RealNav: Exploring Natural User Interfaces for Locomotion in Video Games

Brian Williamson Chadwick Wingrave Joseph J. LaViola Jr.
University of Central Florida *

ABSTRACT

We present a reality-based locomotion study directly applicable to video game interfaces; specifically, locomotion control of the quarterback in American football. Focusing on American football drives requirements and ecologically grounds the interface tasks of: running down the field, maneuvering in a small area, and evasive gestures such as spinning, jumping, and the "juke."

The locomotion interface is constructed by exploring data interpretation methods on two commodity hardware configurations. The choices represent a comparison between hardware available to video game designers, trading off traditional 3D interface data for greater hardware availability. Configuration one matches traditional 3D interface data, with a commodity head tracker and leg accelerometers for running in place. Configuration two uses a spatially convenient device with a single accelerometer and infrared camera. Data interpretation methods on configuration two use two elementary approaches and a third hybrid approach, making use of the disparate and intermittent input data combined with a Kalman filter. Methods incorporating gyroscopic data are used to further improve the interpretation.

Our results show spatially convenient hardware, currently in many gamers' homes, when properly interpreted can lead to more robust interfaces. We support this by a user evaluation on the metrics of position and orientation accuracy, range and gesture recognition.

KEYWORDS: Wiimotes, locomotion, video games, football

INDEX TERMS: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction Techniques; K.8.0 [Personal Computing]: General Games;

1 INTRODUCTION

Physical locomotion in video games cannot rely on robust traditional 3D commercial trackers due to costs, maintenance and installation issues [21]. As such, tracking the user's natural and realistic motions must be performed with commodity consumer-oriented hardware. Additionally, video game control requires more than just position data that commercial trackers provide, so even these systems present recognition problems for game-specific actions such as jumping or walking in place.

* {bwilliam, cwingrav, jjl} @eecs.ucf.edu

IEEE Symposium on 3D User Interfaces 2010
20-21 March, Waltham, Massachusetts, USA
978-1-4244-6847-8/10/$26.00 ©2010 IEEE

Figure 1 – Reality-based interaction in an American football video game

In this paper, we explore physical locomotion video game interfaces using commodity hardware. Specifically, American football was the video game genre used due to its physical requirements of motion and interactive movement (see Figure 1). To address the needs of football video games with commodity hardware, we used the Natural Point TrackIR device, which provides typical positional and orientation tracking, but is difficult to utilize for all of the requirements in American Football locomotion. As such, this was contrasted with the data of a spatially convenient device [21], the Nintendo Wii Remote (Wiimote), which provides multiple spatial data channels but are partial and intermittent (see Figure 2). Both are commodity hardware designed for use by gamers.

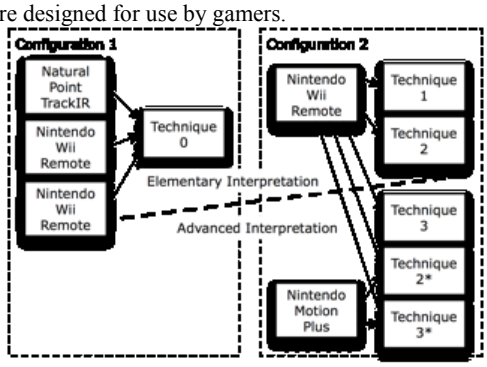

Figure 2 - Two hardware configurations were used to develop multiple data interpretation techniques. Additionally, techniques 2 and 3 were extended to incorporate gyroscopic data.

The Natural Point TrackIR device [12] is a cost effective desktop 6DOF tracker (see Figure 4). It consists of an infrared camera surrounded by infrared lights. The device uses three reflectors, which we placed on a hat for head tracking, to reflect light back to the camera. While other position tracking systems exist [18], the Natural Point is the only device fitting commodity cost requirements.

The Wiimote is the controller to the Nintendo Wii gaming console that can be connected to a PC via Bluetooth and accessed by open source software [13]. The Wiimote fits the definition of a

Authorized licensed use limited to: Javii Austin. Downloaded on May 14,2026 at 17:43:32 UTC from IEEE Xplore. Restrictions apply.

SAMSUNG EXHIBIT 1023
Samsung Electronics Co., Ltd. v. CM HK Limited
IPR2025-01071

spatially convenient device [21] by incorporating its three aspects 1) spatial data: accelerometers (see Figure 5), an infrared camera and gyroscopes, 2) functionality: a rumble pack, speaker, and buttons, and 3) commodity design: wireless transmission, gamer friendly price, a wide distribution to stores, easy setup and high durability.

In traditional American football video games, the quarterback is controlled by a game controller, with a joystick and a complex series of buttons presented for the many options the user performs (see Figure 3). In these games, the player's locomotion tasks include:

- Running - locomotion down the field to score
- Maneuvering - small movements to avoid being tackled
- Evasion - tackler avoidance motions (jump, spin, juke[1]).

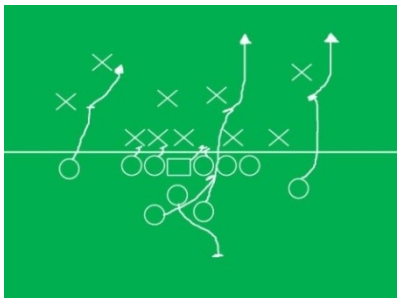

Figure 3 - Above is a typical representation of a football play, where the shapes represent players and lines direct their movements. In these movements, players run while maneuvering around and evading other players.

Ideally, having users perform reality-based interactions [10] instead of using the controller will provide a more enjoyable and, through physical activity, healthier gaming experience. The four tenants of reality-based interaction match gamers' needs for quarterback control: naive physics, body awareness and skills, environmental awareness and skills, and social awareness and skills. This reality based experience is created by exploring two hardware configurations: the Natural Point TrackIR with accelerometers on the legs, and a single Wii remote, designed for minimal encumbrance and fast-passing of control between gamers. We then developed one technique for the first configuration and three for the second and evaluated their performance on American football locomotion tasks.

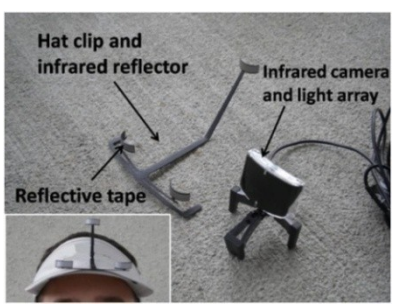

Figure 4 – Natural Point TrackIR infrared tracker

[1]Juking is the act of suddenly switching direction to confuse an opponent.

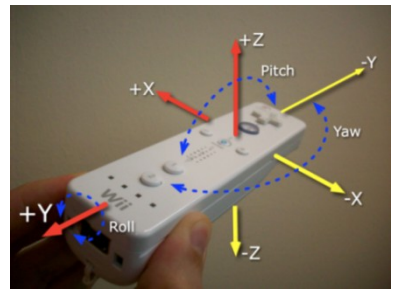

Figure 5 - Wiimote with axes illustrated

In section two, we review literature in the 3DUI field related to travel and locomotion, and recent research with the Wiimote hardware. Section three covers the elementary techniques developed, followed by section four which coveres advanced interpretation methods. Section five is an evaluation of the techniques. Lastly, section six and seven discuss the results, future work and the conclusion.

2 Related Work

Locomotion is defined as the motor component of travel [3]. Several methods of locomotion are possible, such as gaze or wand-directed travel that uses a button or other indication to move the user along a direction. This is in contrast to physical locomotion, where real walking is faster and more precise than walking-in-place, which is faster and more precise than joystick controlled travel [20]. Additionally, physical travel has been shown to incorporate higher-level cognitive understandings of space [22] and walking-in-place was shown to be an effective alternative to real walking for maintaining presence [17]. Real walking has also been augmented in [8] to cover larger distances. Limitations of the physical approaches to locomotion are the range of tracking systems, room size limitations, and limitations of recognizing walking-in-place gestures.

Real walking can generally be tracked in two ways: position tracking and user instrumentation. Traditional tracking systems can detect the user's position change to determine their walking. This can be through outside-looking-in [5] and inside-looking-out [19] approaches. Instrumentation of the user can be through accelerometers or other sensors attached directly to the user's feet or body, in this way, there are no environmental limitations. While previous methods [5] [19] have been successful in tracking a user's position, the hardware used is not spatially convenient or commodity based hardware.

The Wiimote's use in 3D user interface research is becoming common for novel and universal interaction tasks [11] [15] [21]. In [16], two Wiimotes were used to control the animation of a virtual character through arm, hand and leg movements, which had a similar design to our configurations, but did not attempt to track the actual position of the user. For navigation purposes the Wiimote was used for pointing and selection in a multi-wall virtual reality theatre in [14] and in [6], the Wiimote was used for navigation of three-dimensional MRI medical data. These showed the capabilities of the device, but not for locomotion problems. It has also been used for gesture-based musical instrument control [2] and dance-based physical game control [4] which illustrates the device as mapping human movement to actions, but none attempting to map the user's location to the virtual world. The Wiimote continues to show itself more than a wireless button device and more than just the sum of its parts. To the best of our knowledge this work presents the first study on creating natural locomotion interfaces in football video games using spatially convenient hardware that is already available in the home.

Authorized licensed use limited to: Javii Austin. Downloaded on May 14,2026 at 17:43:32 UTC from IEEE Xplore. Restrictions apply.

3 ELEMENTARY INTERPRETATION TECHNIQUES

Elementary data interpretation methods were used on the two hardware configurations discussed in the introduction. For configuration one, its interpretation resulted in technique zero and, because of its match to traditional 3D interface position and orientation data, it set the standard for the following techniques. Technique one interpreted acceleration data for the locomotion tasks while technique two interpreted the infrared camera data.

3.1 Technique Zero: Traditional 6DOF tracking

Because hardware configuration one provides 6DOF pose tracking and acceleration data for each leg, it was expected to be the most accurate under ideal conditions. For the maneuvering task, the position data from the TrackIR API was matched to the virtual camera's position. The orientation data was noisy and disruptive to the user, so we applied an alpha-beta filter, defined as

$$X_n = (\beta * X_{n-1}) + \left(\alpha * \dot{X}_n\right) \tag{1}$$

$$\alpha + \beta = 1,$$

where n is the current state in time, X is the prediction from the alpha-beta filter, and $\dot{X}$ is the current measurement.

The running task mapped acceleration data from the two Wiimotes on the user's legs to movement. Acceleration peaks over a threshold indicated users wanted to travel forward. For the evasion tasks, infrared position tracking was used, with jumping recognized by an elevated vertical position and spinning recognized by rapid yaw changes. Lastly, the juke looked for lateral acceleration of the user's legs.

This resulted in a technique that was sufficient for all the American football locomotion tasks we chose to examine. However, it involved hardware not designed for console gaming systems and not widely found in gamers' homes, multiple Wiimotes and Wiimotes strapped to a gamer's legs, creating issues of encumbrance and for quickly passing of control between gamers.

3.2 Technique One: Acceleration Only Method

Technique one uses only the accelerometer data. In this way the user does not have to look at or near their sensor bar, retaining their full movement capability and is only limited by the range of the Wiimote's Bluetooth connection. Two design iterations explored this technique.

The first iteration analyzed the signal from the accelerations in order to determine the direction the user was moving. The reported accelerometer data would show the direction the user initially moved in, followed by noise as they continued stepping, and then stabilization when they stopped. When the system saw this pattern, the virtual world moved in the first detected direction until the stabilization was seen. Unfortunately, it suffered from latency as it gathered data about the user's motions, it could not move in the exact direction of the user and stopping was required to switch direction because stabilization needed to be observed. None of this was acceptable for the maneuvering task.

The second iteration was the more simple solution of double integration of the acceleration in order to determine position. The resulting formulas from this integration are

$$\hat{X}_n = \hat{X}_{n-1} + \hat{V}_n * t + \frac{1}{2}\hat{A}_n * t^2 \tag{2}$$

$$\hat{V}_n = \hat{V}_{n-1} + \hat{A}_n * t$$

where $\hat{X}_n$ is the current position with $\hat{X}_{n-1}$ being the previous position, $\hat{V}_n$ is the velocity, $\hat{A}_n$ is the accelerometer reading, and t is the change in time since the last update.

This worked well under two major assumptions; that the data contained little noise and the gravity vector was completely removed. For noise removal, an alpha beta filter (see Equation 1) was used which introduced minor latency, but with more accuracy. For the gravity vector, its orientation was updated every time the reported accelerometer data hovered near 1.0.

Running in place was determined when upward acceleration beyond a threshold was seen at a frequent rate. This placed the system into a state in which other gestures were ignored as the user moved forward. While this gesture had latency, it was only at the start when it looked for a steady upward movement.

For the evasive gestures, recognition looked for heuristics in the accelerometer data, but since the data was already being used for maneuvering and running in place, disambiguation between gestures became a problem. While possible, recognition accuracy was too low for gaming requirements.

3.2.1 Results

Regarding maneuvering tasks, a second design iteration was required. However, it too failed as it could not disambiguate between all the evasive tasks. As such, a new technique was explored.

3.3 Technique Two: Head Tracking Method

Technique two was designed around infrared maneuvering and accelerometer gestures. For this, the Wiimote was mounted to a hat with Velcro as seen in Figure 6. The Wiimote's sensor bar emits two IR points at a fixed width for the camera to see, forming the sensor bar connection (SBC). From this, the Wiimote API returns the X,Y camera coordinates for both points and we calculate a midpoint from which horizontal and vertical translation are identified. Depth is also determinable using the SBC as the distance between them relates to the distance between the Wiimote and sensor bar. This is possible as the sensor bar IR emitters are assumed to be fixed and the user stationed roughly perpendicular (see [21] for handling non-perpendicular cases). For simplicity, as the user approaches the sensor bar and the points grow further apart at a non-linear rate, the square root of the distance is mapped to the user's depth.

Figure 6 - A Wiimote attached to a cap allows for easy passing of control between players and unencumbered head tracking in configuration 2.

The other tasks were handled with a mix of infrared and accelerometer data. For the running task, accelerometer data aligned with the gravity vector was used to identify running in place. With the evasive tasks, jumping was distinguished by seeing a large increase in the standard deviation of the vertical position. Lateral acceleration over a threshold indicated juking. Finally, spinning relied on the standard deviation of the horizontal position of the infrared data.

Authorized licensed use limited to: Javii Austin. Downloaded on May 14,2026 at 17:43:32 UTC from IEEE Xplore. Restrictions apply.

3.3.1 Results

The translation and depth calculations were shown to be an accurate solution for maneuvering when the infrared connection was made. The evasive gestures and running in place were also successfully recognized.

3.4 Elementary Techniques Discussion

Technique zero was found to be acceptable but required encumbering hardware and hardware not currently found in gamers' homes. By using elementary data interpretation, technique one was created but was not acceptable for all the tasks we required. Technique two was as acceptable as technique zero, and without its hardware issues, but does not provide orientation tracking. As the American football domain requires the ability to look around the field, limiting technique two, and realistic movements, limiting techniques zero and two which require infrared connections, no technique was found to be sufficient under all constraints.

4 Advanced Interpretation

The elementary interpretation methods treated all channels as distinct data. Advanced techniques merge these channels in a hybrid approach, allowing for weaknesses in the elementary approaches to be resolved. Additionally, a gyroscope is added in later techniques, which introduces a new data channel while remaining unobtrusive.

4.1 Technique Three: Hybrid

The first hybrid attempt selected infrared data when available and failing that, accelerometer data. While this noticeably improved upon the earlier techniques, it still operated on distinct data channels. To merge these data channels, a Kalman filter was applied to produce an optimal position prediction. With this, not only does the technique fail gracefully when the infrared data is not present, but the infrared tracking is improved by the acceleration data. Regarding gesture tracking, technique 3 reused technique 2's successful approach.

Because of its similarities to technique two, the same gesture recognition class was used to resolve both running in place and the evasive gestures. Also, a version implementing the Wii Motion Plus was created using the exact same methods above.

4.1.1 Position/Velocity/Acceleration Kalman Filter

We used a Kalman filter based on [1]. The state vector is

$$X_t = \begin{bmatrix} X \\ \dot{X} \\ \ddot{X} \end{bmatrix} \tag{3}$$

where X is the position, $\dot{X}$ is the velocity, and $\ddot{X}$ is the acceleration of the system and the time update step is defined as

$$\begin{aligned} \hat{X}_t &= X_{t-1} * A_{\Delta t} \\ \hat{P}_t &= A_{\Delta t} * P_{t-1} * {A_{\Delta t}}^T + E \end{aligned} \tag{4}$$

where $\hat{P}_t$ is the predicted covariance matrix, coming from the last state P_{t-1}. E is a constant error matrix for the prediction step and the fundamental matrix is

$$A_{\Delta t} = \begin{bmatrix} 1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

where Δt is the change in time since the last time update step was called.

The next step is to perform the measurement update with a measurement vector of

$$Z_t = \begin{bmatrix} X \\ \ddot{X} \end{bmatrix} \tag{6}$$

where X is the position and $\ddot{X}$ is the accelerometer measurements and the measurement update step is defined as

$$\begin{aligned} K_t &= \hat{P}_t * H^T * \left(H * \hat{P}_t * H^T + R\right)^{-1} \\ X_t &= \hat{X}_t + K_t * \left(Z_t - H * \hat{X}_t\right) \\ P_t &= (I - K_t * H) * \hat{P}_t \end{aligned} \tag{7}$$

where K_t is defined as the optimal Kalman gain, H is the observation matrix, and R is an error covariance matrix representing noise from the devices. The covariance matrices and initial values were based on the data observed in [1] and adjusted slighted based on manual optimization.

There was a predictor step added in from Azuma's dissertation based on predicting the state based on previous data when measurements are unavailable. This is the time update step, except no states are actually changed within the filter. It was

$$X_p = X_t + \dot{X}_t * (p - t) + \frac{1}{2} * \ddot{X}_t * (p - t)^2 \tag{8}$$

where pp is the point of time in which a prediction is made, tt is last point in time in which the Kalman filter's time update and measurement update steps were performed, X is the position, $\dot{X}$ is the velocity, and $\ddot{X}$ is the accelerations from the state vector.

4.1.2 Results

We found technique three to be as capable at maneuvering, running and evasive gestures, if not better, than techniques one and two. This was possible even when no infrared connection was available. As such, we felt this was the best match for locomotion control in American football gaming.

4.2 Integration of the Wii Motion Plus

The Wii Motion Plus is a recent addition to the Wiimote which augments the device with a MEMs gyroscope and still remains commodity priced. The hardware produces angular rate data, but gyroscopes are known for problematic drift. Thus, we chose to fuse these predictions using an extended Kalman filter (EKF) found in [1].

First, the angular rates were read from the Motion Plus. Next, if the Wiimote's accelerometer data was relatively stationary pitch and roll were calculated from the gravity vector using pseudo code and is

$$\begin{aligned} roll &= \arctan2\left(a_x, sqrt\left({a_y}^2 + {a_z}^2\right)\right) \\ pitch &= \arctan2\left(a_y, sqrt\left({a_x}^2 + {a_z}^2\right)\right) \end{aligned} \tag{9}$$

where a_x,a_y, and a_z are the accelerometer data along each axis. This created an equation reliable against singularity and other rotational problems, but made pitch only move from zero and ninety degrees. For this reason, the correction

Authorized licensed use limited to: Javii Austin. Downloaded on May 14,2026 at 17:43:32 UTC from IEEE Xplore. Restrictions apply.

$$f(pitch) = \begin{cases} pitch & , a_z \geq 0 \\ pitch + \left(\frac{\pi}{2} - pitch\right) * 2, & a_z < 0 \end{cases} \quad (10)$$

was applied. Where a_z is the accelerometer data along the Z axis.

This was a heuristic approach that if the Z axis of the Wiimote accelerometers were reporting the device being upside down, the angle was corrected to produce a correct value.

The values from (9) and (10) were then sent as measurement updates with the angular rates to the extended Kalman filter discussed below. With yaw corrections a heuristic approach was taken based on the horizontal position of the infrared data. The angle of the user's head could be predicted based on the horizontal displacement of the infrared. With this mapping we were able to correct yaw drift.

Also, the motion plus data had a threshold value created to signal to the system if the user was rotating their head or not. This was done to resolve any ambiguities in why the infrared data may be moving, whether the user is physically moving horizontally or vertically, or if they are rotating their head. If rotating, camera movement was ignored; otherwise, it took the normal effect described above.

Finally, the Motion Plus data proved valuable in recognizing the spin gesture in the evasion task. If the yaw angular rate had a sudden and consistent increase, the user was considered to be spinning and was flagged as such.

4.2.1 Extended Kalman Filter

Non-linear equations used for determining orientation from the Wii Motion Plus required an extended Kalman filter. The equations (4) and (7) were used in the EKF, but the state vectors, fundamental matrix, and other variables were re-defined for this filter.

Also, a quaternion was used to represent the orientation of the Wiimote with the first value being the scalable real number and the following three representing the imaginary vector. Furthermore the angular velocity and angular acceleration were tracked in the filter. This is

$$X_t = [Qw \quad Qx \quad Qy \quad Qz \quad \omega 0 \quad \omega 1 \quad \omega 2 \quad \dot{\omega 0} \quad \dot{\omega 1} \quad \dot{\omega 2}]^T \quad (11)$$

where Qw, Qx, Qy and Qz represent the quaternion state, $\omega 0, \omega 1$ and $\omega 2$ are the angular velocity, and $\dot{\omega 0}, \dot{\omega 1}$ and $\dot{\omega 2}$ are the angular acceleration. The process model is

$$\int_{t-1}^{t} \frac{1}{2} * (Q * \omega) dt \quad (12)$$

where Q is the quaternion, ω is the angular velocity and a quaternion multiplication occurs between them with the scalar component of ω being set to zero. We then use a Jacobian matrix to linearize the fundamental matrix[2] for use in the EKF.

The measurement taken from the system was assumed to know the orientation to some degree and contain the angular velocities. This resulted in a transition matrix H that only took out the angular acceleration data from the state vector. The measurement state is

$$Z_t = [Qw \quad Qx \quad Qy \quad Qz \quad \omega 0 \quad \omega 1 \quad \omega 2]^T \quad (13)$$

where Qw, Qx, Qy and Qz represent the quaternion state measured and $\omega 0, \omega 1$ and $\omega 2$ are the angular velocity measured.

Furthermore we used Azuma's EKF prediction formulation step [1] when good measurements were not available, defined as

$$Q_P = \left[I cos(d) + \frac{M(t_c)}{d} \sin(d) \right] Q_{tc} \quad (14)$$

where Q_P is the predicted quaternion state, Q_{tc} is the quaternion state of the last successful measurement update, d is the length of the integral across the angular velocity vector, and $M(t_c)$ is the matrix containing the quaternion multiplication represented in (12).

This can calculate a prediction quaternion Q at time P given a valid time update and measurement update steps performed on the extended Kalman Filter at time T.

4.2.2 Results

The Motion Plus was integrated to determine the user's orientation with corrections in place to avoid drift. Since this was combined with the user's head, and required infrared for corrections, it was only integrated into techniques two and three. This produced what will be referenced as technique 2* and technique 3*. These modified techniques improved gesture recognition of the spin gesture by utilizing angular rate data on the yaw axis.

5 EVALUATION

Two approaches were used to gauge the effectiveness of these techniques: performance evaluation against metrics and a formative user study evaluation.

5.1 Performance Evaluation

Four metrics were chosen to evaluate the techniques: accuracy, tracking range, gesture recognition and orientation. The rough measures used in these evaluations are purposeful, remaining grounded in video game tasks.

5.1.1 Accuracy

Accuracy was defined as the ability for the system to return to zero after the user moved and returned to the starting position. For each technique involving tracking the user's position (this excludes 2* and 3*), we performed fifteen movements either back, left, right, forward or in a complex manner such that the infrared connection was lost. Button presses at the start and end defined the movements. The measure was the magnitude of the positional difference in yards (the measurement unit in American football). As shown in Figure 7, the average accuracy is relatively consistent for every technique except the first. While no technique touched exactly zero, the minor difference in yards is acceptable given the rough unit of measure and the gaming task.

[2] Due to space concerns the fundamental matrix is too large to illustrate, but can be derived by taking the Jacobian matrix of (12).

Authorized licensed use limited to: Javii Austin. Downloaded on May 14,2026 at 17:43:32 UTC from IEEE Xplore. Restrictions apply.

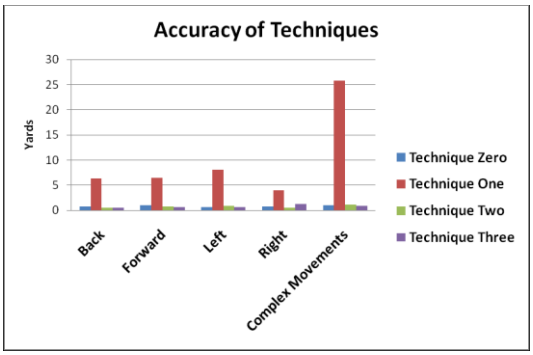

Figure 7 - After user movements, all but technique one was accurate in returning to the starting position.

5.1.2 Tracking Range

Both hardware configurations had infrared tracking that was limited to line-of-sight, but the Wiimote's accelerometer and gyroscope were only limited by Bluetooth's wireless range. The Bluetooth range is quite sufficient for gaming needs, where the gamer would lose the ability to readily see the screen before they lost the Bluetooth connection. Comparing the infrared tracking, the Wiimote's range is much larger than TrackIR, with depths up to fifteen feet and allowing lateral movements between three to four feet from center of the screen. In contrast, the TrackIR was designed for seated desktop users and lost its connection within the range needed by physical gamers. Its range was only five feet in depth and three feet in lateral movement from the center.

5.1.3 Gesture Recognition Accuracy

Evasive gesture recognition with and without the Motion Plus was compared across twenty-five trials for each gesture. Our results indicate that without the Motion Plus gestures were recognized correctly 71%, and with the Motion Plus the accuracy was increased to 93%. This is consistent with the improvements seen in [7] and approaching acceptability for some gaming needs.

5.1.4 Yaw Orientation Accuracy

Technique 3* provided yaw orientation data with the Motion Plus. The Wiimote's infrared connection compensated for drift but this connection can be intermittent due to movement. To measure orientation accuracy without the infrared update, an Intersense IS-900 tracker [9] was placed on the Wiimote for a ground truth measurement. The figures below represent full yaw revolutions with the Wiimote drifting up to ninety degrees off of the truth data when the infrared is not present. When present, some drift may occur, but it self corrects (see Figure 9). The average root mean square RMSE without infrared is 121.09 degrees versus a RMSE of 16.33 degrees.

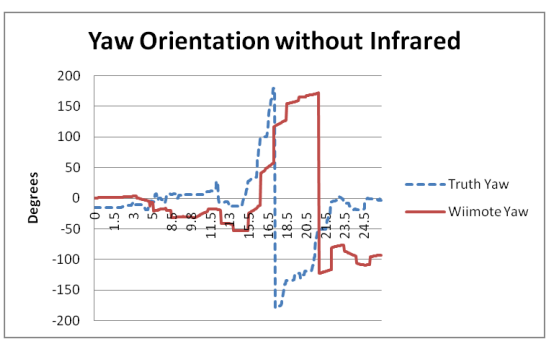

Figure 8 - Without infrared the Motion Plus exhibits substantial drift on the yaw axis

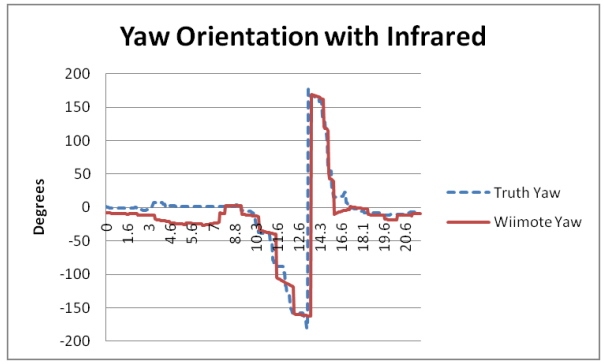

Figure 9 - With infrared the Motion Plus has drift corrected and is suitable for determining yaw orientation

5.2 User Evaluation

A user study was performed with two aims: determine if the advanced interpretation techniques are preferred over elementary to participants and determine if the ability to look around during locomotion, termed orientation-controlled gaze, is preferred.

5.2.1 Participants and Apparatus

Ten participants were recruited for the study through word of mouth. There were six males and four females with a mean age of 25 and in a range of 19-28. In the study, they were facing a large screen 50" Samsung DLP 3D HDTV with a refresh rate of 120 Hz and a resolution of 1920 x 1080, as seen in Figure 1. Additionally, they wore a hat with the Wiimote attached (see Figure 6). The Motion Plus was added or removed as needed.

5.2.2 Experimental Task

Two tasks tested the maneuvering and gesturing interfaces. The first task, the maneuvering task, involved a small droid moving randomly on the screen; firing spherical shots in the participant's direction (see Figure 10). The participants had to dodge these using the techniques.

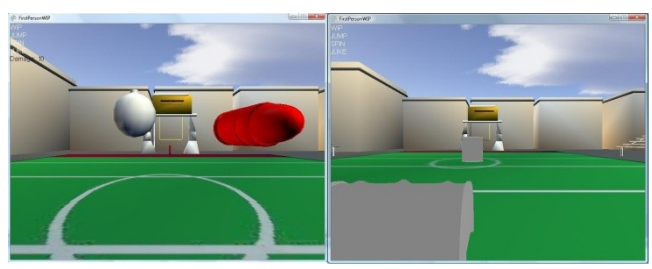

Figure 10 - The two tasks for participants were to dodge projectiles (left) and practice evasive tasks while traveling down the field (right).

The second task, the evasive task, involved locomotion and evasive gestures by having the participant run in place to move down the field and overcoming several obstacles in their path (see Figure 10). Obstacle one was a barrel on its side that they had to jump. Obstacle two was a barrel standing up which they had to run up to and spin around. Obstacle three was an opponent that appeared and pushed them back until they performed the juking gesture. Afterwards, they ran to the end zone completing the trial.

5.2.3 Experimental Design and Procedure

The participants performed the four conditions in random order. These conditions were technique 2, orientation-controlled technique 2 (technique 2*), technique 3 and orientation-controlled

Authorized licensed use limited to: Javii Austin. Downloaded on May 14,2026 at 17:43:32 UTC from IEEE Xplore. Restrictions apply.

technique 3 (technique 3*). For each task, participants performed a practice trial, followed by two live trials. The maneuvering task trial was performed for 30 seconds while the evasive task, which had no time limit by nature, roughly took participants over 15 seconds.

Fatigue was a significant factor in these trials, especially for the evasive task that involved running down the field. As such, the conditions were limited to two trials and participants were asked to take breaks with mandatory breaks after each condition.

5.2.4 Performance Results

Quantitative performance data was obtained during the experiment but was not the object of the study. The effects of fatigue on performance further make these results less useful to our study of participant preference. For the maneuvering task, the system recorded how many times a participant was hit by projectiles in each trial with an average participant being hit 3.3 times and ranging from 0 to 10 hits. With the evasive gesture task, the quantitative measure was time to travel down the field. This was on average 15.85 and ranged from 9.88 to 33.6 seconds. As this was a formative evaluation, statistical significance in the results was not the goal. However an ANOVA analysis was performed and we found no significant differences between conditions for the maneuvering task ($F(3,7)=1.04$, $p = 0.39$) and the evasive task ($F(3,7)=1.41$, $p = 0.26$).

5.2.5 Subjective Results

After completing all four conditions, a questionnaire was administered. Participants were asked: to pick a favorite technique for both the maneuvering and evasion tasks, to say if they liked the orientation-controlled version of each technique, and to explain why.

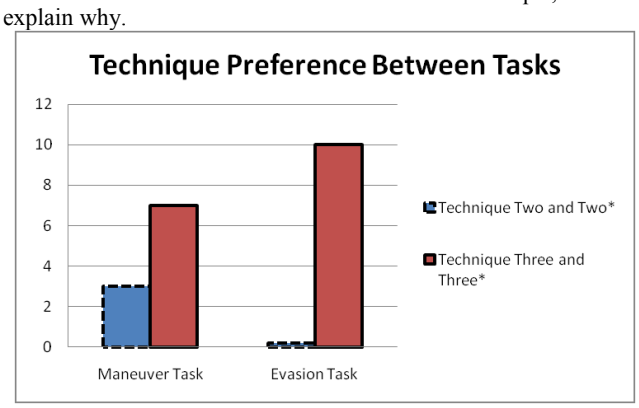

Figure 11 – Participants prefer technique three and three* between the maneuvering and evasion task

Figure 11 shows that 7 of the 10 participants preferred technique three to technique two for the maneuvering task and all preferred it for the evasive gesture task. Participant's oral comments were that they could feel the constraints of the infrared bounds while using technique two and written comments stated they felt technique three to be the "smoothest and had the largest range of motion".

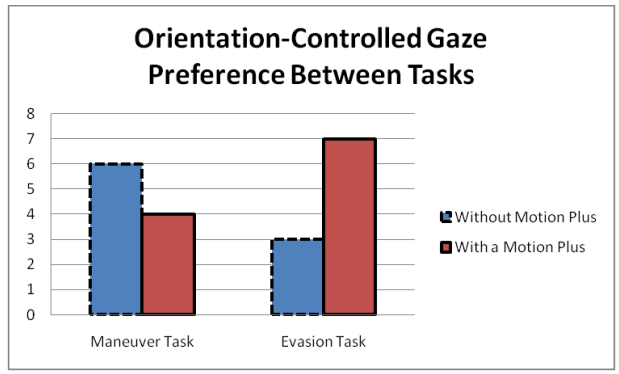

Figure 12 - Orientation-controlled gaze preference was mixed between tasks but participants seemed to prefer it when their task included gesture recognition

Regarding orientation-controlled gaze, Figure 12 shows that for maneuvering, only 4 of the 10 participants preferred the orientation-controlled gaze. Of those that did not, their comments were that the head movement was distracting since it did not relate to the maneuvering task. However, 7 of the 10 participants preferred orientation-controlled gaze for the evasive gestures task. This was a result of better gesture recognition due to hardware improvements, rather than a factor attributed to the orientation-controlled gaze. Users made oral comments that they found the gestures easier to perform when orientation-controlled gaze was a factor and written comments that it "picked up jumping better" and "felt more precise", but as section 5.1.3 shows; this is a result of better interpretation made possible by additional hardware.

5.2.6 General Observations

Several other comments were made by the participants, such as verbally expressing that juking past an opponent was fun and a greatly enjoyed evasion task. We also asked about using the system to complement an exercise routine with eight users expressing interest in doing so. Lastly, we asked users if they would want to replace a traditional controller with a 3D user interface, with seven out of ten saying they would. Some of this feedback included liking, "not having to hold a controller" and that it allowed players to, "get more involved and active." Some other suggestions included a user saying the types of games they liked, "do not seem to go over well with a 3D user interface" such as strategy and RPG games. Another user suggested that they only play video games, "as a way to relax and do not find physical activity conducive to that."

6 Discussion

Below, we discuss the three iteratively designed techniques and their analysis.

6.1 Techniques Contrasted

Techniques three and three* came out as the best for meeting the tasks with performance comparable to technique zero. It also was preferred over technique two because two had limited functionality due to the infrared tracking bounds. Users made oral comments about feeling constrained to a box, while technique three allowed more freedom of movement. As for technique one, it allowed the user to simply pick up the Wiimote and play short interactive bursts, typical of Wii games, but did not have the accuracy needed for our domain of American football video games.

Authorized licensed use limited to: Javii Austin. Downloaded on May 14,2026 at 17:43:32 UTC from IEEE Xplore. Restrictions apply.

6.2 Orientation-Controlled Gaze

The Motion Plus's orientation accuracy, compared against the truth data, shows that it can be relatively accurate, especially with an infrared connection to compensate for drift. There is still some jitter. Lastly, techniques two and three's gesture recognition was improved.

6.3 Future Work

Based on our experiments there are several avenues for future work. First, there are improved methods for interpreting the Motion Plus data including determining the yaw angle from the infrared connection [21]. Second, control-input models can be added to the Kalman filter for better use of orientation in gaze-control. Third, the evasive gestures were mostly recognized heuristically and more sophisticated methods, such as Rubine's algorithm used in [7] can be integrated. Fourth, the system's physical activity and gaming aspects could motivate exercise. Fifth, locomotion is only one aspect of American football and the quarterback is only one position. Other work can focus on jumping to catch a ball, blocking, or throwing a pass using head orientation and arm movements. This is non-trivial as each additional gesture complicates the gesture classifier. Lastly, this work is formative in nature, but once a complete football interface has been built, a summative evaluation against an existing American football game interfaces can be performed.

7 Conclusion

We presented an exploration into natural locomotion interfaces with specific interest in American football video games. We developed a set of techniques utilizing a single Wiimote compared to a traditional 6DOF tracker and Wiimote configuration. We discovered that with a Kalman filter and the Motion Plus hardware, a single Wiimote was able to perform as well as technique zero showing that we can obtain comparable results using tracking devices that are not traditional position and orientation trackers. While robustness improvements exist, our work is a good starting point for these types of natural locomotion in video games using hardware already existing in gamers' homes.

8 Acknowledgements

This work is supported in part by NSF CAREER award IIS-0845921 and NSF Award IIS-0856045. We wish to thank the anonymous reviewers for their valuable suggestions.

References

[1] Azuma, R. (1995, February). Predictive Tracking for Augmented Reality. Chapel Hill, North Carolina, United States of America: University of North Carolina.

[2] Bott, J., Crowley, J., & LaViola, J. (2009). Exploring 3D Gestural Interfaces for Music Creation in Video Games. *Fourth International Conference on the Foundations of Digital Games*, (pp. 18-25).

[3] Bowman, D., Kruijff, E., LaViola, J., & Poupyrev, I. (2005). *3D User Interfaces: Theory and Practice.* Addison-Wesley.

[4] Charbonneau, E., Miller, A., Wingrave, C., & LaViola, J. (2009). Understanding visual interfaces for the next generation of dance-based rhythm video games. *ACM Siggraph Video Game Symposium* (pp. 119-126). New Orleans: ACM.

[5] Foxlin, E. (2002). Motion Tracking Requirements and Technologies. In K. Stanney, *Handbook of Virtual Environments: Design, Implementation, and Applications* (pp. 163-210). Lawrence Erlbaum Associates.

[6] Gallo, L., De Pietro, G., & Marra, I. (2008). 3D interaction with volumetric medical data: experiencing the Wiimote. *1st annual conference on Ambient media and systems.* ICST.

[7] Hoffman, M., Varcholik, P., & LaViola, J. (2010). Breaking the Status Quo: Improving 3D Gesture Recognition with Spatially Convenient Input Devices. *To be appear in IEEE VR.*

[8] Interrante, V., Ries, B., & Anderson, L. (2007). Seven League Boots: A new Metaphor for Augmented Locomotion through Moderately Large Scale Immersive Virtual Environments. *Symposium of 3D User Interfaces* .

[9] Intersense Inc. (2009). *Intersense Inc - IS-900 systems*. Retrieved 10 16, 2009, from Intersense Inc - Precise Motion Tracking Solutions: http://www.intersense.com/IS-900_Systems.aspx?

[10] Jacob, R., A, G., Hirshfield, L., Horn, M., Shaer, O., Solovey, E., et al. (2008). Reality-Based Interaction: A Framework for Post-WIMP interfaces. *CHI* (pp. 201-210). New York, NY: ACM.

[11] Lee, J. (2008). Hacking the Nintendo Wii Remote. *IEEE Pervasive Computing , 7* (3), 39-45.

[12] Natural Point, I. (2009). *TrackIR :: head tracking view control immersion for flight racing and action simulator :: TrackIR :: Premium head tracking for gaming*. Retrieved September 20, 2009, from NaturalPoint :: Optical Tracking Cameras :: Motion Capture Solutions :: Hands Free Ergonomic Mouse Alternatives: http://www.naturalpoint.com/trackir

[13] Peek, B. (2008, June 7). *WiimoteLib -.Net Managed Library for Nintendo Wii Remote*. Retrieved September 20, 2009, from Brian's Blog - BrianPeek.com: http://www.brianpeek.com/blog/pages/wiimotelib.aspx

[14] Schou, T., & Gardner, H. (2007). A Wii remote, a game engine, five sensor bars and a virtual reality theatre. *19th Australian Conference on Computer-Human Interaction: Entertaining User Interfaces* (pp. 231-234). Adelaide: ACM.

[15] Shirai, A., Geslin, E., & Richir, S. (2007). WiiMedia: motion analysis methods and applications using a consumer video game controller. *SIGGRAPH Sandbox.*

[16] Shiratori, T., & Hodgins, J. (2008). Accelerometer based User Interfaces for the Control of a Physically Simulated Character. *ACM Transactions on Graphics , 27* (5).

[17] Slater, M., Usoh, M., & Steed, A. (1995). Taking Steps: The Influence of a Walking Technique on Presence in Virtual Reality. *ACM Transactions on Computer-Human Interaction , 2* (3), 201-219.

[18] Welch, G., & Foxlin, E. (2002). Motion Tracking: No Silver Bullet, but a Respectable Arsenal. *Computer Graphics and Applications* , 24-38.

[19] Welch, G., Vicci, L., Brumback, S., Keller, K., & Colucci, D. (2001). High Performance Wide-Area Optical Tracking: The HiBall Tracking System. *Presence: Teleoperators and Virtual Environments* , pp. 1-21.

[20] Whitton, M., Cohn, J., Feasel, J., Zimmons, P., Razzaque, S., Poulton, S., et al. (2005). Comparing VE Locomotion Interfaces. *IEEE Virtual Reality* , 123-130.

[21] Wingrave, C., Williamson, B., Varcholik, P., Rose, J., Miller, A., Charbonneau, E., et al. (2010). Wii Remote and Beyond: Using Spatially Convenient Devices for 3DUIs.*IEEE Computer Graphics and Applications,30(2), March/April.* .

[22] Zanbaka, C., Lok, B., Babu, S., Xiao, D., Ulinksi, A., & Hodges, L. (2004). Effect of Travel Technique on Cognition in Virtual Environments. *IEEE Virtual Reality*, (pp. 149-156). Chicago, IL.

Authorized licensed use limited to: Javii Austin. Downloaded on May 14,2026 at 17:43:32 UTC from IEEE Xplore. Restrictions apply.