Journal of Universal Computer Science, vol. 19, no. 8 (2013), 1086-1104
submitted: 30/11/12, accepted: 29/3/12, appeared: 28/4/13 © J.UCS

An Evaluation of Targeting Accuracy in Immersive First-Person Shooters Comparing Different Tracking Approaches and Mapping Models

Arturo S. García
(SymbiaIT, Albacete, Spain
arturo@symbiait.com)

Andrés Olivas, José P. Molina, Jonatan Martínez, Pascual González
(LoUISE research group, University of Castilla-La Mancha, Albacete, Spain
Andres.Olivas@alu.uclm.es, JosePascual.Molina@uclm.es
Jonatan.Martinez@uclm.es, Pascual.Gonzalez@uclm.es)

Diego Martínez
(University of Bristol, Bristol, United Kingdom
Diego.Martinez-Plasencia@bristol.ac.uk)

Abstract: Immersive Virtual Environments typically rely on a tracking system that captures the position and orientation of the head and hands of the cyber-user. Tracking devices, however, are usually quite expensive and require a lot of free space around the user, preventing them from being used for gaming at home. In contrast with these expensive capture systems, the use of inertial sensors (accelerometers and gyroscopes) to register orientation is spreading everywhere, finding them in different control devices at affordable prices, such as the Nintendo Wiimote. With a control like this, the player can aim at and shoot the enemies like holding a real weapon. However, the player cannot turn the head to observe the world around because the PC monitor or TV remains in its place. Head-mounted displays, such as the Oculus Rift, with a head-tracker integrated in it, allows the player to look around the virtual world. Even if the game does not support the head-tracker, it can still be used if the sensor driver emulates the mouse, so it can control the player’s view. However, the point of view is typically coupled with the weapon in first-person shooting (FPS) games, and the user gets rapidly tired of using the neck muscles for aiming. In this paper, these two components -view and weapon- are decoupled to study the feasibility of an immersive FPS experience that avoids position data, relying solely on inertial sensors and the mapping models hereby introduced. Therefore, the aim of this paper is to describe the mapping models proposed and present the results of the experiment carried out that proves that this approach leads to similar or even better targeting accuracy, while delivering an engaging experience to the gamer.

Keywords: Video games, first-person shooters, virtual reality, human-computer interaction
Categories: I.3.6 [Computer Graphics]: Methodology and techniques

1 Introduction

First-person shooters (FPS) are an outstanding genre among videogames [Adams, 06]. They are characterized by the fact that the game environment is rendered as seen through the main character's eyes, showing only a small part of that character, usually its hands and the weapons it carries.

This genre had its beginnings in 1974, when the two first games following the philosophy of first-person view appeared: Maze War (Steve Colley, 1973-1974) and Spasim (Jim Bowery, 1974). In the market for PC games, it is a must mentioning the games Wolfenstein 3D (1992), Doom (1993) and Quake (1996), all three developed by Id Software [Cotton, 97]. The last one, Quake, was the first FPS that used 3D graphics acceleration hardware.

Over the years, however, the control mode has not changed much. The first games used only the keyboard to move around the environment and shoot the enemies. Soon, the mouse joined the keyboard to form a combination that has characterized FPS games for years. Currently, two peripheral configurations are used for the game controls: mouse and keyboard (WASD keys) in the case of PC games, and a joystick or a gamepad in the case of videogame consoles. Despite their obvious differences, the two are based on the same principles: the fingers of one hand controls the character's movement around the environment, and the fingers of the other hand are used to aim the weapon at the target.

Although FPS games have amply fulfilled their entertainment purpose for years, the use of these peripherals, along with the monitor in PCs or the TV in consoles, makes the gameplay far from reality. First, if the user wants to see the virtual world around the main character, the avatar, then the user does not turn his head and body as he would do in reality, but instead he presses a button or moves a stick. Second, the user does not wield the weapon in his hands, again he uses the keyboard, mouse or joystick to aim at. And, finally, in the vast majority of FPS games these two characteristics are clearly coupled, and it is common that the avatar shoots in the direction of sight and, vice versa, the view follows the movement of the weapon, thus reducing the degrees of freedom of the character to simplify its control [Piekarski, 03].

There are other games, however, where the sight and the weapon can be controlled independently, as in battle tank simulation games, for example Tiger Hunt [Exelweiss, 02]. On a tank, the player can move in one direction while shooting in another one, and even looking in a completely different direction. With no doubt, this poses a challenge when mapping the different actions of the game to the keyboard and the mouse. Looking for an experience closer to reality, many simulator enthusiasts include in their gaming system more than one monitor, thus increasing the field of view and reducing the need to change the viewpoint using the controls. Other gamers choose to incorporate a head-tracker such as the NaturalPoint TrackIR [NaturalPoint, 12], consisting of a camera mounted on the monitor and a set of reflective labels or LEDs on the user's head, which allows them to switch views with simple head movements. In any case, whether with one or more monitors, the user can not move the head very much as the window he looks through to observe the world, the monitor, does not move from its place.

There are also gaming systems where the user has the gun in his hand instead of a keyboard or mouse, as in the consoles Nintendo Wii [Lee, 08] and Sony PlayStation 3 with the Move motion controller [Humphries, 10], or even the user himself is the controller as in the Microsoft Xbox360 with the Kinect system [Zhang, 12]. The Wii, for example, has a game controller (Wiimote) which, combined with a bar of LEDs mounted on top of the TV, makes it possible to obtain the position and orientation of the controller itself, allowing to pointing and shooting enemies like holding a real weapon. In fact, accessories for the Wiimote can make it look like a pistol, shotgun or crossbow. But again, whether with the bare Wiimote or dressed with any accessory, the user can not turn to shoot around because the world is shown through the TV, and it does not move from its place.

To immerse the player in a 360 degree FPS experience, he can be surrounded with multiple displays, playing inside of a dome or CAVE-like installation, or one single display can follow the user wherever he looks, mounting a helmet and a head-tracker. In [Humphries, 11] an installation with projection screens is running the game Battlefield 3, but apart from military uses, it is beyond the reach of the common FPS player, in terms of money and space. This player may find more affordable the second option, with the current offer of personal displays aimed at the consumer market, such as the i-glasses [I-O Displays, 12] front-mounted display (FMD), or the Sony HMZ-T1 and -T2 [Sony, 11] head-mounted displays (HMD). Aiming at the videogame market, some companies have released HMDs that integrate a head-tracker, as for instance the VictorMaxx Stuntmaster [Gradecki, 94], the Trimersion helmet [Kuntz, 07], and the latest development Oculus Rift [Oculus, 12]. It is not mandatory that the FPS game has native support for the head-tracker to use it, providing that the head-tracker driver can emulate the mouse as a device. In this case, the sensor can be used to perform the function typically associated with the mouse in this kind of games, which is to control the character's view and, together with it, to aim the weapon. The player, now immersed in the game, will find himself looking around in a more natural way, but will struggle to aim his weapon, as he will have to do this, not with the hands, but with his head, forcing the neck muscles to a point that the player gets tired quickly.

For a total immersion of the player in the game, it is preferable to have a system that captures the position and orientation of both the user's head and the hand that holds the weapon. This is what was offered by Virtuality arcade machines [Cotton, 92], running titles like Dactyl Nightmare [Lavroff, 92]. The machine model that this title was designed for was the 1000CS, in which the user stands during the game, with a HMD on his head and a joystick in one hand, whose positions were captured independently by a motion capture (MoCap) system. If this hardware was used in current FPS games, then this would give rise to a form of entertainment much more realistic than the one obtained with the keyboard and mouse or game controllers. The problem with this hardware is its high cost, prohibitive for the common FPS player, and the difficulty to have such hardware at home, as it needs plenty of space around the user.

In contrast with these expensive capture systems, the use of inertial sensors (accelerometers and gyroscopes) [Burdea, 03][Sherman, 03] to register orientation is spreading everywhere, finding them in different control devices at affordable prices, such as the Logitech Air Mouse [Logitech, 07], the Cyberstik2 mid-air joystick [Cline, 05], the Wiimote controller as already said, and even in other devices such as smartphones and tablets. Therefore, the main objective of this work is to study whether it is possible to use these money- and space-savers inertial sensors, not only to capture the orientation of the player's head, as a head-tracker does, nor the player's weapon, as the Wiimote does, but both of them, and without position data then check if the result is similar to using a MoCap in terms of realism and playability. This way, this paper aims to answer the following question: could the user have an immersive FPS game at home using just orientation sensors instead of both position and orientation sensors? This will be answered by the proposal and study of several models that try to use orientation data (that can be extracted from cheap inertial devices) to provide a similar experience to the use of position and orientation data. This would lead to an immersive FPS game at a reasonable cost.

Some previous work can be found in the literature related to FPS games. In [Torchelsen, 07], a similar experiment is proposed, but this time comparing a standard FPS game control (keyboard and mouse) to an immersive set-up (HMD, data gloves and tracking system). Even though the user was seated in the immersive set-up, the results regarding the user engagement are similar to the experiment described here. On the other hand, as it will be discussed in section 3, [McMahan, 12] proposes a CAVE and a 6 degrees of freedom tracking system to compare again between an immersive and a desktop set-up.

This paper is structured as follows. The next section describes the models used to map the user movements to its corresponding avatar in the FPS game developed. Section 3 describes the experiment carried out and section 4 its main conclusions. Finally, section 5 lists some future work.

2 Models with and without position information

In contrast to FPS games based on commodity hardware, virtual reality applications typically decouple the view and pointing direction by tracking the user's head and hand independently [Bowman, 05]. Thus, every movement that the user makes in the real world can be accurately translated into a virtual world using a position and orientation capture system. However, if only the orientation information is available, it is necessary to compensate the lack of position information with assumptions that allow the interpretation of the actual movements of the user as accurately as possible. For example, it can be assumed that the position of the user’s hand will not separate more than the length of an arm from the body trunk. These assumptions are based on a detailed study of the user and the movements performed when wielding and aiming a gun with one or two hands, and they led to the models discussed in section 2.2.1 and 2.2.2.

2.1 Models with position

These are the ideal models from the point of view of realism, since the actual data of the position and orientation of the user can be used at any time to place both their viewpoint and their gun in the videogame scenario. Two variations were considered, in the first one the gun is held with one hand and in the second it is held with two hands.

2.2 Models without position

In this case, the only thing known is the orientation of the user's head and hands, and then the models fill the missing information by assuming a certain position for the point of view and the gun in the game scenario.

Thus, the following models assume that the user is standing and barely moves from his place. This allows setting the position of the point of view at the same height than the user's eyes. Then, the combination of that position with the orientation returned by the head-tracker makes the game follow the movement of the head in a reasonably accurate way.

The models also assume that when the user moves his head, it is accompanied by the movement of the body trunk and the upper limbs. They also assume that when the user moves the gun, his head accompanies this movement, although the orientation of the head and the gun does not necessarily match. Thus, although the orientation of the weapon is given by the tracker of the hand, or the device held by the user, its position can be fixed at a point relative to the point of view. If the user moves his head, the point of view will move and the representation of the weapon will move along with it. The difference between the models presented in this section is found in that relative position, considering two pivot points, the elbow in one case and the wrist in the other. In addition, the models will also consider the differences between wielding the gun using one or both hands.

2.2.1 One-hand models

- **Model of aiming using the wrist.** This model assumes that the user holds the gun with one hand and that it is the wrist what rotates when aiming, so it fixes the position of the wrist and applies there the rotation given by the tracker. Fig. 1 (left) shows the position at the time of firing that has been chosen as reference, and fig. 1 (right) shows the representation of the hand in the game scenario.

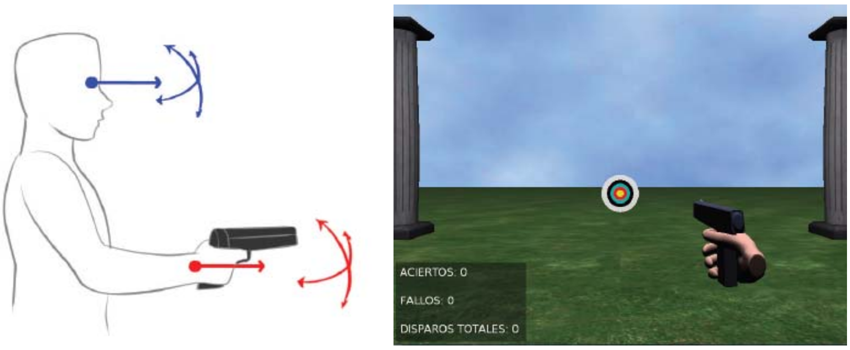

Figure 1: One-hand model, aiming with the wrist

- **Model of aiming using the elbow.** This model assumes that the user rotates the elbow, and not the wrist, when aiming with one hand. The relative position taken for this model is shown in fig. 2 (left). This time a representation of the forearm is used in the game, the orientation data received from the tracker will be used to set the rotation of the elbow, as shown in fig. 2 (right).

Figure 2: One-hand model, aiming with the elbow

2.2.2 Two-hand models

- **Model of aiming using the wrists.** This model assumes that the user wields the gun with both hands and aims by rotating the wrists. Fig. 3 (left) shows the position at the time of shooting that has been taken as reference, and in fig. 3 (right) a 3D model of the hands. The weapon will be oriented according to the data coming from the tracker.

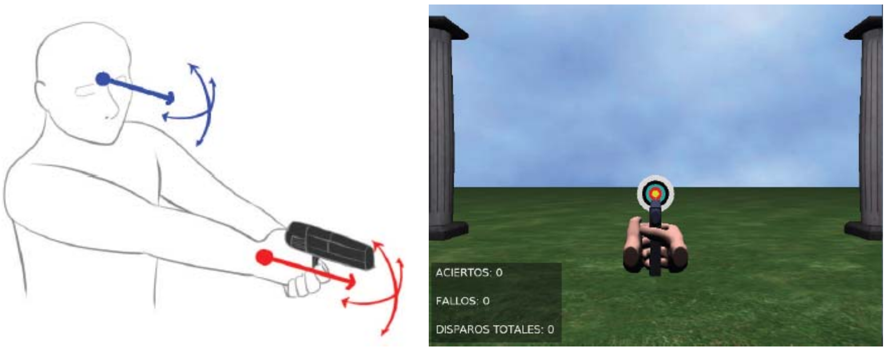

Figure 3: Two-hand model, aiming with the wrists

- **Model of pointing with the elbows.** Again, this model assumes that are the elbows, and not the wrists, what the user rotates when aiming with both hands. The reference position of this model is shown in fig. 4 (left). The game will use a 3D model of the two arms, as shown in fig. 4 (right), whose orientation comes from the tracker.

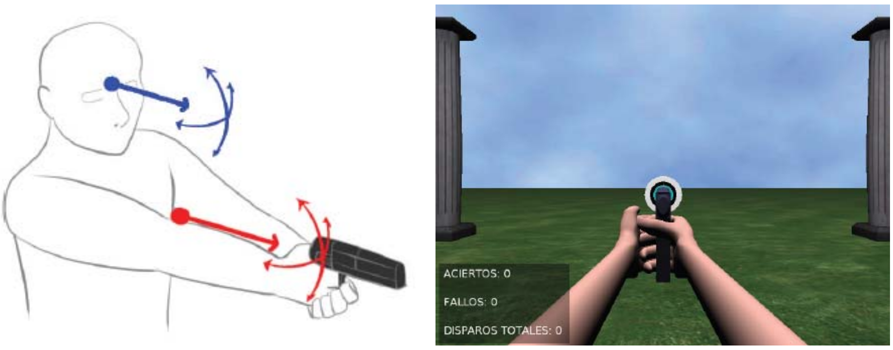

Figure 4: Two-hand model, aiming with the elbows

2.2.3 Models for left-handed people

Left-handed models are symmetrical to those described for right-handed people, changing the position of the weapon to the left side of the point of view.

3 Experiment

A series of tests have been conducted with real users in order to assess which model performs best. This experiment is somewhat similar to the one described in [McMahan, 12], in which the combination of high/low interaction fidelity and high/low display fidelity in a FPS environment is compared. However, they did not try to find an affordable solution to the high-interaction fidelity, since the setup described in [McMahan, 12] uses a CAVE and a 6 degrees of freedom tracking system for the head and the weapon, while the low-interaction fidelity setup uses a keyboard and a mouse. Also, they studied two tasks combined, aiming and locomotion, and plan to study them independently in the future, as it has been done regarding the aiming task in the experiment described in this section.

The following sections detail the experiment carried out.

3.1 Hardware and software

3.1.1 Hardware

An iotracker tracking system with four cameras and two rigid bodies, an i-glasses head-mounted display, and a Wiimote control (fig. 5) were used for this experiment.

Iotracker is a motion tracking system that is able to capture the position and orientation in real time [Pintaric, 07]. For this purpose, this system makes use of infra-red cameras, rigid bodies and a computer running the iotracker software. The rigid bodies, or targets, are covered with a material that reflects the infra-red light that is cast from the cameras, which is used to detect the targets and send the information to the computer. The specialized software uses the data coming from the four cameras and provides the exact position of each target by triangulation. The computer is also running a VRPN server [Tailor, 01] that provides TCP/IP data about the position and orientation obtained from the iotracker software to any client connecting to it.

The i-glasses [I-O Displays, 12] has two displays, one for each eye, and stereo headphones. Combined with the iotracker, it allows the user to look at any point of the environment, achieving a more immersive experience by completely replacing the real-world by the game environment.

The Wiimote is the game controller of the Nintendo Wii console [Lee, 08]. It is wireless and can be easily connected to the computer via Bluetooth. In addition, there are accessories to use it like a weapon, in this case a gun, and shoot by pulling the trigger, which increases the sense of realism and immersion of the game.

Figure 5: Participant with the i-glasses and the Wiimote

3.1.2 Software

The FPS game was created with Ogre3D [Kerger, 10]. Ogre3D is based on C++ and provides excellent tools for creating three-dimensional scenes, but other functionalities require the use of plug-ins and additional libraries. Thus, OIS has also been used for keyboard and mouse input, VRPN [Taylor, 01] for dispatching data from the iotracker, Wiiyourself! for the Wiimote, OpenAL for the sound, Gorilla to create the HUD, and Sion Tower Collisions for the collision detection. Autodesk Maya 2010 has been used for the creation of 3D models, together with OgreMax Scene Exporter to export models to Ogre3D compliant files (.mesh). Finally, Microsoft Visual Studio 2010 was used as the development environment, having everything running on Windows 7.

3.2 Design of the experiment

All the variables that could influence the tests were analysed to carry out their design. They also were isolated to the extent possible so that they can be individually examined and generalize the results of the tests.

3.2.1 Game scenario

The scenario where the tests are conducted has been kept as simple as possible. The user should focus on his task, without being distracted by other elements. Initially, the scenario was formed only by the ground and the sky, but some columns were added for the sole purpose of serving as a reference to the user when the user moves through the environment.

3.2.2 Targets

Since the task of the user in an FPS game involves shooting at targets in the scene, the main variables come from the characteristics of these targets:

- **Shape.** The geometry of the targets defines much of the user behaviour when it comes to aiming. The targets could have human form, but its irregularity will difficult any conclusion about the users aiming accuracy. It is not the same aiming at the head or the body of an enemy, as they usually have different size. Therefore, the best option is the circular or spherical shape, where the outer points are equidistant from the centre of the object, and this way a failure has always the same weight.
- **Size.** This characteristic has a direct impact on the difficulty of hitting the target. It is therefore reasonable to assume that the larger the target, the easier it will be to hit it.
- **Spatial distribution.** The placement of the targets can be divided according to their relationship to the user: *azimuth*, if distributed around himself; *elevation*, if they are within a specific distance from the ground; and *distance*, if they are close or far from the user.

3.2.3 Other variables

Other variables were taken into account when designing the tests: number of targets, sequence of targets, number of shots, time for completion, cross sight, and head-up display (HUD).

3.3 Tests

From the design, four synthetic tests were developed that focus individually on specific variables of the targets, to study if any has more weight than other in the number of hits and misses of the user: azimuth, elevation, distance and size. A combined synthetic test was also added, which mixes all the variables; as well as a test based on an existing commercial game. Finally, one last scenario was added for the learning task.

3.3.1 Synthetic tests

All the synthetic tests have the following characteristics in common: there are ten circular targets, all with one-meter diameter except in the size test; there is not ammunition or time limitation, although the time is recorded for further analysis; there is not any cross sight that can ease the task; and the HUD only shows hits, misses and total number of shots.

- **Azimuth test.** The targets are placed around the user so that he has to rotate 360 degrees to reach them all. Fig. 6 (left) shows this arrangement from a bird's-eye view. All the targets are shown in place from the beginning of the test, the user will discover them as he rotates.
- **Elevation test.** The targets are arranged in front of the user at different heights, as shown in fig. 6 (right). In this case, the targets are shown in sequence, so once a target is hit, next target appears. This is intended to prevent the user from shooting the targets by doing only small movements from one target to the closest one. In addition, the order of appearance is different for each model of control, which also tries to prevent the user from using his memory when the test is repeated with another model of control.
- **Distance test.** The targets are placed in front of the user, at the same height but with different distances. Fig. 7 (left) shows the distribution from an elevated view. As in the case of the previous test, the ten targets appear in sequence but in a different order depending on the control model.
- **Size test.** The targets are located in front of the user and have different sizes. Again, the targets are shown in sequence but in an order that varies from one the control model to other. Fig. 7 (right) shows the target from the user point of view in the game scenario.
- **Mixed test.** In this test, all targets are visible from the start, and their placement across the game scenario mix all parameters evaluated separately in previous tests. There are targets of different size, height, distance and distributed around the user.

Figure 6: Azimuth (left) and elevation (right) tests

Figure 7: Distance (left) and size (right) tests

3.3.2 Real-game test

This test is to verify that the models are perfectly applicable to an existing commercial game, recreating one of them.

Among the various titles, the chosen one is “Link’s Crossbow Training” for Wii, which uses the Wiimote to point at the screen and shoot the targets that appear on it to get the best score possible within the given time. The game has different levels, but level 3-1 has been chosen because the targets are similar to the other tests (fig. 8). Its features are: the number of targets is 31, there is a time limit, 60 seconds, but enough to finish set battery of targets; each hit is worth 100 points and each failure means the loss of 20 points; the HUD includes the score so far and the remaining time.

Figure 8: Real-game test

3.3.3 Learning scenario

This is a game scenario that participants use to become familiar with the different control models tested in the experiment. When they complete this scenario, or enough skills are obtained, they are ready to conduct the synthetic and the real-game tests.

3.4 Procedure

As explained in previous sections, there are two different ways to wield the weapon, with one hand or two hands, and three different control models has been defined for each way, making a total of six possible combinations. In addition, we have designed a total of five synthetic tests, a real game test and a learning scenario. Completing all possible combinations of control models and tests was time-consuming, which may alter the results because of the fatigue and the gained experience. To reduce these effects, we chose an approach "between subjects", so that each participant would perform tests only for a hand or two. Thus, each user completed the tests three times, one for each control model, but in different order of tests in each round.

At the beginning of the experiment, the participant was given a participation agreement, following the ethics code for usability research of the Asociación Interacción Persona-Ordenador (AIPO) [Concejero, 06], a general questionnaire and instructions.

Afterwards, each round began with the learning scenario so that the user could familiarize with the control model that he was going to use in that test round. Once completed, it was the turn of synthetic tests: azimuth, elevation, distance and size. Here the order varies for each user, to counter-balance the effects of the experience they gain. To this extent, a Latin-square approach was used. After this, the test to be performed was the mixed one, a mixture of the parameters of the former tests. Finally, the real test is performed which, as it has been seen, is similar to a commercial game.

During each test, the system provided the following data: *hits*, which must be constant as users must reach all targets to complete the test, but is recorded in case some user does not complete it for some reason; *misses*, total number of shots failed; *time spent*, though the participant was not aware of it.

After each round, each participant complimented a questionnaire, which included his opinion on the control model used. At the end of the experiment, he also complimented a final questionnaire of usage created to collect the opinion on the whole system, which also included a section where the user had to sort from best to worst the different control models used.

3.5 Participants

The Latin-square method was used to vary the order of the synthetic tests with the purpose of avoiding the possible influence that this could have on the results. Following this approach, a multiple of twelve was the number of participants required to evaluate each way of wielding the weapon. As the gun can be held using one or two hands, the total amount of participants was twenty-four. All of them were university-level students aged from 20 to 24 that had already had a Human-Computer Interaction course and with varied experience in video games. Finally, no study of the behaviour of the participants was performed based on their gender.

4 Results

This section shows the results of the evaluation performed for the different models.

4.1 One-hand models

a) System data

Figures 9 and 10 show the average number of misses and the average task completion time (TCT) for each test. The great similarity in the graphs is due to fact that the failures and the TCT are closely related, since a miss forced to shoot again, with the corresponding loss of time. As can be seen in these figures, the model that behaves worse is the one that uses the wrist to aim, showing more misses in all tests compared to other models. In turn, the model with position is the one that performs better, although its results are similar to the elbow model.

The non-parametric Friedman test -equivalent of the parametric test ANOVA for repeated measures- was used to corroborate the differences in the performance of the models that can be seen in fig. 9. It was used to test if the differences between the three models were statistically significant. The test results χ^2(2) = 11,400, p = 0 .003 show that the three models do not behave similarly. Post-hoc analysis with Wilcoxon Signed Ranks test was conducted with a Bonferroni correction applied, resulting in a significance level set at p<0 .017. There were significant differences between models 1 and 2 -position and wrist models- (Z =–2,398, p = 0 .016), models 2 and 3 -wrist and elbow- (Z=-3,036, p=0,002) but this was not the case between models 1 and 3 -position and elbow- (Z=-0,414, p=0,679).

This corroborates the first impression, the elbow and position models behave similar and outperform the wrist model in a statistically significant way.

Regarding the time measures, although the results for the three models are similar, having a closer look at fig. 10 reveals that the elbow model takes longer in some tests compared to the model with position. This suggests that aiming is more difficult and it takes more time before being certain about pointing to the target. This effect is much more pronounced in the wrist model, with worse TCT than the rest. There is an extra time needed to complete the mixed and the azimuth tests, since the user has to rotate 360 degrees in space to shoot all the targets.

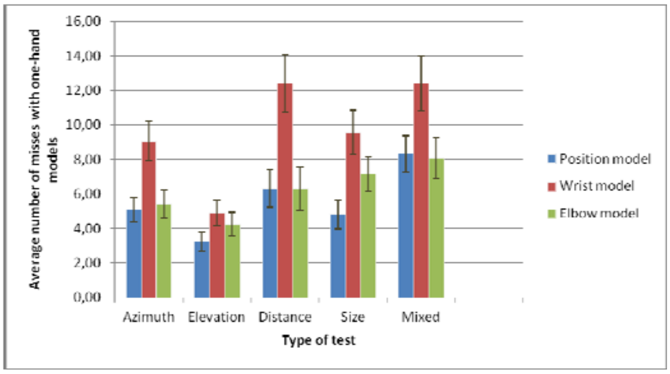

Figure 9: Average number of misses with one-hand models

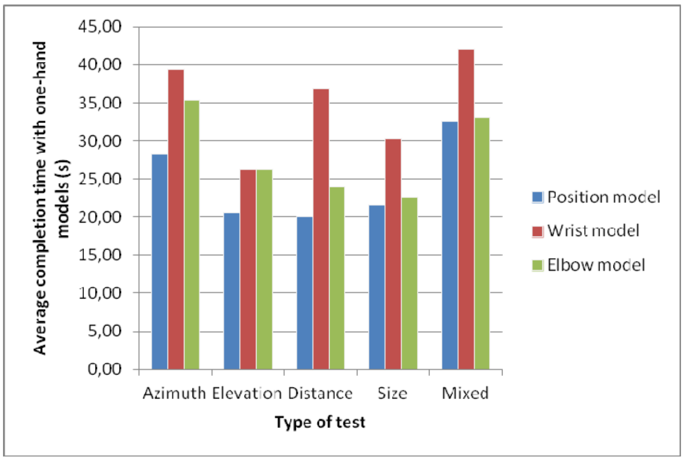

Figure 10: Average completion time with one-hand models

b) Questionnaires

In the questionnaires, the difficulty in aiming was scored from "1" to "5", corresponding "1" to having no problem at aiming and "5" to having much trouble. The results expressed by users confirm those collected by the system, they indicate that the worst model is the wrist one with 3.17 of average perceived difficulty, and the next one is the elbow with 2.17. The best model would be the one with position that receives 1.82 in average. Based on these data the elbow model obtains good results, being very close to that obtained by the ideal model, which makes use of the position plus orientation.

One of the most interesting data collected is how the models were ordered by the users according to their accuracy in aiming. Seven users chose the model with position in first place, five the elbow model and none the wrist model. This is quite significant since almost half of the users prefer the model without position to the one that makes use of it, even though it is more realistic.

4.2 Two-hand models

a) System data

Figures 11 and 12 show the average number of misses and the time in seconds (TCT) taken by the users in each test. Again, there is a clear relationship between misses and time spent, since a miss implies a new shot and an extra time spent on it as a result. The goodness of the model also influences the accuracy, since it will take more or less time to aim properly at the target depending on it. For example, there is a peak of misses in the distance test which impacts on another peak in the time figure. Part of the increase of time spent on the mixed and azimuth test is due to fact that the user has to rotate 360 degrees to get to the objectives and they do not only need to aim, but also move in space.

The failures and the TCT figures illustrate the good behaviour of the elbow model, which gets very good results, clearly better than the rest. The model with position is next in efficiency and is significantly better than the wrist model, getting this one the worst records.

Following this first impression, further tests were performed to the data in order to extract more significant conclusions about the average number of misses. The Friedman test reported a statically significant difference χ^2(2) =69,924, p = 0 .000. In this case, the Wilcoxon Signed Ranks Test showed that the three results were different in a statistically significant way (Z = - 2,453, p = 0 .014) for models 1 and 2 (position and wrist), (Z= -5,219, p=0,0001) for models 1 and 3 (position and elbow), and (Z= -6,887, p=0,0001) for models 2 and 3 (wrist and elbow).

This also corroborates the first impression, the elbow model is the one that best performs, followed by the model with position. It is worth noting that the fact that the elbow model surpasses the model with position means that the users were more accurate with it. It is true since the elbow model simplifies the process of aiming and the user still has to position and rotate their gun to aim in the model with position, being more realistic but more difficult to use at the same time.

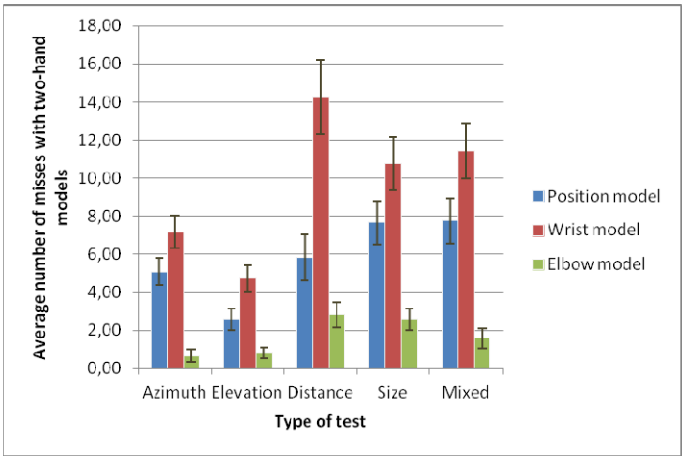

Figure 11: Average number of misses with two-hand models

b) Questionnaires

The opinions gathered in the questionnaires indicate that the model that is considered to be most difficult by users is the wrist one, with an average of perceived difficulty of 2.83. The next one is the model with position, obtaining a good result, with an average of 1.58. The elbow model comes out the best again, being very close to the best possible score with an average of 1.17. It is significant that only two users rated this model with a grade worse than "1", and in that case the grade was "2".

According to the ordering from high to low in terms of accuracy in aiming, the same conclusion can be extracted again. In this case, ten users give exactly the same order, which is very revealing. The elbow model is the best valued by users, followed by the model that uses position plus orientation.

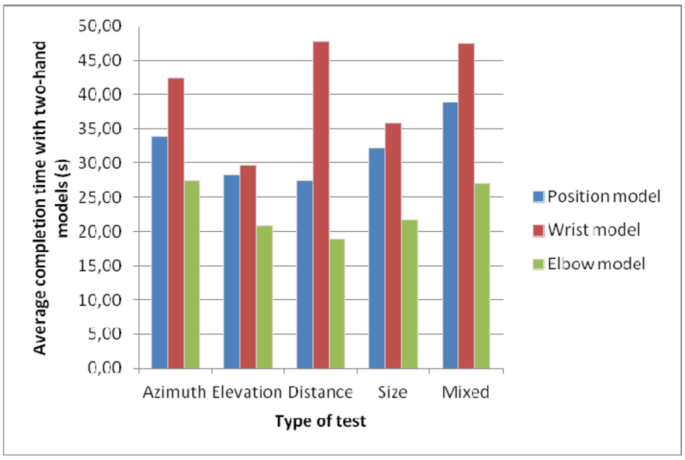

Figure 12: Average completion time with two-hand models

4.3 Results from the usage questionnaires

Several questions regarding their experience in the use of the system were presented to the users in this final questionnaire. This included questions about the environment, the use of the HMD, problems with the devices, and immersion. In general, the attitude of users towards the system was very positive. Initially, they were all surprised by the realism and fidelity of their movements in the game. After the tests, many users were interested in knowing how the system worked.

4.4 Conclusions

The purpose of the experiment described in this paper was to compare a system that uses position and orientation data for aiming, with another which lacked the position data. As seen, similar results to the position and orientation model are obtained with a model that maps the gun tracked orientation to the avatar's elbow when holding the weapon with one hand, and even better results are obtained with the model that maps the orientation on both elbows when yielding the gun with two hands. This confirms the feasibility of replacing the position and orientation model with one that relies solely on the orientation.

Even though it was not the most realistic model, it was observed that two-handed model that maps orientation to the avatar's elbows obtained great results. They way it is designed points out the reason of this good behaviour, since it eventually simplifies the process of aiming. Users were very receptive to this model and they enjoyed it more because of its ease of use. They also made fewer mistakes and allowed them to aim faster.

On the other hand, the worst model was undoubtedly the one that maps the orientation data on the wrists. Its fixed position makes the aiming task too difficult and it is not realistic. This model could be discarded in favour of the elbow model.

The one-hand models did not show the same good results than the two-hand ones. This is partly due to the fact that the relative position chosen for the 3D hand model did not reflect the way that many users played the game. Most of them used to play centring the Wiimote on the screen, not placing it on his right side as the relative position chosen -and thus the one defined for the virtual gun- did. They would have performed certainly better by placing the virtual gun centred on the screen, as it was one of the keys of the good results obtained by the two-hand model.

Playing with an immersive system can be tiring if it is used for a long time. The fact of having the arms raised for aiming ends up leading to discomfort. Something similar happens when too many head movements are needed to look for targets and aim at them.

Something that also became apparent while testing is the willingness that users have to this new way of playing. After testing, the majority of users were surprised and praised the realism and the degree of immersion achieved. Some of them even asked if there is something similar for sale to enjoy at home. In short, users are very attracted by this technology.

Following the same line, the test that recreates an existing commercial game was the one that the users enjoyed most. The challenge of having a limited time and a score counter did certainly encourage their competitiveness. Almost everyone wanted to improve their initial score when they repeated the test, asking also for the final score of other participants to see how good they had played. This response from users makes it clear the possibility of bringing a system similar to the one proposed in this work to the videogame market.

5 Future work

The misses figure illustrates the good behaviour of the elbow model, which gets very good results, clearly better than the rest. The model with position is next in efficiency and is significantly better than the wrist model, getting this one the worst records.

Although satisfactory results were obtained, some improvements or additions can be considered. On the one hand, the models could be adapted to the physical measurements of each user. The system would collect users' personal data before starting, and thus could raise the point of view according to user's height or make an adjustment of the relative position of the weapon given the length of his arm. On the other hand, the ability to move around the environment could be included, because the user remains stationary in the proposed system, he cannot move from the centre of the scene.

Finally, it should not be forgotten that the orientation data has always been obtained from the iotracker system in this experiment, and the fact of replacing this system with other devices such as gyroscopes and accelerometers would bring other problems that have not appeared here, as the accumulation of errors (drift) [Burdea, 03][Sherman, 03]. Therefore, further work remains to create a prototype that makes use of these devices, and to repeat the same tests in order to corroborate the conclusions obtained here.

Acknowledgements

This work has been supported by the projects PEII09-0054-9581 and TIN2012-34003. Thanks to the users who have participated voluntarily in the experiments.

References

[Adams, 06] Adams, E., Rollings, A.: Fundamentals of Game Design, Prentice-Hall Inc, 2006

[Bowman, 05] Bowman, D.A., Kruijff, E., LaViola Jr., J.J., Poupyrev, I.: 3D User Interfaces: Theory and Practice, Addison-Wesley, 2005

[Burdea, 03] Burdea, G.C., Coiffet, P.: Virtual Reality Technology, Second Edition, Wiley-Interscience, 2003

[Concejero, 06] Concejero-Cerezo, P.: Código ético de la investigación en usabilidad e Interacción Persona-Ordenador para pruebas con usuarios, Asociación Interacción Persona-Ordenador, 2006, URL: http://www.aipo.es

[Cline, 05] Cline, M.S.: Power, Madness, and Immortality: The Future of Virtual Reality, University Village Press, 2005

[Cotton, 92] Cotton, B., Oliver, R.: Understanding Hypermedia: From Multimedia to Virtual Reality, Phaidon Press Ltd., 1992

[Cotton, 97] Cotton, B., Oliver, R.: Understanding Hypermedia: Multimedia Origins, Internet Futures, Phaidon Press Ltd., 1997

[Exelweiss, 02] Exelweiss Ent.: Tiger Hunt, 2002, URL: http://www.exelweiss.com/desarrollo/videojuegos/pc/tigerhunt.php

[Gradecki, 94] Gradecki, J.: The Virtual Reality Construction Kit, John Wiley, 1994

[Kerger, 10] Kerger, F.: OGRE 3D 1.7 Beginner's Guide, Packt Publishing, 2010

[Humphries, 10] Humphries, M.: Review: PlayStation Move, Geek.com, 2010, URL: http://www.geek.com/articles/games/review-playstation-move-20100917/

[Humphries, 11] Humphries, M.: The ultimate Battlefield 3 simulator has been created, Geek.com, 2011, URL: http://www.geek.com/articles/games/the-ultimate-battlefield-3-simulator-has-been-created-20111031/

[I-O Displays, 12] I-O Display Systems: i-glasses, Accessed November 2012, URL: http://www.iglassesstore.com/,

[Kuntz, 07] Kuntz, S.: The story behind the Trimersion HMD, 2007, URL: http://cb.nowan.net/blog/2007/02/09/the-story-behind-the-trimersion-hmd/

[Lavroff, 92] Lavroff, N.: Virtual Reality Playhouse, Wiate Group Press, 1992

[Lee, 08] Lee, J.C.: Hacking the Nintendo Wii Remote, IEEE Pervasive Computing, 7(3), p.39-45, July 2008

[Logitech, 07] Logitech: Logitech Introduces the Future of PC Navigation , 2007, URL: http://news.logitech.com/press-release/consumer-products/logitech-introduces-future-pc-navigation

[McMahan, 12] McMahan, R.P., Bowman, D.A., Zielinski, D.J., Brady, R.B.: Evaluating Display Fidelity and Interaction Fidelity in a Virtual Reality Game. IEEE Transactions on visualization and computer graphics. Vol. 19, N. 4. IEEE Computer Society. 2012.

[NaturalPoint, 12] NaturalPoint Inc.: TrackIR - 6DOF Head Tracking, Accessed November 2012, URL: http://www.naturalpoint.com/trackir/

[Oculus, 12] Oculus VR Inc.: Oculus Rift Headset, Accessed November 2012, URL: http://www.oculusvr.com/

[Pierkarski, 03] Piekarski, W. and Thomas, B.H.: ARQuake - Modifications and hardware for outdoor augmented reality gaming, In 4th Australian Linux Conference, Perth, Australia, 2003

[Pintaric, 07] Pintaric T., Kaufmann H.: Affordable Infrared-Optical Pose-Tracking for Virtual and Augmented Reality, Proceedings of “Trends and Issues in Tracking for Virtual Environments” Workshop, IEEE VR 2007, Charlotte, NC, USA, pp. 44-51, March 2007

[Sherman, 03] Sherman, W.R.: Understanding Virtual Reality: Interface, Application, and Design, Morgan Kaufmann Publishers, 2003

[Sony, 11] Sony Electronics Inc.: Sony Introduces the Worlds First Personal 3D Viewer, 2011, URL: http://news.sel.sony.com/en/press room/consumer/television/release/ 60813.html.

[Taylor, 01] Taylor II, R.M., Hudson, T.C., Seeger, A., Weber, H., Juliano, J., Helser, A.T.: VRPN: A Device-Independent, Network-Transparent VR Peripheral System, Proceedings of the ACM Symposium on Virtual Reality Software & Technology 2001, VRST 2001. Banff Centre, Canada, November 15-17, 2001

[Torchelsen, 07] Torchelsen, R.P., Slomp, M., Spritzer, A., Nedel, L.P.: A Point-and-Shoot Technique for Immersive 3D Virtual Environments, Simpósio Brasileiro de Jogos para Computador e Entretenimento Digital, SBGames, Brazil, 2007

[Zhang, 12] Zhang, Z. Y.: Microsoft Kinect Sensor and Its Effect, IEEE Multimedia, 19(2). p. 4-10, February 2012