Healthcare and Medical Devices, Vol. 171, 2025, 1–10
https://doi.org/10.54941/ahfe1006185

The Accuracy of the Eye Tracking in a Virtual Reality Headset for Possible Research and Clinical Applications

Ryan Hall[1], Rui Wu[2], Matthew Joyner[3], Zhen Zhu[3], Brian Sylcott[3], and Chia-Cheng Lin[4]

[1]Department of Computer Science, East Carolina University, Greenville, NC, USA
[2]Department of Information Technology, Kennesaw State University, Marietta, GA, USA
[3]Department of Engineering, East Carolina University, Greenville, NC, USA
[4]Department of Physical Therapy, East Carolina University, Greenville, NC, USA

ABSTRACT

Virtual Reality (VR) headsets are widely used in various clinical and research settings. The reliability and quality of this technology heavily rely on the accuracy of data collected by the VR headsets. The specifications of the VR headsets' accuracy are normally published by the manufacturers. However, the performance claimed by the manufacturers is rarely validated by third-party organizations. Despite its importance, limited research has been focused on the data accuracy of VR headsets, even less so in eye-tracking applications. The main purpose of this project is to invest in the accuracy of eye rotation measurements recorded by the eye-tracking system built into a VR headset. A VR headset with eye-tracking capabilities (HTC Vive™ Pro Eye) was used in this study. For testing purposes, we also developed an eye model that can simulate human eye motion on a controlled pattern with a 30-degree range. The model was used to test the eye-tracking system data collection at frequencies ranging from 10 Hz to 500 Hz, with a step size of 10 Hz. A Unity program was developed to read and export the data, from which the headset's eye tracking accuracy was assessed at each of the tested frequencies. Our experimental results suggested that the VR headset showed great potential for precise eye rotation measurement. Overall, the correlation between the VR headset measurement and the truth reference was between 0.97 and 0.99 from 10 Hz to 500 Hz. The root mean square error (RMSE) values were from 4.39 to 4.74 for the left eye and 3.63 to 3.67 for the right eye, both in degrees. We suspect that the increased RMSE values in the right eye may be due to the relative position between the VR headset and the eye system during testing. Nevertheless, the high correlation between the measurement and truth reference indicates precision in the estimation.

Keywords: Virtual reality, Eye tracking, HTC vive pro eye, Accuracy under different frequencies

INTRODUCTION

Ocular research typically requires a lot of resources, including the cost of the medical devices and other equipment, and the expertise required to operate them. The cost of these devices can exceed tens of thousands of dollars, depending on the specific type and target operations. Consequentially, it becomes a financial challenge for research teams, especially student research projects that require acquiring ocular research data. This paper proposes an alternative method of collecting ocular data based on the HTC Vive headset. This approach is advantageous due to its ease of use and relatively low cost, as much as a tenfold reduction when compared with professional clinical devices.

HTC Vive headsets are commonly used to display Virtual Reality (VR) for a variety of scenarios such as the entertainment industry, scientific simulations, and educational applications. For this project, we used the HTC Vive Pro Eye headset. This version of the headset has eye-tracking capabilities and is officially recommended to collect data at 120 Hz. It is important to note that this device is marketed at a retail price of $1,399 USD; strikingly cheaper than the traditional medical-grade eye-tracking devices. However, at any frequency other than 120 Hz, the accuracy of the data collected is not provided by the manufacturer.

To facilitate data collection from the headset, we developed a graphical application using Unity (Technologies, 2025) in conjunction with the SRanipal Software Development Kit (SDK) (Team, 2025), the official SDK for the Vive headset family. This application acquires data from the headset, stores the values in memory during runtime, and exports the left and right eye data, along with corresponding timestamps, to a .CSV file upon program termination. The sampling frequency can be adjusted by modifying a single variable within the codebase, followed by recompilation.

For testing, plastic eyes were attached to a servo motor wired to an Arduino. The Arduino was programmed to move from 0 degrees to 30 degrees, and then from 0 degrees to –30 degrees, with 10-degree increments, with a waiting interval of 10 seconds. This sequence was used for every test. The Arduino is programmed to wait for certain signals, which were sent through the Arduino Integrated Development Environment (IDE) console (Arduino, 2023). After those signals are received, the sequence begins. The servo motors were then placed on a stand to ensure stability and maximize the replicability and consistency of the data. The goggles were then placed on top of the frame, with the eye positions corresponding to the position of the eyes used to calibrate the headset, and the test was started. The data for these tests were collected at frequencies ranging from 10–500Hz. After the test is stopped, and the .CSV file is created, the RMSE and correlation values are manually calculated for the data collected in each test.

The primary contribution of this paper is addressing the limitations of data collection with the HTC Vive Pro Eye, specifically by investigating the accuracy of sampling frequencies other than the officially supported 120 Hz, which remains insufficiently understood.

RELATED WORK

Prior studies have explored the accuracy of HTC Vive Pro Eye headsets. Here, three such studies are discussed to highlight the differences between prior work and the current work presented in this paper.

Sipatchin et al. investigated the usability of the HTC Vive Pro Eye for clinical ophthalmology (Sipatchin, Wahl & Rifai, 2021), specifically for at-home perimetry testing and visual enhancement applications. Their study assessed eye-tracking accuracy, precision, and temporal latency across different screen regions of the VR headset, as well as under both head-still and head-free conditions. They found that while the central field of vision maintained high accuracy, eye-tracking performance declined significantly at the periphery (±25° from the midline), with increased data loss and decreased precision when subjects moved their heads. Additionally, their latency analysis showed a median system response time of 58.1 milliseconds, suggesting that while the HTC Vive Pro Eye may not be ideal for high-precision perimetry, it could be useful for visual enhancement applications that do not require extreme accuracy. In contrast, our study focuses on the angular accuracy of eye-tracking using a controlled experimental setup with a servo-driven dummy eye model, allowing us to systematically measure accuracy across a broader range of frequencies (10 Hz to 500 Hz). By removing human variability, our approach provides a more controlled evaluation of the device's tracking limitations, particularly regarding manufacturer-stated accuracy at different sampling rates. While both studies aim to assess the capabilities of the HTC Vive Pro Eye, our research contributes to understanding its potential as a low-cost alternative to traditional medical-grade eye-tracking devices.

Schuetz and Fiehler conducted a comprehensive evaluation of the HTC Vive Pro Eye's spatial accuracy, precision, and calibration reliability, providing a real-world benchmark for expected performance in VR-based eye-tracking experiments (Schuetz & Fiehler, 2022). Their study systematically measured accuracy across a 30-degree visual field, testing multiple participants over ten measurement sessions to assess calibration stability. They found that accuracy and precision were highest in central gaze positions but significantly decreased with greater eccentricity. Additionally, they noted that participants wearing glasses experienced reduced tracking performance, whereas those using contact lenses maintained comparable accuracy to those with no vision correction. Their study also revealed minor differences in accuracy between two different Vive Pro Eye headsets, raising concerns about hardware consistency across different devices. These findings contrast with our study, which evaluates eye-tracking performance across a broad frequency range (10 Hz to 500 Hz) using a controlled experimental setup with servo-driven dummy eyes. While Schuetz and Fiehler focused on calibration reliability and real-world participant data, our research eliminates biological variability to provide a controlled accuracy assessment based on systematic gaze direction changes at different tracking frequencies. By combining insights from both studies, we can better understand the HTC Vive Pro Eye's suitability for research applications, particularly in clinical and experimental settings.

Imaoka et al. investigated the feasibility of using the HTC Vive Pro Eye as an assessment tool for saccadic eye movement (Imaoka, Flury & De Bruin, 2020), particularly in the context of neurodegenerative disorders. Their study followed a standardized saccade measurement protocol and tested latency, peak velocity, and error rate in pro- and anti-saccade tasks conducted in a VR environment. The researchers found that the HTC Vive Pro Eye's eye-tracking results aligned closely with those from previous research using traditional eye-tracking systems, indicating its potential for clinical use. However, they noted limitations in time-related measurement parameters, particularly in the accuracy of timestamp data recorded with the SRanipal SDK, which impacted the device's precision in high-temporal-resolution applications. While their study relied on human participants performing natural saccades, our research eliminates biological variability by using a controlled servo-driven dummy eye model, allowing for a systematic evaluation of eye-tracking accuracy across different frequencies (10 Hz to 500 Hz). By comparing findings from both studies, we provide a broader understanding of the HTC Vive Pro Eye's capabilities, demonstrating its potential not just for clinical assessment, but also for high-precision research applications where controlled accuracy testing is required.

PROPOSED METHOD

To start the experiment, a pair of 3D-printed plastic posts were used to hold dummy eyes, which are simply life-sized replicas of human eyes. Not only do the posts hold human-like eyes, but they also elevate them, so that they are within the range of the headset's eye-tracking sensors. The bottoms of these posts allow for attachment to the servo motors as shown in Figure 1. The posts were then placed onto the motors. It was noted, however, that the motors were quite unstable by themselves, as they tended to move whenever a rotation was done. Therefore, it was decided that a motor frame was necessary to maximize the accuracy and the replicability of the experiment. These motors are wired to the Arduino board.

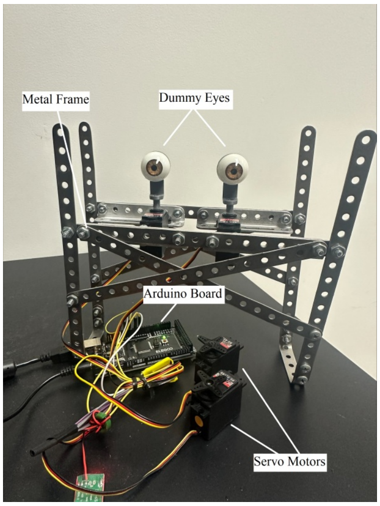

Figure 1: This is the metal frame that stabilized the motors to maximize the replicability of the experiment. The headset is placed on top of this frame and the eyes are used to simulate human eye movements. The distance between the dummy eyes can be modified to fit the eyes of the person who calibrated the headset.

Before the experiment could begin, the servo motors with the fake eyes installed were mounted on the frame. This frame is a group of metal brackets joined with screws. Each metal bracket has a series of holes. There are a number of advantages that this frame provides to the experiment. The first key benefit is adjustability. The user can modify each characteristic of where the eye is placed for the experiment. If the goggles are calibrated to a different eye width setting, then the pupil distance of the dummy eyes can also be modified to match that measurement. A second advantage is that the frame provides the experiment with stability and replicability to ensure the data can be measured as accurately as possible. These plate brackets have several holes to allow various heights and frame widths to be attained. This frame was designed to hold two servo motors with a screw. Four equal-sized angle brackets were screwed onto the frame. These angle brackets are held onto the frame by a screw. Each of these angle brackets features a wide opening to allow for a variety of pupil distances. The servos can slide along these angle brackets. To prevent inaccuracies due to looseness between the screw and the bracket, 4 rubber washers were installed –1 washer on each side of both motors.

To use the headset, it was first necessary to do a calibration. It was not possible to calibrate the headset with the dummy eyes for a number of reasons. The first major reason is that this version of the frame does not support looking upwards. The second factor that prohibits the calibration from being completed with the eyes is the randomization of each test. The calibration requires a number of prerequisite tests: one test that ensures that the height of the goggles is appropriate for the eyes and a second that ensures that the lens distance is correct. Additionally, the calibration process requires the user to look in several random directions. These directions are randomized, and the dummy eyes would have to match their position and speed perfectly for the calibration to be a success. As these tests were intended to be carried out with a human face and eye movement, it was decided that a human would be used to calibrate the headset and the placement on the motor frame would be modified accordingly. After the frame was prepared, and the eye pupil distance matched the eye distance used to calibrate the headset, the headset was ready to be mounted onto the frame. However, the headset is much longer than the frame, making it less stable. To fix this, we stacked several foam chunks on both sides of the frame to balance the headset.

This study used an Arduino ELEGOO (Inc., 2025), which can control motors and power them with a maximum limit of 5 volts. However, the servo motors used are 6-volt servo motors, so a 9-volt power source was used, and a reducer was installed to it so that it could be attached to the 6-volt motors, as Figure 2 shows. The 6-volt motors and their power supply were then wired to the Arduino board. This board enables the attached components to be programmed, allowing them to move according to the code's instructions. The Arduino board sent rotation commands to the motors to rotate the fake eyes. The fake eye rotations are then measured by an HTC Vive headset using a Unity application implemented by us.

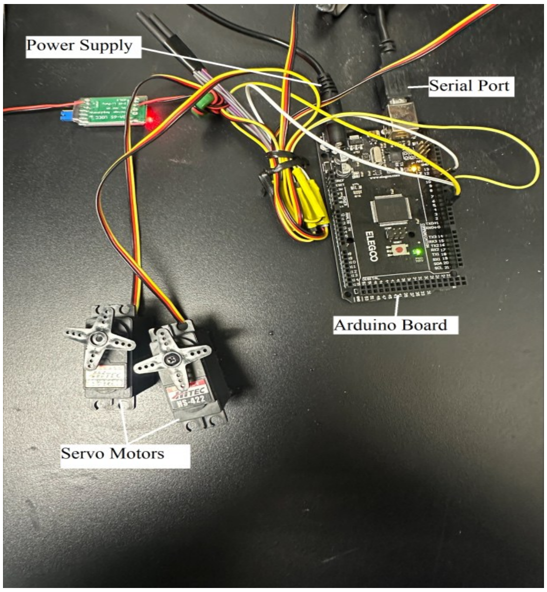

Figure 2: This figure shows the parts that were used for the experiment. Dummy eyes were attached to the servo motors and placed on top of the metal frame. The other parts, such as the board and the wires for the power supply were left underneath.

Unity and SRanipal were used to capture and export the data read from the headset. Unity is a game engine that is commonly used to make both low-cost and commercial games. It allows for a simpler and more efficient way to create games, simulations, and other graphically intensive software. SRanipal is the official SDK distributed by Vive to be used with the HTC Vive headset. It provides functions that allow the values captured from the HTC Vive Pro Eye headset to be quickly accessed in Unity. With these variables accessible, an export function was created to format the data in the .CSV file format. These functions to export data were programmed in C#. Since Unity has a 90 FPS cap, this meant that the data was only capable of being exported at 90 Hz. To fix this, multithreading was necessary. Multithreading is the creation of another process separate from the primary process on which calculations can be carried out, without affecting the speed at which the primary thread runs. After multithreading was implemented, the data was able to be exported at a much higher frequency.

For the experiment, the Arduino board was programmed to rotate the fake eyes from 0 degrees to 30 degrees, then return to 0 degrees, and then rotate from 0 degrees to –30 degrees. One iteration takes 80 seconds. These rotations were done in 10-degree increments with 10-second intervals between each rotation. The Arduino rotations were then measured by the headset, and the data was printed into the .CSV file. After the data was collected the Arduino rotation values were compared to the values that were measured by the headset. Since these values were put into the .CSV format.

After the data was captured from the headset to the .CSV, we created two more rows on the spreadsheet that held the offset values for the rotation degrees. The offset values were used to ensure that the first degree collected will be 0 degrees since the Unity program does not consider the grey point in Figure 3 to be the origin. After adding offset values to our collected fake eye rotation data. The root mean square error (RMSE) and correlation values were calculated for each eye. Since the data from the Arduino is guaranteed to be correct, the rotation will be equal to the sum of the offset values and the degrees collected from the headset. These values will be compared to the degree rotations defined in the Arduino's code for the RMSE and correlation value calculations.

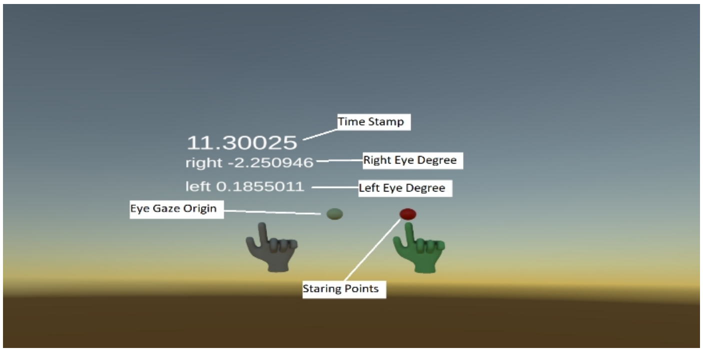

Figure 3: This is a screenshot of our unity application integrated with the SRanipal library. It is used to collect rotation degrees of the left and right fake eyes and show them in real time. The grey ball shows the gaze origin.

EXPERIMENTAL RESULTS AND ANALYSIS

To evaluate the eye-tracking accuracy of HTC headset under different frequencies, the RMSE and correlation values are calculated using Eq. 1 and Eq. 2.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \quad (1)$$

$$Correl(X, Y) = \frac{\Sigma\,(x-\bar{x})\left(y-\bar{y}\right)}{\sqrt{\Sigma\,(x-\bar{x})^2\,\Sigma\left(y-\bar{y}\right)^2}} \quad (2)$$

The RMSE values show the differences between ground truth eye degrees and HTC headset collected degrees. These values were calculated with an offset value for each eye. The offset values are necessary because we used a human head to calibrate the HTC Vive Pro headset instead of the frame and fake eyes. When a human wears the HTC Vive headset, the headset can fit firmly on that person's head. However, when we attached the headset to our frame, there were gaps. Because of this, the offset is generated. The RMSE and Correlation results of both eyes are shown in Figures 4 and 5.

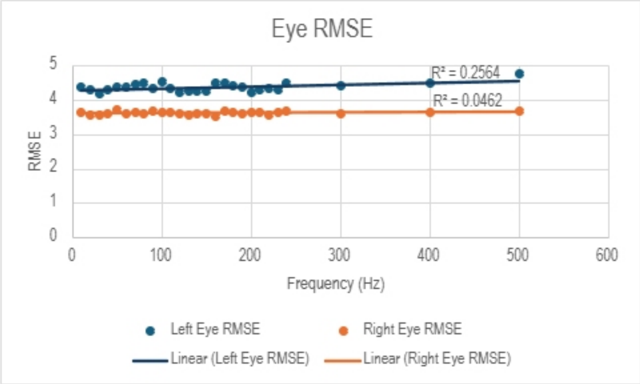

Figure 4: This is a graph showing the RMSE values for both eyes. A regression line is shown.

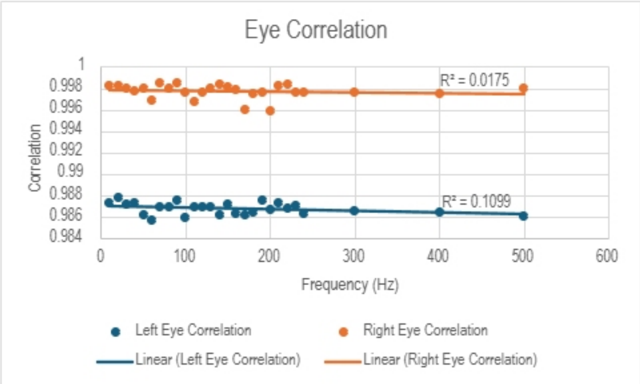

Figure 5: This is a graph showing the correlation values for both eyes. A regression line is shown.

According to the experimental results the left eye RMSE values maintained a similar RMSE value throughout every test, with the left eye showing a higher R^2 value than the right eye. As shown in Figure 4, the left and right eye RMSE values maintained a similar pattern. Figure 5 demonstrates a slightly different level of variation, but the left eye's R^2 value is slightly higher than the right eye's R^2 value.

CONCLUSION

Conducting medical eye research is inherently expensive due to the specialized devices required, which demand expertise for proper operation. Even when such expertise is available, the cost of utilizing professional-grade equipment can be prohibitively high, reaching up to $50,000 USD. We proposed that the Vive Pro Eye headset could be used for the initial screening at the primary care setting since it features eye-tracking capabilities and is marketed at $1,399 USD. The purpose of this paper was to find out the accuracy of the HTC Vive Pro Eye headset under different frequencies. This headset is marketed to capture data at most 120 Hz. However, before the experiment, the accuracy at which the data could be collected at other frequencies was unknown. After the experiment, it was discovered that as the values approached 120 Hz, the RMSE values increased, indicating that they became less accurate. We also found that the correlation values between left and right eye degrees collected by HTC Vive Pro Eye under different frequencies are close to 1. The collected eye rotation degrees are very consistent (i.e., changing with the same trend).

FUTURE WORK

Upon analyzing the results, it was determined that additional variables could have been incorporated to enhance the quality of the data. In this experiment, the headset was mounted on a metal frame equipped with a pair of dummy eyes, and the yaw values of the eyes were recorded. Prior to data collection, the headset was calibrated using human eyes before being repositioned onto the metal frame. Nevertheless, several potential improvements to the experimental setup were identified.

The metal frame that was used for the eye-tracking was beneficial for supporting the eye motor movement, but it was limited to only representing yaw values. The medical eye-tracking device is capable of measuring both pitch and yaw. Therefore, the pitch accuracy of the movement of the HTC Vive Pro Eye headset is unclear. Some further modifications to the dummy eye stand would need to be made to support both pitch and yaw movement. However, that is one of the many ways in which the medal frame could be improved to better suit the experiment. Another flaw in the frame is its lack of support for the headset. If the frame featured a circle or semicircle on the back to simulate a human head, then the headset could be readjusted to better fit the size of the frame. Moreover, if the frame featured more stands, then it would likely be more stable, and the pieces of foam would no longer be necessary.

The calibration of the headset is something that is important and, at times, quite challenging. The quality of the calibration is a determining factor in the accuracy at which the headset returns rotation values. For this experiment, a human was used to calibrate the headset, since the calibration test requires that a series of points be followed with human eyes. If the dummy eyes were used instead of the human eyes for calibration, it would likely return more accurate results. However, this would require a number of prerequisites. For example, pitch support with the dummy eye frame would be necessary. Our team would like to tackle this challenge in the future.

ACKNOWLEDGMENT

This material is based in part upon work supported by: The National Science Foundation under grant number(s) NSF awards #2142428. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

Arduino, 2023, *Arduino Integrated Development Environment (IDE)*.

Imaoka, Y., Flury, A. & Bruin, E. D. De, 2020, 'Assessing saccadic eye movements with head-mounted display virtual reality technology', *Frontiers in Psychiatry*, 11, 572938.

Inc., E., 2025, *ELEGOO Arduino Starter Kits and Components*.

Schuetz, I. & Fiehler, K., 2022, 'Eye tracking in virtual reality: Vive Pro Eye spatial accuracy, precision, and calibration reliability', *Journal of Eye Movement Research*, 15(3), 1–12.

Sipatchin, A., Wahl, S. & Rifai, K., 2021, 'Eye-tracking for clinical ophthalmology with virtual reality (VR): A case study of the HTC Vive Pro Eye's usability', *Healthcare*, 9(2), 180.

Team, V. D. R., 2025, *SRanipal Software Development Kit (SDK)*.

Technologies, U., 2025, *Unity Engine*.