The 3D Interaction Tool: A Pointing Device for Virtual Reality Applications

Lukas Gemar

March 26, 2016

Abstract

The 3D Interaction Tool is a hand-held device that enables a user to interact with virtual reality applications. Since both the position and orientation of the tool are tracked, user input with the tool accomplishes tasks such as selecting, translating, and rotating three-dimensional objects, or free-hand drawing in space. The tools position is tracked by a software application that filters observations from a camera attached to the virtual reality display, and the orientation is found by means of observations from an inertial measurement unit attached to the tool itself. Simple computer-aided design and drawing applications demonstrate the capabilities of the 3D Interaction Tool.

1 Nomenclature

Position vectors definitions in the inertial coordinate frame:

$\vec{p}_s \equiv$ position of the spherical tracking target on tool rear

$\vec{p}_r \equiv$ position of the tool rear

$\vec{p}_f \equiv$ position of the tool front

$\vec{p}_c \equiv$ position of the tracking camera

$\vec{p}_d \equiv$ position of the virtual reality display

Rotated position vectors:

$\vec{p}_s' \equiv$ position of the spherical tracking target on tool rear, camera coordinates

$\vec{p}_r' \equiv$ position of the tool rear in display coordinates

$\vec{p}_f' \equiv$ position of the tool front in display coordinates

Attitude matrices:

$\mathbf{A}_c \equiv$ attitude of the tracking camera

$\mathbf{A}_d \equiv$ attitude of the virtual reality display

$\mathbf{A}_t \equiv$ attitude of the interaction tool

$\mathbf{A}_t' \equiv$ attitude of the interaction tool in display coordinates

Body-referenced quaternion representations of attitude:

$\overline{\mathbf{q}}_c \equiv$ attitude of the tracking camera

$\overline{\mathbf{q}}_d \equiv$ attitude of the virtual reality display

$\overline{\mathbf{q}}_t \equiv$ attitude of the interaction tool

Position error covariances in inertial coordinate frame:

$\mathbf{P}_{ps} \equiv$ covariance of spherical tracking target position

$\mathbf{P}_{pr} \equiv$ covariance of the position of the rear of the interaction tool

$\mathbf{P}_{pf} \equiv$ covariance of the position of the front of the interaction tool

$\mathbf{P}_{pc} \equiv$ covariance of the position of the tracking camera

$\mathbf{P}_{pd} \equiv$ covariance of the position of the virtual reality display

Attitude error covariances in the inertial coordinate frame:

$\mathbf{P}_{ac} \equiv$ covariance of the tracking camera attitude quaternion error

$\mathbf{P}_{ad} \equiv$ covariance of the display attitude quaternion error

$\mathbf{P}_{at} \equiv$ covariance of the interaction tool attitude quaternion error

Position error covariances in rotated body coordinates:

$\mathbf{P}_{ps}' \equiv$ covariance of the sphereical tracking target position

$\mathbf{P}_{pr}' \equiv$ covariance of the position of the rear of the interaction tool

$\mathbf{P}_{pf}' \equiv$ covariance of the position of the front of the interaction tool

Attitude error covariances in the rotated body coordinates:

$\mathbf{P}_{ac}' \equiv$ covariance of the tracking camera attitude angular error

$\mathbf{P}_{ad}' \equiv$ covariance of the display attitude angular error

$\mathbf{P}_{at}' \equiv$ covariance of the interaction tool attitude angular error

2 Introduction

2.1 Motivation

As much as \$25 billion could be spent on virtual reality (VR) and augmented reality (AR) software applications by 2025 [1]. There are two primary modes of user-input that are possible for these

applications: indirect and direct. Indirect input devices translate the user's physical movements into the motion of a virtual cursor by means of a fixed transformation. Common examples of indirect input devices are a mouse or joystick. Direct input devices also translate physical to virtual movements, but this transformation is one-to-one. For example, a touch screen is a direct input device because the physical location of the user's press is colocated with the virtual click.

With the advent of virtual reality applications the following question must be answered: which of these input paradigms – indirect or direct – will prove most useful for interacting with virtual reality applications? While indirect input devices have achieved widespread popularity for two-dimensional displays, the direct input paradigm for VR applications embraces the immersiveness of VR experiences by enabling intuitive interactions for users. Direct input devices allow users to reach out and interact with the objects of their virtual space, just as they would with objects in their physical space.

2.2 Background

2.3 Usability Objectives

The usability requirements presented in Figure 1 must be met for the tool to succeed as an *input* device for VR software applications.

Usability		
Objective	Justification	Target
Good portability	Target scenarios are seated applications for home and workplace.	The interaction tool setup should be easily translatable between uses.
Low configuration overhead	----	Limited setup time.
Natural form factor	The interaction tool should not require the user to learn a new interaction paradigm.	Form factor similar to a pen or stylus.
Large tracking volume	The user should be able to use the interaction tool over their full range of movement.	Tracking volume of about 1m³.
Low cost	----	Less than $500.

Figure 1: Usability design objectives

2.4 Performance Objectives

The performance requirements presented in Figure 2 must be met for the tool to succeed as a *direct* input device for VR software applications.

Performance		
Objective	Justification	Target
High Accuracy	Tracking apparatus must translate user movements into direct input signals.	10cm or less.
Very High Precision	High precision ensures that cursor movements are repeatable.	1-2cm.
Low Latency	Too much discrepancy between physical and virtual motion of the tool induces simulator sickness.	Less than 60ms.
High Robustness	The system should be usable in non-ideal environments and use-cases.	Robust to tool velocity.

Figure 2: Performance design objectives

3 Tracking System Design

3.1 Requirements

In order to enable direct input with the interaction tool, the tool's tracking system must estimate both the position and orientation of the interaction tool. Additionally, rendering a virtual representation of the tool requires measuring its position and orientation relative to the virtual reality display orientation, given by the attitude matrix $\mathbf{A}_d$. To track the tool's absolute trajectory, its position is determined relative to the world – that is, relative to a non-accelerating, inertial coordinate system in the user's physical environment. The position and orientation of the tool in the world coordinates of the inertial frame are $\vec{p}_f$ and $\mathbf{A}_t$, respectively. The position of the tool in the coordinate frame of the virtual reality display is denoted by a 'prime' sign: the tip or front of the tool in display coordinates is $\vec{p}'_f$. Thus, there are four tracking targets: $\vec{p}_f$, $\mathbf{A}_t$, $\vec{p}'_f$, and $\mathbf{A}_d$; these are shown in Figure 3.

Tracking Requirements:
$\vec{p}_f$ ≡ position of the front of the tool in world coordinates
$\vec{p}'_f$ ≡ position of the front of the tool in display coordinates
A_t ≡ orientation of the tool
A_d ≡ orientation of the VR display

O_w ≡ origin of world coordinate frame
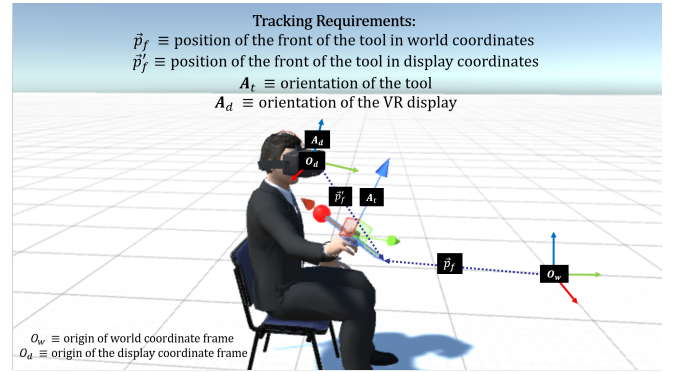O_d ≡ origin of the display coordinate frame

Figure 3: Position and orientation tracking requirements

3.2 Geometry

The geometry of the tool tracking system affects both the usability and the performance of the interaction tool. The display-referenced geometry is chosen because it eliminates the burden on the user to pre-calibrate their environment before providing direct input to VR applications. At the same time, this geometry is compatible with VR applications with which a user interacts while seated.

3.2.1 Option 1: World-referenced geometry

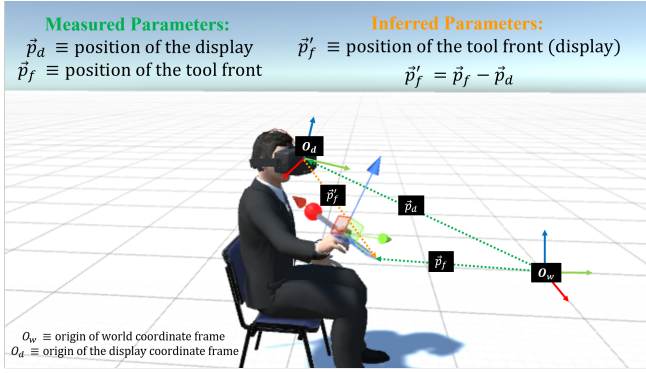The world-referenced tracking geometry is shown in Figure 4.

Figure 4: Position tracking with a world-referenced geometry

3.2.2 Option 2: Display-referenced geometry

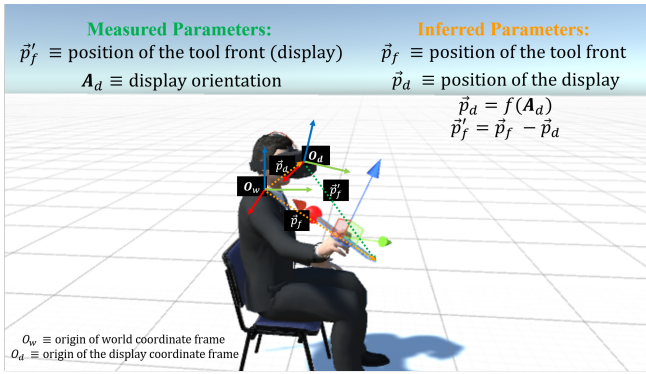The display-referenced tracking geometry is shown in Figure 5.

Figure 5: Position tracking with a display-referenced geometry

3.2.3 Geometry Selection: Comparing portability and tracking volume

Tracking geometries 1 and 2 have advantages and disadvantages when evaluated using the usability design objectives. Relative to the usability criteria, the two geometries differ greatly in their portability and tracking volume.

The world-referenced tracking geometry requires a fixed coordinate system in the user's physical environment, decreasing portability by increasing the configuration overhead for using the direct input device. However, once this coordinate system is established, the user can provide direct input to a VR application over a very large tracking volume, up to the size of the room in which the tracking system is located. The display-referenced geometry, on the other hand, has the advantage of enabling a user to

provide direct input to a VR application without pre-calibrating the physical environment in which they are using the VR display. The tradoff is that the tracking volume is small. Since the origin of the world coordinate system in the display-referenced geometry is located beneath the user's head, the user must remain seated while using a direct input device with a display-referenced tracking geometry.

3.3 Technology

A combination between optical and inertial sensors was chosen to track the 3D Interaction Tool's position and orientation. The chart in Figure 6 summarizes the comparison between various alternatives for the choice of the tracking technology.

Tracking Technology	Advantages	Disadvantages
Mechanical	Very high **accuracy** and **precision**. Good **portability**.	Small **tracking volume**.
Magnetic	Very high **accuracy** and **precision**. Low **latency**. Good r**obustness**.	Very high **cost**.
Ultrasound	High **accuracy** and **precision**.	Poor **portability**. Incompatible with slender **form factor**. Poor **robustness** to acoustic noise.
Optical	High **accuracy** and **precision**. Low **configuration overhead**. Low **cost**.	Poor **robustness** to occlusions and tool velocity. High **latency**. Narrow **field of view**.
Inertial	High **accuracy** and **precision** of orientation. Very low **latency**. Low **cost**.	Low **accuracy** and **precision** of position over large time scales.

Figure 6: Comparison of five tracking technologies

3.4 Final Design

3.4.1 Tracking measurements

Not all of these three tracking targets are measured directly. The tracking targets are inferred from the measurements of the position of the rear of the interaction tool relative to the camera, the attitude of the camera, and the attitude of the tool. The position of the tool relative to the camera is denoted by $\vec{p}'_s$, the attitude of the camera relative to the inertial coordinate system by $\mathbf{A}_c$ and the attitude of the tool relative to the inertial coordinate system by $\mathbf{A}_t$. The vector $\vec{p}'_s$ is measured by finding the .

3.4.2 Inferring the tool position

3.4.3 Inferring the tool attitude

$$\vec{p}_r = \mathbf{A}_c^{-1}\vec{p}'_s - \vec{p}_c$$

$$\vec{p}_f = \vec{p}_r + \mathbf{A}_t^{-1}\vec{l}$$

$$\vec{p}'_f = \mathbf{A}_d(\vec{p}_f + \vec{p}_d)$$

$$\mathbf{A}'_t = \mathbf{A}_d\mathbf{A}_t^{-1}\vec{l}$$

6 Performance Analysis

$$\mathbf{P}'_{pr} = \mathbf{P}'_{ps} + \left[\ \vec{p}'^{\,X}_s\ \right] \mathbf{P}'_{ac} \left[\ \vec{p}'^{\,X}_s\ \right]^T$$

$$\mathbf{P}_{pr} = \mathbf{A}_c^{-1}(\mathbf{P}'_{ps} + \left[\ \vec{p}'^{\,X}_s\ \right] \mathbf{P}'_{ac} \left[\ \vec{p}'^{\,X}_s\ \right]^T)(\mathbf{A}_c^{-1})^T + \mathbf{P}_{pc}$$

$$\mathbf{P}_{pf} = \mathbf{P}_{pr} + \mathbf{A}_t^{-1} \left[\ \vec{l}^{\,X}\ \right] \mathbf{P}'_{at} \left[\ \vec{l}^{\,X}\ \right]^T (\mathbf{A}_t^{-1})^T$$

$$\mathbf{P}'_{pf} = \mathbf{A}_d(\mathbf{P}_{pf} + \mathbf{P}_{pd})\mathbf{A}_d^{\,T}$$

$$\mathbf{P}'_{pf} = \mathbf{P}'_{ps} + \left[\ \vec{p}'^{\,X}_s\ \right] \mathbf{P}'_{ac} \left[\ \vec{p}'^{\,X}_s\ \right]^T + \mathbf{A}'_t \left[\ \vec{l}^{\,X}\ \right] \mathbf{P}'_{at} \left[\ \vec{l}^{\,X}\ \right]^T (\mathbf{A}'_t)^T$$

7 Conclusion

8 Future Work

References

[1] J. Verhage, "Goldman Sachs Has Four Charts Showing the Huge Potential in Virtual and Augmented Reality." `http://www.bloomberg.com/news/articles/2016-01-13/goldman-sachs-has-four-charts-showing-the-huge-pote` 2016. Accessed: 2016-03-19.