Integrating Head and Full-Body Tracking for Embodiment in Virtual Characters

David Borland*

Universitat de Barcelona and Renaissance Computing Institute (RENCI)

ABSTRACT

In virtual embodiment scenarios the participant in an immersive virtual environment is presented with a first-person view of a virtual body, giving them the illusion that the body is, to some extent, their own. This body-ownership illusion can be strengthened by animating the virtual body based on the user's motion. The sometimes poor head-tracking quality of a full-body tracker can induce simulator sickness, especially when wearing a head-mounted display, so a separate higher-quality head-tracking system is used. We discuss the issues present when integrating the data from two such tracking systems, outline principles for generating appropriate first-person views that maintain the user's body-ownership illusion, and describe two related methods based on these principles.

Index Terms: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual reality;

1 INTRODUCTION

1.1 Tracking for Virtual Embodiment

Head tracking in virtual reality (VR) enables the displayed images to be updated based on the head movements of the user, providing the fundamental sensorimotor contingency that gives rise to a sense of presence, or *place illusion* (PI), in virtual environments [2]. Discrepancies between the user's physical motion and displayed virtual images can cause simulator sickness, so the user's head motion must be tracked accurately and with low latency [5].

Full-body tracking coupled with an animated self-avatar is a strong determining factor in the degree of PI and *plausibility illusion* (Psi) felt by VR participants [4]. For virtual embodiment– an emerging area of VR research in which the participant is presented with first-person view of a virtual body, giving them the illusion that the body is, to some extent, their own [1, 6]–this full-body motor contingency can strengthen the body-ownership illusion [3].

Although most full-body tracking systems provide head pose data, they are often not accurate enough to generate first-person views without increasing simulator sickness. Such systems optimize multiple constraints, so other body parts may affect head-tracking, and tracking accuracy depends on a good calibration between the participant and a virtual skeleton. Head-tracking problems are exacerbated when wearing a head-mounted display (HMD), so a separate HMD-mounted head tracker can be used. Effective virtual embodiment requires that the data from these two separate systems be integrated such that simulater sickness is avoided while maintaining a faithful virtual representation of the participant.

1.2 System Description

We developed these methods using a wired Intersense IS900 head tracker and an OptiTrack Arena full-body tracker with 12 V100:R2 cameras. Virtual environments were implemented using XVR and the HALCA character-animation library. VRPN transmitted head-tracking data, and OptiTrack's NatNet client/server architecture transmitted body-tracking data. Although developed with this system, the methods are applicable to any combination of head tracker, body tracker, and skeletal animation system.

We make the following assumptions: 1) the head tracker position has been transformed to the midpoint of the user's eyes (the two offset cameras used for stereo views in VR are trivially generated from this position), 2) the body-tracking data has been applied to the virtual avatar, and 3) the data from both systems has been transformed into the application coordinate system (Figure 1).

2 TRACKER INTEGRATION

There are two objects to control with this virtual-embodiment system: the *viewpoint* (virtual camera) and *representation* (avatar bone positions and rotations) of the user. Ideally the output of the two tracking systems match–applying the head-tracking data to the virtual camera places it at the eye midpoint of the virtual avatar looking in the same direction for a matching viewpoint and representation–however in general there is a mismatch.

One option is to decouple the viewpoint and representation, moving the virtual camera with the head tracker and the avatar with the body tracker. This method provides the correct first-person sensorimotor contingency required in VR, however it also results in a discrepancy between viewpoint and representation, especially apparent when looking in a virtual mirror often used for virtual embodiment. Another option is to move the virtual camera with only the body tracker, however the resulting tracking inaccuracy of the head can increase the likelihood of simulator sickness.

2.1 Principles and Constraints

Assuming the head tracker provides more accurate data for the user's head pose that the body tracker, any method for integrating the two tracking systems should apply the following principles to match the viewpoint and representation: 1) use head-tracking data whenever possible for the virtual camera, and 2) modify body-tracking data as little as possible to match the head-tracking data. These principles should be applied given the following constraints: 1) avatar eye midpoint and look-at direction match virtual camera exactly, and 2) no additional foot motion introduced.

Matching the head rotation is straightforward, as the rotation for both the virtual camera and avatar head can be set directly from the head tracker without affecting the rest of the avatar. Matching the head position is more difficult. Directly translating the avatar head independently of the rest of the avatar deforms the avatar near the head and neck, which can be very noticeable. Translating the entire avatar–thus translating the head along with it–results in the feet sliding across, penetrating, or floating above the ground, which negatively affecting the body-ownership illusion.

2.2 Implementation

We developed two methods based on the above principles and constraints that exhibit some tradeoff in head-tracking fidelity vs. avatar-representation fidelity. Both methods assume the upper-body

*e-mail: borland@renci.org

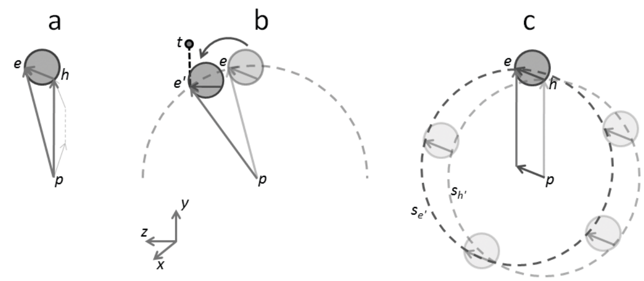

Figure 1: a) Simplified avatar upper-body bone structure. There are typically a number of bones between p and h, forming the "spine". b) Rotating p such that e matches e' also rotates h to no longer match the head tracker. c) Pivot-and-project method. Two spheres offset by $\overrightarrow{he}$ enable correct handling of the head rotation.

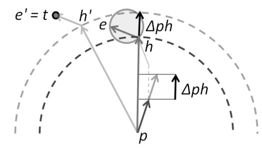

Figure 2: Pivot-and-stretch method. Adjusting the length of p by Δph enables the placement of h at h' and e at $e' = t$ while maintaining the head rotation from the head tracker.

skeletal structure shown in Figure 1-a, where p is the pivot bone at the base of the upper body, h is the head bone, e is the eye midpoint, **ph** is the vector from p to h given the current body-tracking data, and **he** and **pe** are the vectors from h to e and p to e respectively, given the current head-tracker rotation.

Any rotation of p translates all children of p, including h and therefore e. In both methods p rotates to translate e toward t. Rotating only p has the advantages that a) the entire lower body is left untouched, thus matching the body tracker exactly, b) the entire upper body from p to h is left untouched, matching the body tracker exactly, and c) the angle of rotation applied to p is minimal compared to rotating any other bone between p and h.

2.2.1 Pivot-and-Project Method

This method takes advantage of the fact that most head motion in a typical VR scenario is in the horizontal plane (e.g. panning the head from left to right). It uses the x and z components from t, forming t_{xz}, but the y component from the avatar eye midpoint after matching in x and z.

Rotating p rotates **pe**, forming a sphere with center p and radius $|\mathbf{pe}|$. A naïve approach projects t vertically onto this sphere, giving e', the desired avatar eye midpoint that matches t_{xz}. Rotating p by the angle between **pe** and $\mathbf{pe}'$ causes e to match e', and therefore t_{xz}. However, Figure 1-b shows that this method rotates the head such that it no longer matches the head tracker.

Instead p should be rotated such that keeping the head rotation constant matches e' to t_{xz}. To do so the desired head position h' must be computed such that when **he** is added to h', e' matches t_{xz}. Sphere $s_{h'}$, with center p and radius $|\mathbf{ph}|$ represents all possible positions of h', and sphere $s_{e'}$, with center $p + \mathbf{he}$ and radius $|\mathbf{ph}|$ represents all possible positions of e' such that $e' - \mathbf{he} = h'$ (Figure 1-c). Projecting t vertically onto $s_{e'}$ gives e', such that subtracting **he** produces h'. Rotating p by the angle between **ph** and $\mathbf{ph}'$ results in h matching h', and reapplying the rotation from the head tracker results in e matching e'. Setting the virtual camera position to e' thus results in a method in which the virtual camera matches the head tracker orientation exactly and the head tracker position in x and z, and in which the avatar eye midpoint matches the virtual camera exactly.

2.2.2 Pivot-and-Stretch Method

The previous method deviates from principle 1 in Section 2.1 by not using all of the head-tracking data. To do so there is a necessary trade-off with principle 2, as the avatar must be manipulated beyond the rotation of a single bone. The simplest such manipulation is to stretch one bone, enabling a perfect match of the avatar eye midpoint to the head-tracker position while maintaining the correct head rotation.

To do so, e' is set equal to t, and h' to e' - **he**. To match h exactly with h', **ph** must be adjusted by Δph such that it has the same length as $\mathbf{ph}'$ (Figure 2). To adjust $|\mathbf{ph}|$ by Δph, p's length must be adjusted by $\Delta ph/(\mathbf{ph} \cdot \mathbf{p})$. Once **ph** and $\mathbf{ph}'$ have the same length, p is rotated such that h matches h', and the head-tracker rotation is reapplied such that e matches e', and therefore t. Setting the virtual camera position to e' results in a method in which the virtual camera matches the head tracker orientation and position exactly, and in which the avatar eye midpoint matches the virtual camera exactly.

3 Conclusion

The benefit of the pivot-and-project method is that the only avatar manipulation is the rotation of a single bone, however the vertical component of the head tracker is not used. The benefit of the pivot-and-stretch method is that it uses all of the head-tracker data, although it does require an additional manipulation of the avatar by stretching the pivot bone. From our experience with both novice and expert VR users, this stretching is not noticeable.

Although developed to enable integration of body-tracking and head-tracking data, both techniques have also been successfully employed in virtual-embodiment applications where no body tracking is used, e.g. sitting in front of a virtual mirror, enabling the user to lean forward, backward, and side to side while maintaining correct head tracking.

Future work includes a quantitative assessment of the improvement of these techniques over using just body tracking, and user studies to determine differences with respect to user preference, simulator sickness, and virtual embodiment.

4 Acknowledgments

This work was funded under the European FP7 FET Integrated Project VERE (contract number 257695) and Mel Slater's ERC Project TRAVERSE (227985).

References

[1] V. I. Petkova and H. H. Ehrsson. If I were you: perceptual illusion of body swapping. *PloS ONE*, 3(12), Dec. 2008.

[2] M. Slater. Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. *Philosophical Transactions of The Royal Society B*, 364(1535):3549–3557, Dec. 2009.

[3] M. Slater, D. Perez-Marcos, H. H. Ehrsson, and M. V. Sanchez-Vives. Inducing illusory ownership of a virtual body. *Frontiers in Neuroscience*, 3(2):214–220, Sept. 2009.

[4] M. Slater, B. Spanlang, and D. Corominas. Simulating virtual environments within virtual environments as the basis for a psychophysics of presence. *ACM Transactions on Graphics*, 29(4), July 2010.

[5] K. M. Stanney. *Handbook of Virtual Environments: Design, Implementation, and Applications*. Lawrence Erlbaum Associates, Mahwah, NJ, 2002.

[6] N. Yee and J. Bailenson. The proteus effect: the effect of transformed self-representation on behavior. *Human Communication Research*, 33(3):271–290, July 2007.