to ensure that the rotations of the IMU are relative to the identity matrix. After PIP is given the IMU data for all 6 trackers, we need to consider how to compare between PIP's output skeleton and the reference Vive tracker positions.

Matching PIP Skeleton Data to Ground Truth

After IMU data is parsed by PIP, the output skeleton is given in the SMPL format. However, this skeleton does not match the scale of the reference Vive tracker locations. In reality, the output of PIP's skeleton is smaller than the reference body pose. We need a way to compare between positions of the skeleton and the Vive tracker while accounting for the difference in scale. To this end, we developed calculations to calculate the expected position where PIP skeleton joints are expected to be.

First, we consider the root coordinate system to be the hip tracker for both PIP's output pose and our Vive tracker. This allows us to compare body poses relative to the root trackers. However, we also need to determine the offset vectors to compare PIP's joints against where the joints should be. This is achieved through using the Vive tracker as the parent of the expected ground truth position of the PIP joint. This requires us to calculate the offset vector between the Vive tracker and the accurate location where the PIP joint needs to be located. We can compare the t-pose of the Vive trackers and PIP to determine the offset vector because all recordings need to begin with a t-pose. This vector is computed as such:

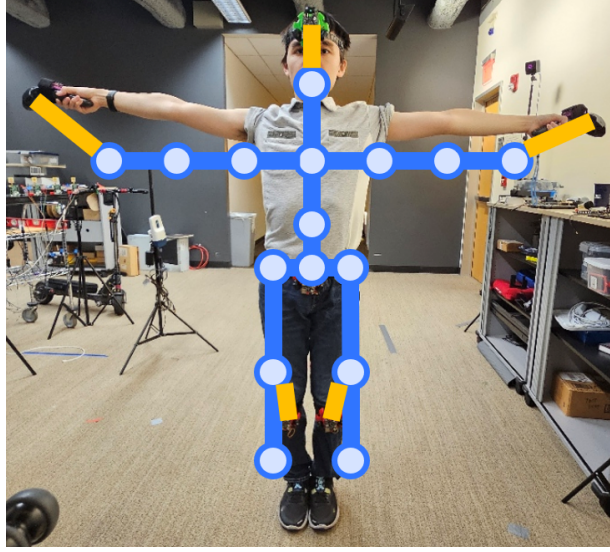


Figure 3.3: A visualization showing how PIP’s skeleton does not match the position of Vive trackers. Yellow lines indicate the offset vector between PIP’s t-pose and the recorded Vive tracker t-pose.

$$\overrightarrow{offset} = (x, y, z)_{PIPt-pose} - (x, y, z)_{Vivet-pose}$$

In addition to this, this offset needs to account for the rotation of the tracker. However, depending on the joint, we need to remove the corresponding rotation on the axis of the body segment’s rotation. Afterwards, we note down the rotation matrix of the Vive tracker’s t-pose and apply it as such:

$$(x, y, z)_{PIP,GT} = ((x, y, z)_{Vive} - \overrightarrow{offset}) \times R_{Vivet-pose}$$

This allows us to determine the ground truth location of where PIP’s skeleton point should be. For every subsequent point of the motion recording, this vector is applied to each Vive tracker and becomes the ground truth comparison for each respective joint of PIP. This full computation is seen as the following equation:

$$||Error_{GT}|| = |(x, y, z)_{PIP,GT} - (x, y, z)_{PIP}|$$

Using this process, we are able to obtain data regarding the accuracy of PIP. However, the head joint requires a different process for accuracy determination. This is because the offset calculation process would result in the expected location of PIP's head to rotate with respect to the Vive tracker's head rotation. As such, we instead determine the delta distance of the Vive tracker's motion, scale the delta down, and apply this distance on the Vive tracker position. This results in the expected PIP point for deriving the error:

$$||scale_{HEAD}|| = ||\overrightarrow{offset}||$$

$$(x,y,z)_{PIP,HEADGT} = ||scale_{HEAD}|| \times ((x,y,z)_{Vivet-pose} - (x,y,z)_{Vivecurframe})$$

In addition to this calculation, we also reorient the axes of the body pose to match the coordinate system of typical full body poses. The coordinate system that we reference is indicated in Figure 3.4.

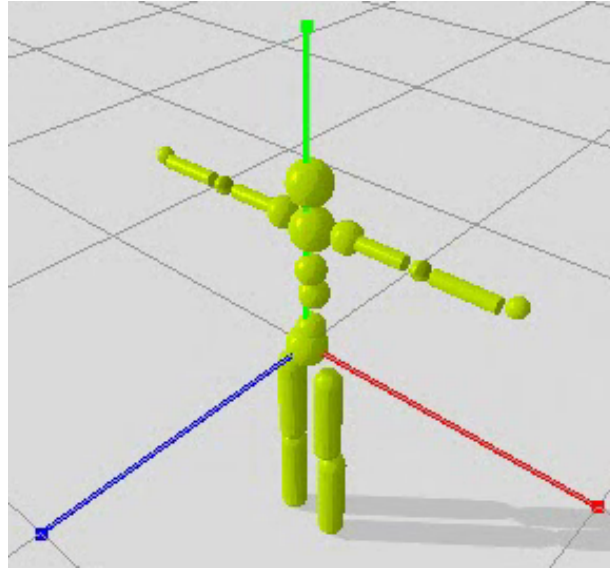


Figure 3.4: A reference t-pose and the coordinate system that recorded motion data needs to match.

3.1 Experimentation Results

To effectively analyze the data retrieved after retrieving the recordings from 14 tests across 4 trials and evaluating the ground truth, we examine the data in two ways. The first is examining the results of one trial and drawing conclusions about the behavior of PIP. Afterwards, we will investigate the consistency of such results across other trials.

3.1.1 Evaluating Accuracy of Single Trial

In examining the results of one trial, we find that a large majority of PIP body poses have high error measurements. We observe this through the four categories determined from detailing the experimentation process. These graphs are denoted in Figures 3.7, 3.8, 3.9, and 3.10.

From these graphs, we can determine key characteristics of PIP that results in the model showing signs of inaccuracies and a lack of robustness. The first conclusion we draw from this data is that PIP tends to prioritize lower body pose accuracy over upper body poses. This is visible in Figures 3.9 and 3.10 where the errors for the lower body, which are the knees, are much lower compared to the left and right wrists. Because this trend appears across all the tests done for this trial, we can theorize that PIP's neural network that determines the body pose is weighted toward ensuring lower body poses over upper body poses.

Second, the error data suggests that PIP does not provide accurate body poses when multiple body segments are in motion. This is shown in Figure 3.9, which demonstrates motions from multiple body segments. We can see that when lower body motions occur alongside upper body movements, the error generally increases with respect to the other motions that only involve arm movement. Thus, PIP operates best with fewer body sections are moving at the same time. This behavior is visible by examining the motion data. In Figure 3.5, we can visualize the movement for specifically hip movement where the head and arms are only moving. We can see that there are deviations between the ground truth of the left and right wrists, indicating that PIP

struggles to maintain accurate body poses in this motion. On the other hand, we can also observe the better accuracies in Figure 3.6, where leg motions are closer to the reference ground truth.

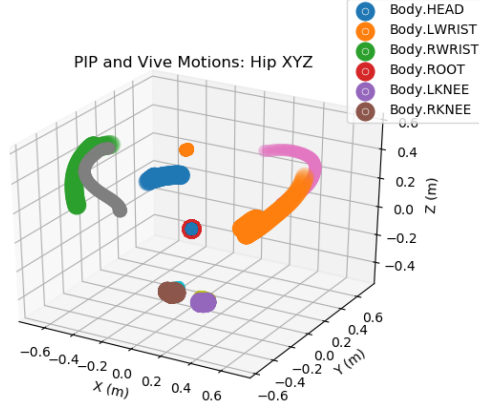


Figure 3.5: 3D plot of the hip motion. This visually shows the disparity between ground truth and PIP's output pose.

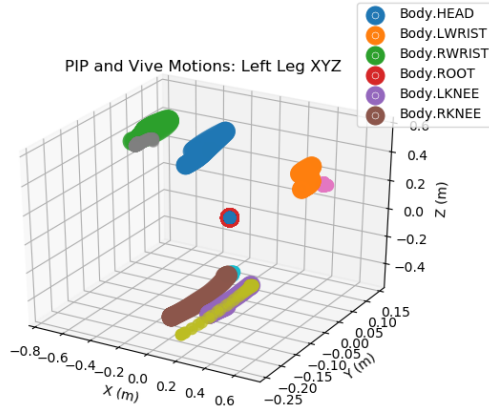


Figure 3.6: 3D plot of the left leg motion. Simple motions are more accurate in comparison to Figure 3.5, but also shows PIP's preference to lower body pose accuracy.

Lastly, we notice from Figure 3.10 that the prior conclusions do not allow for robust or precise body pose estimations. The results reveal that there exist high measurements of error across the 8 minutes of motion data. This is a consequence of the conclusions we draw from before and thus PIP is not an effective implementation for robust and accurate body pose estimations.

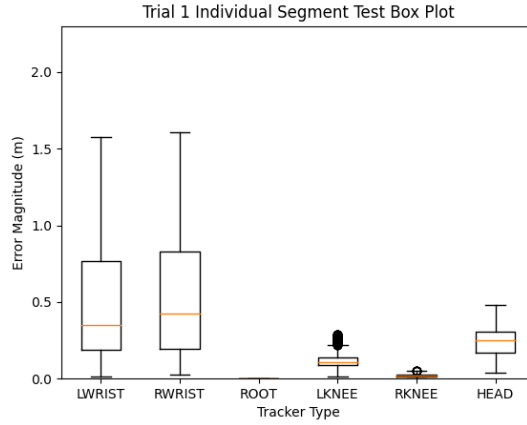


Figure 3.7: Plots of 5 tests of individual body segment movement. Each column represents the target body segment.

3.1.2 Evaluating Consistency of Motions Across Trials

In the previous section, we found characteristics that determine the inability of PIP to deliver accurate poses. Here, we examine whether the errors are consistent across different trials. These are demonstrated in Figures 3.11, 3.12, and 3.13. Across multiple trials, we can observe that the mean error across 4 trials are relatively close to each other in most tests. There are some tests where there are significant differences in mean, though this is likely a result of issues regarding the Vive tracker losing its tracking information for a period of time. Nevertheless, we identify that the behavior noted from the previous section persists over multiple trials.

3.2 Limitations of Experimentation Method

In evaluating the accuracy of SlimeVR and PIP, there were limitations inherent to the equipment used as well as assumptions made on determining the accuracy of the full body pose.

First, the BNO085 IMU used as input for SlimeVR and PIP is not as accurate compared to the sensors used for PIP or similar implementations. PIP and other prior works utilized IMUs that are typically used in high accuracy IMU body tracking suits, which allows for higher measurement

precision, low sensor noise, and polling rate compared to the BNO085, which was limited to a polling rate of 100Hz. We presume that PIP and other implementations using high accuracy sensors allow for more digits of precision to accommodate for the accumulating error that PIP and similar models exhibit; however, it is still a limitation of the testing environment in order to effectively analyze the accuracy of PIP. The device also has a tendency to report inaccurate data when the sensor experiences a sharp change in acceleration. Our recordings try to limit this effect by executing motions without stomping motions or hard impacts, but this may be a factor that affects the accuracy of the output pose.

Second, measuring the accuracy of the skeleton by taking the Vive tracker as the expected parent skeleton point of PIP's skeleton has heavy assumptions regarding the position of the skeleton. In this case, rotations occurring on the axis of each body segment would result in the expected location of PIP's skeleton to rotate due to the nature of offsetting with respect to the Vive trackers. This was mitigated through ignoring rotations occurring on the axis of the segment, but this could not be done for the head, as the head is expected to move proportionally to the Vive tracker at a smaller scale.

We believe that such an evaluation with higher accuracy sensors and an accurate full body pose estimation as ground truth will provide a better analysis on the faults of full body pose estimation algorithms utilizing sparse sensors.

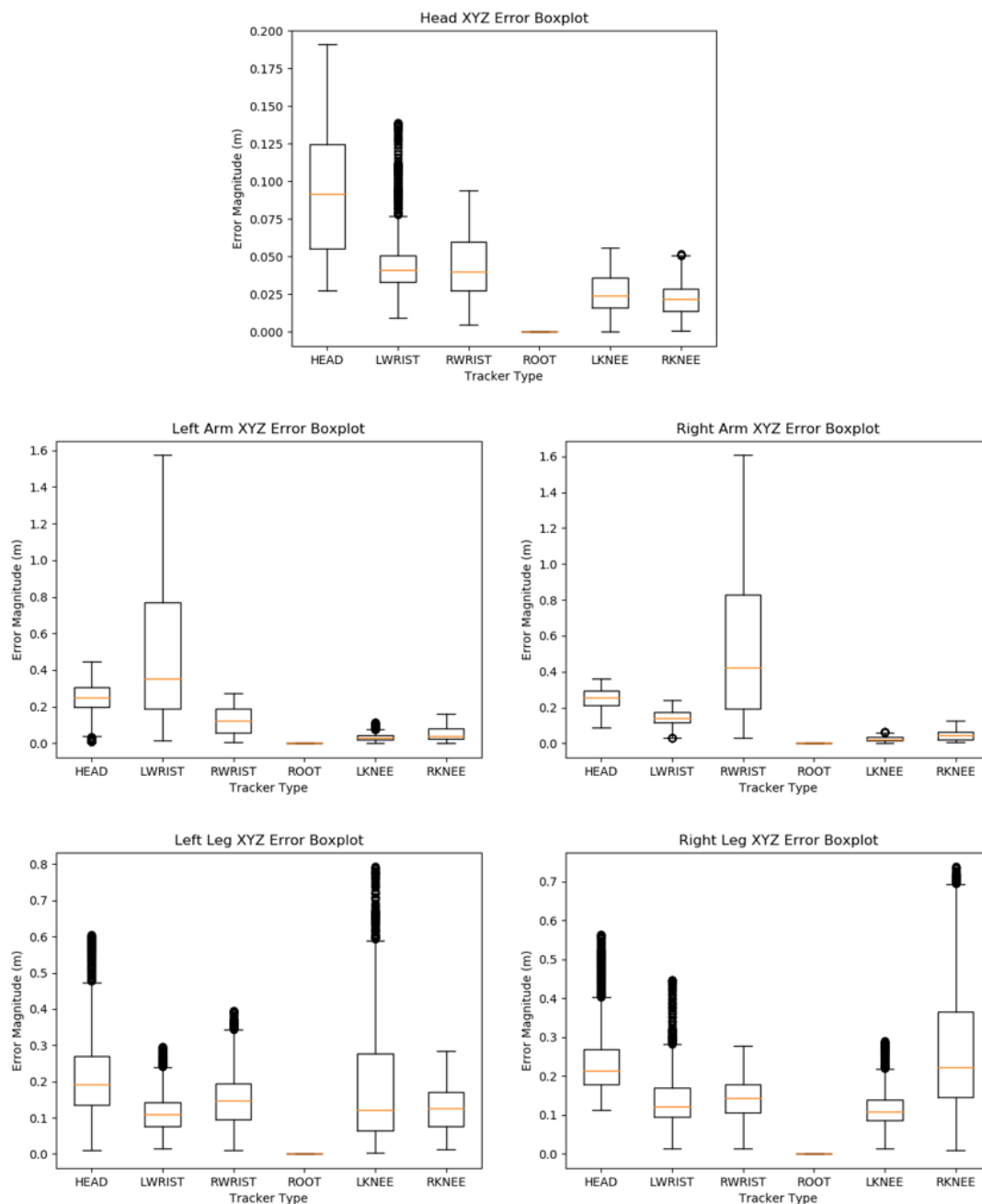


Figure 3.8: Plots of the tracker errors for head, arm, and leg motions.

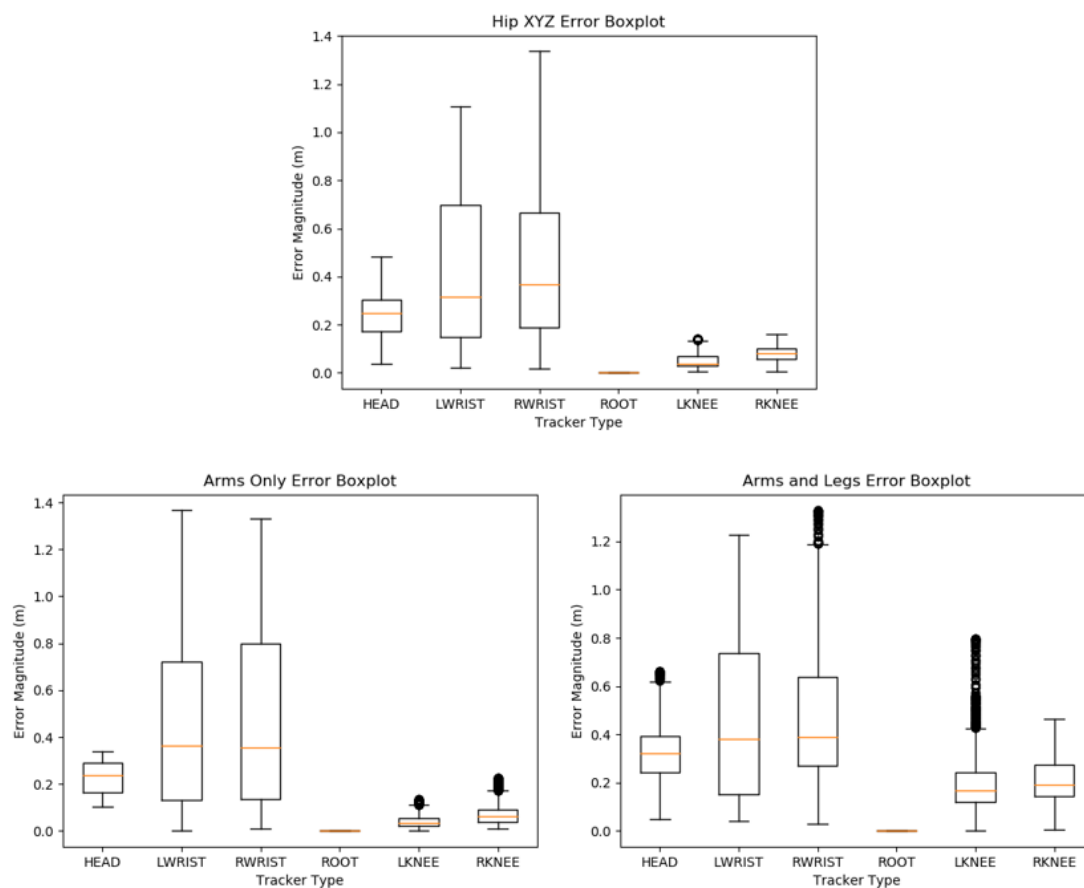


Figure 3.9: Plots of the tracker errors for multiple body segment motions.

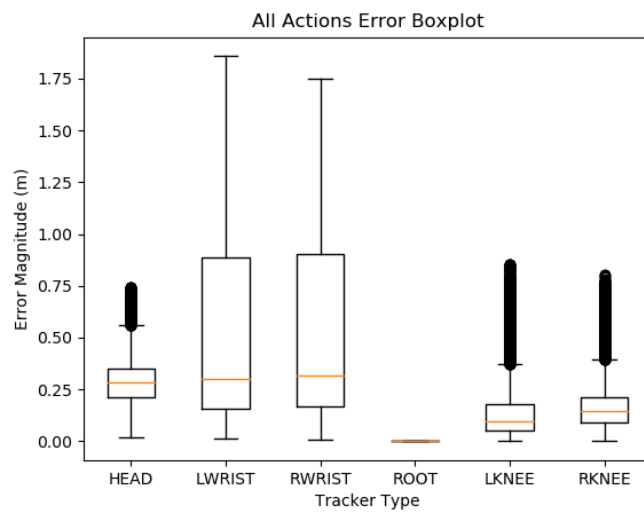


Figure 3.10: Plots of the tracker errors with long motion recorded data for PIP.

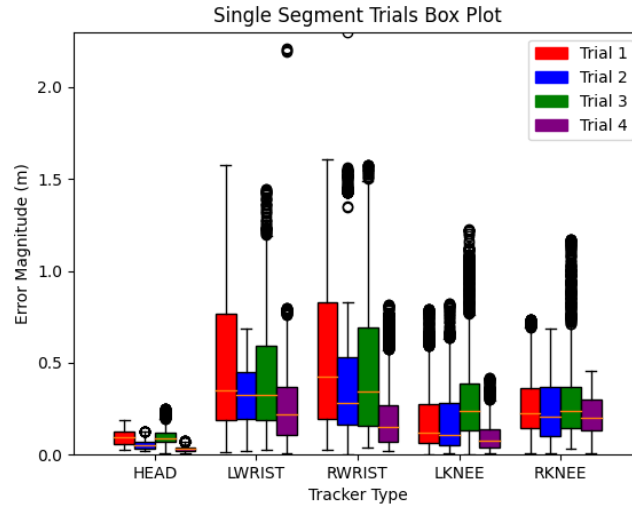


Figure 3.11: Plots of the tracker errors for single body segment movements across multiple trials.

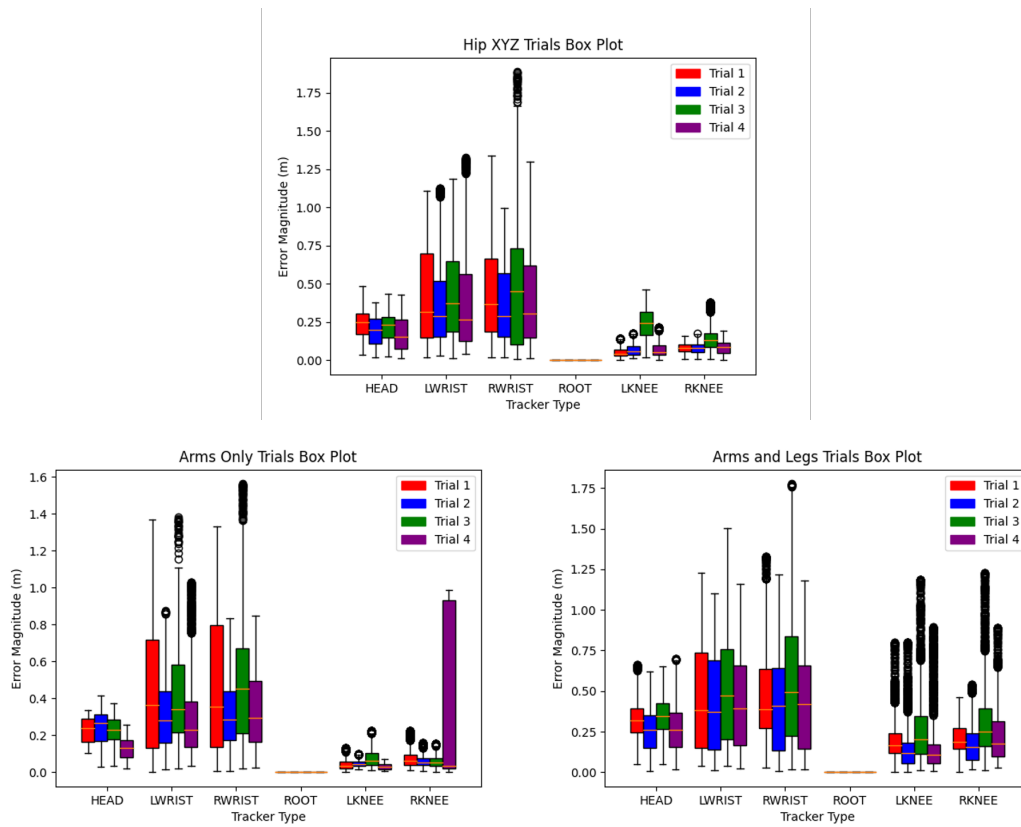


Figure 3.12: Plots of the tracker errors for simultaneous body segment movements across multiple trials.

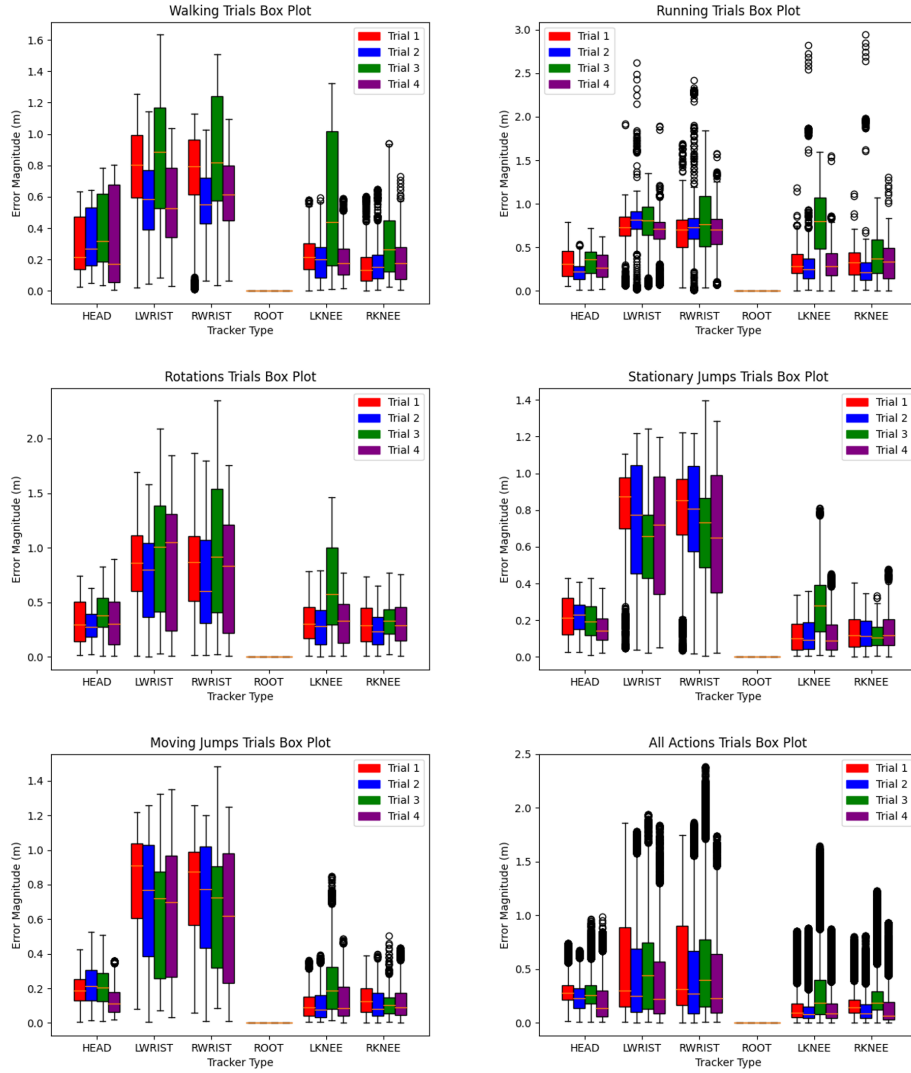


Figure 3.13: Plots of the tracker errors for typical motions and the cumulative motion recording across multiple trials.

Chapter 4

Future Work

In the future, we would like to refine the analysis in the accuracy of full body pose estimations. PIP not only outputs the body pose estimated but also gives an estimation on how the skeleton has travelled in the environment. We highly encourage analyzing this without the use of external devices such as cameras and markers in order to gauge the effect of sensor drift and noise, as well as the robustness of such body pose estimation models.

Furthermore, we see the potential in refining full body pose estimations through introducing intermediate distances as a metric in training the neural network with full body poses.

4.0.1 Potential Improvements Using Intermediate Distances

We propose a method to introduce intermediate distances between tracker devices utilizing ultra wide-band (UWB) implementations. When training neural networks on body pose data with IMUs, we notice that the behaviour of the model tends to correlate the motion of one body segment with other body segments. This results in a lack of flexibility to account for motions that have not been trained on. Thus, measuring distances between each sensor on the body would allow for the neural network to better comprehend the body pose estimation system.

For this matter, we briefly test this potential utilizing Tottag UWB revision H units,

which have the capability of measuring distances between each device. Tottag obtains the distance between itself and other tags by employing an algorithm that broadcasts UWB signals on different channels and getting a distance from each channel based on time of flight. In addition, each tag synchronizes this broadcast using a flood of UWB broadcast packets, allowing for the measurement of distances between multiple tags. Using Tottag, we combined this with a Vive Tracker to evaluate its efficacy in determining accurate distances for use in refining full body pose estimations.

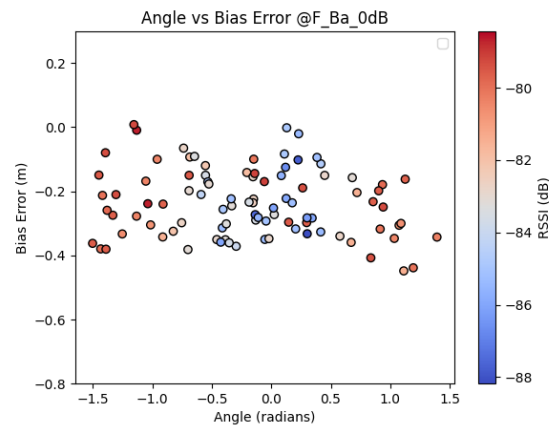


Figure 4.1: The characterization of Tottag to see if the angle between two tags correlate to the bias error. This is for one orientation between the two tags.

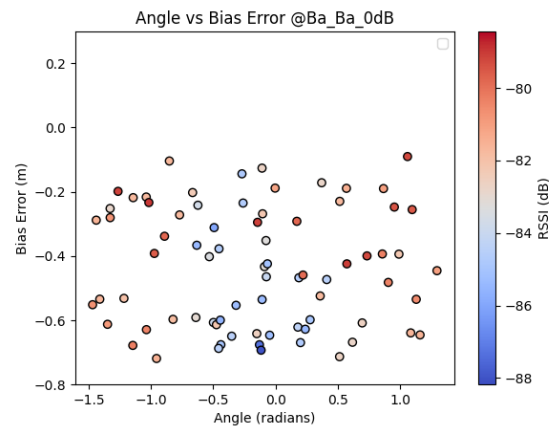


Figure 4.2: Another characterization of Tottag similar to figure 4.1. This is for another orientation between the two tags.

In the above figures, we briefly test whether a correlation can be formed between the angle between trackers and the bias error. From these tests, it is difficult to determine a characterization due to the difference in noise. However, we surmise that future versions of the Tottag have fixed this issue to allow for more accurate UWB distance characteristics [BJT⁺20].

Chapter 5

Conclusion

In this paper, we determine the current state of achieving low setup costs and robust body tracking. Currently, PIP does not demonstrate robust and accurate body poses when tested against ground truth due to large errors involved with upper body poses. PIP's accuracy worsens in cases where multiple body segments move, which indicates that PIP is also not able to handle long durations of motion recording or provide accurate body pose estimates. We also determine the characteristics of PIP's pose estimation in which lower body pose accuracy is prioritized over upper body poses. Lastly, we propose possible improvements to this estimation by measuring distances between trackers, which would give PIP more information to correlate it with IMU data.

Appendix A

Recording Videos and PIP Output

Video recordings of the movements done for this thesis can be found in the link below:

https://ucsdcloud-my.sharepoint.com/:f/g/personal/dbharadia_ucsd_edu/EjGz9QfpKl1BhCmzd9ADPu0Be_DQ_m3SGJuz1-czk8HwjA?e=RhzsCM

Bibliography

- [BJT⁺20] Andreas Biri, Neal Jackson, Lothar Thiele, Pat Pannuto, and Prabal Dutta. Soc-track: infrastructure-free interaction tracking through mobile sensor networks. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, MobiCom '20, New York, NY, USA, 2020. Association for Computing Machinery.
- [CGAG20] Polona Caserman, Augusto Garcia-Agundez, and Stefan Göbel. A survey of full-body motion reconstruction in immersive virtual reality applications. *IEEE Transactions on Visualization and Computer Graphics*, 26(10):3089–3108, 2020.
- [CGAK⁺19] Polona Caserman, Augusto Garcia-Agundez, Robert Konrad, Stefan Göbel, and Ralf Steinmetz. Real-time body tracking in virtual reality using a vive tracker. *Virtual Reality*, 23:155–168, 2019.
- [JSQ⁺22] Jiaxi Jiang, Paul Strel, Huajian Qiu, Andreas Fender, Larissa Laich, Patrick Snape, and Christian Holz. Avatarposer: Articulated full-body pose tracking from sparse motion sensing. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 443–460, Cham, 2022. Springer Nature Switzerland.
- [LMR⁺15] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.
- [MAG⁺23] Vimal Mollyn, Riku Arakawa, Mayank Goel, Chris Harrison, and Karan Ahuja. Imuposer: Full-body pose estimation using imus in phones, watches, and earbuds. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, New York, NY, USA, 2023. Association for Computing Machinery.
- [SNN⁺13] Bernhard Spanlang, Xavi Navarro, Jean-Marie Normand, Sameer Kishore, Rodrigo Pizarro, and Mel Slater. Real time whole body motion mapping for avatars and robots. In *Proceedings of the 19th ACM Symposium on Virtual Reality Software and Technology*, VRST '13, page 175–178, New York, NY, USA, 2013. Association for Computing Machinery.

- [WWY22] Alexander Winkler, Jungdam Won, and Yuting Ye. Questsim: Human motion tracking from sparse sensors with simulated avatars. In *SIGGRAPH Asia 2022 Conference Papers*, SA '22, New York, NY, USA, 2022. Association for Computing Machinery.
- [YZH⁺22] Xinyu Yi, Yuxiao Zhou, Marc Habermann, Soshi Shimada, Vladislav Golyanik, Christian Theobalt, and Feng Xu. Physical inertial poser (pip): Physics-aware real-time human motion tracking from sparse inertial sensors. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.