2018

A Quantitative Evaluation of the HTC Vive for Virtual Reality Research

Ethan Luckett

University of Mississippi. Sally McDonnell Barksdale Honors College

Follow this and additional works at: https://egrove.olemiss.edu/hon_thesis

Part of the Computer Sciences Commons

Recommended Citation


https://egrove.olemiss.edu/hon_thesis/331

This Undergraduate Thesis is brought to you for free and open access by the Honors College (Sally McDonnell Barksdale Honors College) at eGrove. It has been accepted for inclusion in Honors Theses by an authorized administrator of eGrove. For more information, please contact egrove@olemiss.edu.
A Quantitative Evaluation of the HTC Vive for Virtual Reality Research

by

Ethan Luckett

A thesis submitted to the faculty of The University of Mississippi in partial fulfillment of the requirements of the Sally McDonnell Barksdale Honors College.

Oxford
May 2018

Approved by

Advisor: Professor J. Adam Jones

Reader: Professor Dawn Wilkins

Reader: Professor Philip Rhodes
ACKNOWLEDGEMENTS

I would first like to thank my advisor, Dr. J. Adam Jones, whose great enthusiasm sparked my interest in virtual reality and whose patience and guidance were crucial to the completion of this work. I am thankful to my second reader, Dr. Dawn Wilkins, for her additional advice.

I am also grateful for the resources provided by the Sally McDonnell Barksdale Honors College and the Department of Computer and Information Science.

Most of all, I am grateful for the undying support of my family and friends.
ABSTRACT

The equipment typically used in virtual reality (VR) research, including head-mounted displays (HMDs) and motion capture systems, has traditionally been prohibitively expensive. The recent increase in the availability of consumer-grade VR equipment has greatly lowered the barrier to entry for VR research. The equipment typically used for research can cost upwards of tens of thousands of dollars, but the consumer-grade HTC Vive system offers an HMD with room-scale tracking for less than $500. In order for scientific studies to be properly conducted using the Vive, its tracking must be well understood. This study measures the accuracy and drift in the Vive’s tracking and compares it to the research-grade PhaseSpace motion capture system. The methods in this study are based on those in Niehorster et al. [1] with modifications to allow more precise measurements and a reduced possibility for human error.
TABLE OF CONTENTS

ACKNOWLEDGEMENTS ........................................... iii

ABSTRACT ............................................................ iv

LIST OF FIGURES ................................................... vi

INTRODUCTION ...................................................... 1

RELATED WORK ..................................................... 2

HARDWARE AND SOFTWARE .................................... 5

EXPERIMENTS ....................................................... 10

DISCUSSION ......................................................... 22

CONCLUSION ........................................................ 24

BIBLIOGRAPHY ..................................................... 25
CHAPTER 1

INTRODUCTION

Traditionally, research in virtual reality (VR) has been difficult due to the high cost of equipment, including head-mounted displays (HMDs) and room-scale motion tracking systems. The sharp increase in the popularity of VR in the consumer market in recent years has brought with it reliable VR equipment at a fraction of the cost of the systems historically used in research. One such system, the HTC Vive, features room-scale tracking and a fist-sized tracker that can be mounted to ISO 1222:2010 standard 1/4" camera mounts.

At the time of writing, the HTC Vive costs under $500. For comparison, the research-grade PhaseSpace motion capture system used in this study costs approximately $20,000. The Vive’s low cost makes it attractive to VR researchers, but there is limited detailed documentation on its tracking technology [2]. In order for scientific research to be properly conducted using the Vive, the flaws and quirks of its tracking should be well understood. This study aims to measure the accuracy and drift in the Vive’s tracking. The tracking accuracy is measured by comparing a set of positions measured with the Vive to positions triangulated using two laser rangefinders and to positions recorded with a research-grade PhaseSpace motion capture system. Drift is measured both with and without intermittent tracking loss. The methods used in this study extend, in part, the general measurement procedures of Niehorster et al. [1] with some alterations to allow higher precision and a reduced possibility for human error.
CHAPTER 2

RELATED WORK

The methods in this study are based on those in Niehorster et al. \[1\], which analyzed the accuracy of the Vive’s tracking using a grid marked on the floor with a chalk line. The reported mean positioning error of each grid point was 1.7cm (stdev: 0.9cm), and the error in manual placement of the Vive HMD on the grid points was less than 2cm. The accuracy of the Vive’s tracking along this grid was determined with the Vive facing along both the positive and negative x-axis. The error between the grid position and Vive-tracked position was found to be low, but was not explicitly stated. The grid formed by the Vive-tracked positions was slightly rotated relative to the chalk line grid. In the present study, laser rangefinders were used instead of a chalk line grid, which allowed the real-world positions to be measured to within 1cm.

Analysis of the Vive’s tracked positions showed a systematic height difference over the tracking area, suggesting that the Vive’s ground plane is tilted relative to the real-world ground plane. After losing tracking, there was a systematic offset in the reported position, suggesting that the tilt in the ground plane and potentially its position change when tracking is intermittently lost.

Niehorster et al. measured the latency of the Vive HMD by physically moving the HMD and counting the number of screen refreshes that occurred before the view of the environment in the HMD updated to reflect the movement. On average, two refreshes elapsed. At the Vive’s 90Hz refresh rate, the screen updates every 11ms, which puts an upper bound for the system latency at 22ms.
2.1 Head Tracking Latency

The threshold at which a difference in latency is discernible in virtual environments was measured in Adelstein et al. [3] using a Virtual Research V8 HMD, which has a mean latency of 33ms and a refresh rate of 60Hz. The study concluded that in order for latency in a virtual environment to be imperceptible, system latency must be no higher than 17ms. Niehorster et al. determined the upper bound on the Vive’s system latency to be 22ms, which, while slightly larger than the 17ms limit recommended in Adelstein et al. [3], is low and should result in minimal perceived latency.

2.2 Tracking technology

Six different categories of tracking technology are described in Rolland et al. [4], but the most relevant to this study is spatial scan tracking. Spatial scan tracking systems track the position and orientation of a target with respect to a reference coordinate system using optical sensors and emitters. There are two subcategories of spatial scan tracking: outside-in and inside-out tracking.

2.2.1 Outside-in tracking

Outside-in tracking includes one or more cameras placed on the perimeter of the tracking area and one or more markers on the tracked object(s). The system uses images taken from the cameras and a priori knowledge about the configuration of the markers to determine the position and orientation of each tracked object. ThePhaseSpace motion capture system used in this study relies on outside-in tracking.

2.2.2 Inside-out tracking

In inside-out tracking, the placement of sensors and markers is reversed. The sensors are on the tracked object, and the markers are on the perimeter of the tracking area. Inside-out tracking is further divided into videometric and beam-scanning tracking.
Videometric tracking utilizes multiple cameras placed on the tracked object. The perimeter of the tracking area has features whose positions in 3D space are known. Images from the cameras and knowledge of the position of each feature in 3D space are used to determine the position and orientation of the tracked object.

Beam-scanning tracking utilizes sweeping optical beams on the perimeter of the tracking area and optical sensors on the tracked object. The beam sweep is detected across each sensor, and the system uses the delay between the beam hitting each sensor to determine the position and orientation of the tracked object. This is the tracking method used by the HTC Vive and will be discussed in further detail in the next chapter.
CHAPTER 3

HARDWARE AND SOFTWARE

3.1 HTC Vive

The HTC Vive is a consumer-grade virtual reality system designed for use in video games. It consists of an HMD, two controllers, and, optionally, one or more Vive trackers [5]. These components are shown in Figure 3.1.

The Vive tracker is a small (99.65mm x 42.27mm), self-contained unit, which allows a wide range of objects to be tracked [5]. This is in contrast to some research-grade systems which may require many cameras or sensors around the perimeter of the tracking area, as well as many separate markers placed in different locations on the tracked object.

The Vive’s tracking technology is called Lighthouse [2]. There are two base stations or lighthouses, each of which contains an infrared (IR) light beacon and two IR laser fans spinning on orthogonal axes. The IR beacon emits a sequence of light pulses, which is followed by a sweep of one of the IR laser fans. The IR pulse sequence encodes several bits of information, including which of the two laser fans will begin sweeping next. This information along with the time delay between the IR pulse and the laser fan sweep hitting each IR sensor on each tracked object is used to calculate the position and orientation of each tracked object [2].

3.2 PhaseSpace Motion Capture System

A PhaseSpace Improv Motion Capture system was used in Experiment 4. The PhaseSpace system is a commercial motion capture system designed for use in virtual
and augmented reality research as well as motion capture for cinema [7]. It consists of a tracking server that performs the computations necessary for real-time tracking (Figure 3.2), a series of cameras (Figure 3.3) connected to the server, and active marker LEDs attached to one or more LED controllers. The system used in this study utilized four cameras placed in the corners of the tracking area. Each active marker LED modulates at a unique frequency, which eliminates the marker swap seen in passive marker systems. Each LED must be seen by at least two cameras in order for its position to be determined via triangulation. The accuracy of the tracked position increases with the number of cameras to which an LED is visible.

The calibration procedure for the PhaseSpace is initiated from the PhaseSpace Master Client software. A calibration wand, which is fitted with 8 active marker LEDs in known positions, is used to provide a point cloud over the entire tracking area. The PhaseSpace system then determines the relative positions of the cameras using this point cloud. Once relative positions are known, the tracking space must be aligned to the real-world floor. This is also performed via the Master client. The wand is placed vertically at three positions on the floor, and its position is recorded at each location. The first position defines the origin of the coordinate system, and together the three points define the orientation of the ground plane.
Once calibration is complete, the tracked object must be encoded. In the PhaseSpace Configuration Manager, a profile is created for the tracked object. This specifies the number of LED controllers used and the number of LEDs connected to each controller. Next, in the Master client, the active profile is swapped to this newly created profile. The tracked object is placed in the tracking area with every LED visible to at least two cameras, and a rigid body tracker is created in the Master client, which defines the relative positions of the LEDs and the origin of the tracked object’s coordinate system.

![PhaseSpace server](image)

**Figure 3.2: PhaseSpace server**

### 3.3 Unity3D

The Unity3D game engine was used for the majority of the data collection. The SteamVR plugin for Unity3D was used to provide integration with the HTC Vive to allow the position and orientation of the Vive HMD, controllers, and trackers to be recorded. The CameraRig prefab from the plugin, which includes all necessary components to add Vive HMD and controller tracking functionality to a Unity scene, was placed in an empty scene. Additional GameObjects were created as needed for each Vive tracker used, each with a SteamVR_TrackedObject script attached. This script serves to update the position and orientation of the GameObject to which it is attached to the position and orientation of the device that it is configured to track.
The Device ID parameter of the script determines which device will be tracked by the script. The following was observed regarding the ID values of the various Vive components:

- IDs 1 and 2 were always the Vive lighthouses
- If the controllers were turned on, they were mapped to IDs 3 and 4
- Subsequent devices (trackers) were then enumerated beginning at ID 5
- If the controllers were off, tracker enumeration typically began at ID 3, but occasionally it began at ID 5. The reason for this is unknown.
- Although the lighthouses, controllers, and trackers were enumerated in that order, the order in which the individual controllers and trackers were enumerated appeared to be arbitrary. For example, the left controller could be assigned ID 2 during one run of an experiment but might be assigned ID 1 during a second run of the same experiment.

Experimentation showed that the script updates the position and orientation at a maximum rate of 90Hz. A custom C# script was written to record the position
and orientation of each tracked object. The script logged the position of the device to which it was attached every 10ms, and after a specified number of seconds, the positions and orientations were written to a CSV file.

3.4 Rangefinders

Two laser rangefinders were used for real-world distance measurements: one Bosch GLM 42 and one Bosch GLM 50 C. The manufacturer specified that they are accurate to within 1.6mm.
CHAPTER 4

EXPERIMENTS

Four experiments were performed. Experiment 1 measured the difference between the physically measured position of a Vive tracker using laser rangefinders and the position reported by the SteamVR plugin in Unity3D. Experiment 2 measured the drift in the reported position of the Vive HMD and controllers when tracking was intermittently lost, and Experiment 3 measured the drift of the HMD, controllers, and a tracker over a period of 20 minutes with no intermittent loss of tracking. Experiment 4 compared the position reported by the Vive to the position reported by the PhaseSpace system. Each experiment utilized one or both of two tracking areas: a larger tracking area with an inter-lighthouse distance of 7.6m and a smaller tracking area with an inter-lighthouse distance of 6.3m. These are both larger than the recommended maximum of 5.5m, but testing the Vive’s tracking under somewhat stressed conditions gives a better sense of its capabilities.

All data analysis was performed using Python 3.6.4 and the NumPy and SciPy Python libraries, and visualizations were created using the Matplotlib Python library.

4.1 Experiment 1

Experiment 1 measured the physical position of a Vive tracker mounted to a camera tripod and compared it with the position reported by the Vive via Unity’s SteamVR plugin. A second tracker placed on a level surface was used to compare the accuracy of the absolute position of the tripod-mounted tracker reported by Unity with the relative position of the tripod tracker with respect to the second tracker. This experiment was performed in both tracking areas.
4.1.1 Setup

A Vive tracker was attached to a camera tripod and the second tracker was placed on a level table. The table was verified to be level using the digital level function of one of the laser rangefinders. Both trackers were oriented with the positive $z$-axis of their coordinate system (i.e., the flat side of the tracker) pointing directly at the floor. The laser rangefinders were mounted to identical tripods and placed at the edge of the tracking area. The relative positions of the tripods are shown in Figure 4.1.

Figure 4.1: Setup for experiment 1

4.1.2 Procedure

The distance between the rangefinders was measured by pointing one rangefinder at the center of the other rangefinder tripod. The tracker tripod was placed by hand in 30 locations in the tracking area that were randomly chosen while trying to cover as much of the tracking area as possible. In each location, the distance between each rangefinder and the tracker tripod was measured. The average of 5 measurements was taken to account for possible error in aiming the laser rangefinder at the center of the tripod. The distance between the rangefinders ($a$ in Figure 4.1) and the distance between each rangefinder and the tracker ($b$ and $c$) were used to calculate an $x$ and $z$ coordinate using the law of cosines as follows

\[
\alpha = \cos^{-1} \frac{a^2 + b^2 - c^2}{2ab}
\]

\[
x = b \cos \alpha
\]
\[ z = b \sin \alpha \]

The \( x \) and \( z \) values define a 2D coordinate system for the real-world position of the tracker tripod that is parallel to the floor. The height (\( y \) coordinate) of the tripod remained the same throughout the experiment, which allowed the 2D coordinates to be transformed into 3D coordinates simply by setting the \( y \) coordinate to zero.

In each of the 30 trials, 10s of positional data was recorded from the Vive using Unity and the custom logging script and an average position was calculated.

In order for comparisons to be made between the two point sets, the coordinate system of the laser-measured point set was aligned to the Vive-tracked point set. By doing this, any translational or rotational offset between the Vive’s tracking volume and the real-world tracking area was effectively ignored. This alignment is accomplished by multiplying each point in the laser-measured point set, \( L \), by a rotation matrix, \( R \), and adding a translation vector, \( T \). \( R \) and \( T \) minimize the squared positional error between corresponding points in each set and are computed via the method described in Arun et al.\[8\]:

- Find \( \text{centroid}(P_V) \) and \( \text{centroid}(P_L) \), the mean positions of the Vive-tracked and laser-measured points, respectively

- Calculate the 3x3 matrix of the covariances of the positions, where \( P_V^i \) and \( P_L^i \) are individual points in each set

\[
H = \sum_i (P_V^i - \text{centroid}(P_V))(P_L^i - \text{centroid}(P_L))^T
\]

- Perform singular value decomposition on \( H \) to obtain left- and right-singular unitary matrices \( U \) and \( V^T \), where superscript \( T \) represents matrix transposition

\[
H = U \Sigma V^T
\]
• The rotation matrix $R$ is then given by

$$R = VU^T$$

• The translation vector $T$ is given by

$$T = -R^{-1} \text{centroid}(P_L) + \text{centroid}(P_V)$$

• Rotate and translate each point in $L$ using $R^{-1}$ and $T$ to obtain the transformed points $L'$

$$L'_i = R^{-1}P_i - R^{-1} \text{centroid}(P_L) + \text{centroid}(P_V)$$

4.1.3 Results

Figure 4.2 and Figure 4.3 illustrate the Vive-tracked and laser-measured positions as viewed from above (i.e., $y$ coordinates are ignored).

The average distance between the real-world position and the absolute position reported by the Vive is shown in Table 4.1. Relative position of the tripod tracker with respect to the table tracker was computed by subtracting the coordinates of the table tracker from the coordinates of the tripod tracker.

Table 4.1: Error in Vive-tracked positions using laser-measured positions as the "true" position

<table>
<thead>
<tr>
<th></th>
<th>Large area Absolute</th>
<th>Large area Relative</th>
<th>Small area Absolute</th>
<th>Small area Relative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean distance (mm)</td>
<td>7.43</td>
<td>7.56</td>
<td>4.92</td>
<td>5.41</td>
</tr>
<tr>
<td>Max distance (mm)</td>
<td>12.65</td>
<td>14.25</td>
<td>14.98</td>
<td>13.07</td>
</tr>
<tr>
<td>stdev (mm)</td>
<td>2.77</td>
<td>2.81</td>
<td>2.85</td>
<td>2.67</td>
</tr>
</tbody>
</table>

Table 4.1 shows that the error between the expected (laser-measured) position and the position reported by the Vive was less than 1cm on average and was at most 1.5cm.
The difference between the absolute and relative positional error in the larger tracking area was statistically insignificant ($F(1, 58) = 0.034$, $p > .05$). The same was true for the smaller tracking area ($F(1, 58) = 0.471$, $p > .05$).

In the large tracking area, the normal of the regression plane through the Vive-tracked positions was offset from the y-axis of the tracking space by approximately $0.84^\circ$. In other words, the ground plane of the Vive’s tracking space was tilted relative to the real-world floor. In the smaller tracking area, a tilt of $0.25^\circ$ was observed. This observed tilt in the ground plane is consistent with the results found in Niehorster et al. [1].

4.2 Experiment 2

Experiment 2 observed the offset in the tracked position of the Vive HMD and controllers due to an intermittent loss of tracking. This experiment was conducted in
Figure 4.3: Comparison of Vive-tracked positions relative to laser-measured positions in the smaller tracking area.

both tracking areas.

4.2.1 Setup

The Vive HMD was placed on a stool approximately in the middle of the tracking area and oriented such that its orientation was as close to zero as possible (i.e., the HMD was facing the positive $z$-axis). The controllers were placed on the floor next to the stool.

4.2.2 Procedure

The position of the HMD and controllers was recorded for 2s with the logging script. The HMD was covered with a box for 5s, which caused it to lose tracking. The box was removed, and the logging script was run again 5s after tracking was regained to allow time for the tracking to stabilize. The controllers remained in view of the lighthouses during the experiment. This procedure was repeated 20 times in
both the larger and smaller tracking areas.

4.2.3 Results

To determine the amount of variability in the tracked position across the 20 trials, the root-mean-square (RMS) error was calculated using the distance between consecutive positions. RMS error is a measure of the variability of each recorded position about the mean position and gives a sense of the amount of jitter in the tracking system [9]. Absolute position RMS error is calculated using the absolute positions reported by the logging script, and is given by

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i} \text{dist}(P_i, P_{i+1})^2}
\]

where \( N \) is the number of samples and \( \text{dist}(P_i, P_{i+1}) \) is the distance between the positions in consecutive trials.

Absolute RMS error was calculated for the HMD and both controllers. The position of the HMD relative to each controller was used to calculate a relative RMS error for each controller. Relative RMS error is the amount of jitter in the distance between the HMD and each controller. All RMS errors are shown in Table 4.2.

<table>
<thead>
<tr>
<th></th>
<th>RMS error (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Larger area</td>
</tr>
<tr>
<td>HMD</td>
<td>11.48</td>
</tr>
<tr>
<td>Left controller</td>
<td>14.42</td>
</tr>
<tr>
<td>Right controller</td>
<td>9.15</td>
</tr>
<tr>
<td>HMD-Left</td>
<td>4.20</td>
</tr>
<tr>
<td>HMD-Right</td>
<td>4.27</td>
</tr>
</tbody>
</table>

In general, the absolute RMS error was lower in the tracking area for each tracked object. Relative RMS error was much lower than absolute RMS error in the large tracking area, but no conclusion could be made about the smaller tracking area.
4.3 Experiment 3

In Experiment 2, a noticeable drift in the tracked position occurred whenever tracking was lost, but it was not immediately clear whether significant drift would have occurred anyway without an intermittent loss of tracking. Experiment 3 examined the tracked position of the Vive HMD, controllers, and a tracker over a period of 20 minutes to determine if any drift occurs without tracking loss. This experiment was conducted in both the larger and smaller tracking areas.

4.3.1 Setup

The Vive HMD, controllers, and a Vive tracker were placed on a table approximately in the center of the tracking area. The controllers were placed with the trigger facing up on either side of the HMD, and the tracker was attached to a camera mount placed approximately 30cm in front of the HMD. The devices were oriented such that the front of the HMD, the ring of the controllers, and the flat side of the tracker faced the positive $z$-axis.

4.3.2 Procedure

The tracked position of each tracked object was recorded every 10ms for 20 minutes using the logging script.

4.3.3 Results

An average beginning and ending position was taken from the first and last 10s of positional data. The total drift is given by the difference between the ending and starting position. The sample-to-sample RMS error and total drift of each tracked object are given in Table 4.3.

The RMS error of the HMD, controllers, and tracker was less than 1mm, but the total drift varied greatly among the devices. The HMD drifted by only 0.44mm, but the right controller drifted by almost 3mm.
Table 4.3: HMD, controller, and tracker undisturbed RMS and total drift in Large and Small tracking areas

<table>
<thead>
<tr>
<th></th>
<th>RMS (mm)</th>
<th>Total drift (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Large</td>
<td>Small</td>
</tr>
<tr>
<td>HMD</td>
<td>0.16</td>
<td>0.07</td>
</tr>
<tr>
<td>Left controller</td>
<td>0.60</td>
<td>0.16</td>
</tr>
<tr>
<td>Right controller</td>
<td>0.26</td>
<td>0.17</td>
</tr>
<tr>
<td>Tracker</td>
<td>0.10</td>
<td>0.10</td>
</tr>
</tbody>
</table>

4.4 Experiment 4

Experiment 4 measured the position of the tracker tripod using both the Vive and the PhaseSpace system to determine the discrepancy between the positions reported by the two tracking systems. A tripod more suited to be fitted with PhaseSpace LEDs was used instead of the tripod from Experiment 1. This experiment was run twice in only the smaller tracking area due to limitations of the PhaseSpace system used.

4.4.1 Setup

A Vive tracker was mounted to a tripod. An LED controller with eight LEDs and a controller with one LED were used for a total of nine LEDs forming the PhaseSpace constellation. The single LED was attached to the top of the Vive tracker and used as the origin of the constellation. This setup is illustrated in Figure 4.4. The tracking origin of the Vive tracker is the camera mount point on the bottom of the tracker [6], so the height of the tracker (20.4mm) was subtracted from the y value of the positions reported by the PhaseSpace to align the two tracking origins. The xz distance between the two tracking origins was less than 5mm.

4.4.2 Procedure

The tripod was placed in 30 random locations in the small tracking area. The Vive and PhaseSpace positions were recorded at 90Hz and 80Hz, respectively, for 10s.
in each location using the logging script. The two point sets were aligned using the method described in Arun et al. in the same manner as in Experiment 1.

4.4.3 Results

Vive-tracked and PhaseSpace-tracked positions in the $xz$ plane are illustrated in Figure 4.5. Mean distance, maximum distance, and standard deviation between corresponding Vive-tracked and PhaseSpace-tracked positions and the calculated tilt of the ground plane of each tracking space are shown in Table 4.4.

Table 4.4: Vive-tracked positions relative to PhaseSpace-tracked positions

<table>
<thead>
<tr>
<th></th>
<th>Run 1</th>
<th>Run 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean distance (mm)</td>
<td>14.7</td>
<td>19.41</td>
</tr>
<tr>
<td>Max distance (mm)</td>
<td>33.99</td>
<td>37.02</td>
</tr>
<tr>
<td>stdev (mm)</td>
<td>8.55</td>
<td>6.65</td>
</tr>
<tr>
<td>Vive ground tilt (°)</td>
<td>0.44</td>
<td>0.36</td>
</tr>
<tr>
<td>PhaseSpace ground tilt (°)</td>
<td>0.58</td>
<td>0.28</td>
</tr>
</tbody>
</table>

The positional error between the Vive and PhaseSpace was substantially higher
Figure 4.5: Vive-tracked positions relative to PhaseSpace-tracked positions

than the error between the laser rangefinders and the Vive in Experiment 1. The cause is suspected to be the limited number of PhaseSpace cameras available. When the tripod is near the perimeter of the tracking area, some LEDs are out of the field of view of the two cameras required to properly track the LED, which decreases the accuracy of the tracking. This decrease in accuracy can be seen in Figure 4.5.

Figure 4.6 illustrates the tilt of the ground planes of the Vive and PhaseSpace in both runs of the experiment. From the table and plots, it is clear that the amount of tilt in the ground plane of both tracking systems is similar.
Figure 4.6: Ground planes of the Vive and PhaseSpace tracking spaces in (a) the first run and (b) the second run
CHAPTER 5

DISCUSSION

5.1 Experiment 1

The results of Experiment 1 show that even in tracking areas which have a larger inter-lighthouse distance than the recommended maximum of 5.5m, the error between the expected (laser-measured) position and the position reported by the Vive was no more than 1.5cm and was on average less than 1cm. The average error in absolute position was approximately 7.5mm in the larger tracking area and approximately 5mm in the smaller tracking area. The direct relationship between the size of the tracking space and the magnitude of the error is expected to hold for tracking spaces smaller or larger than those used in this study. Based solely on the results of Experiment 1, the tilt in the ground plane appeared to be directly related to the size of the tracking space.

5.2 Experiment 2

The results of Experiment 2 demonstrate a potential method to reduce the drift that occurs when tracking is intermittently lost. In the large tracking area, the RMS error of the HMD and left and right controllers was approximately 11mm, 14mm, and 9mm, respectively. When the position of the HMD relative to each controller was used instead of absolute position, the RMS distance of the HMD dropped to approximately 4.2mm. This suggests that the drift caused by tracking loss could be reduced by using positions relative to a stationary controller or tracker rather than absolute positions. However, results from the small tracking area do not support this
hypothesis. Although the RMS error improved slightly when using relative position
instead of absolute position with the left controller, it worsened with the right con-
troller. The cause of this discrepancy is unclear, but it undermines the efficacy of
using relative positions to reduce RMS error.

5.3 Experiment 3

In Experiment 3, the sample-to-sample RMS of each device was low (less than
1mm). Total drift was also low (< 3mm). However, the amount of drift differed
substantially among the tracked objects.

5.4 Experiment 4

The error observed between the Vive-tracked and PhaseSpace-tracked positions
in Experiment 4 was notably higher than the error between the laser-measured and
Vive-tracked positions in Experiment 1. Average error was approximately 15mm and
20mm for the first and second runs, respectively. For comparison, the error in the
same tracking area in Experiment 1 was approximately 7.5mm.
CHAPTER 6

CONCLUSION

The low cost of the HTC Vive provides an incentive to discover, measure, and attempt to mitigate any flaws in its tracking. The results of Experiment 1 suggest that the positional error in its tracking is modest. This is assuming the tilt in the ground plane is corrected, but as Experiment 4 demonstrated, even some research-grade systems may require this correction.

There are some aspects of its tracking that are not well understood, such as the method by which the origin of the coordinate system is determined. Experiment 2 demonstrated that the offset in the position of the origin following intermittent tracking loss was higher than would be expected from tracking error alone. This is only an issue when tracking loss occurs at some point in an experiment. Attempting to minimize this offset using relative positions yielded mixed results, but there is potential for future research into this issue.


