Massimiliano Di Luca
Max Planck Institute for Biological Cybernetics
Department of Human Perception, Action and Cognition
Tübingen 72076 Germany
and
Brown University
Providence,
Rhode Island 02912

New Method to Measure End-to-End Delay of Virtual Reality

Abstract

A virtual reality (VR) system tracks one or more objects to generate the depiction of a virtual environment from the user's vantage point. No system achieves this instantaneously: changes in the depicted virtual environment are delayed from changes in the position of the objects being tracked. In this paper, a method is proposed to quantify this time difference, the end-to-end delay of the VR system. Two light-sensing devices and two luminance gradients are used to simultaneously encode the position of one tracked object and its virtual counterpart. One light-sensing device is attached to the tracked object and it captures light from the gradient in the physical environment. The other device captures light from the gradient in the virtual environment. A measurement is obtained by moving the tracked object repetitively (by hand) across the gradient. The end-to-end delay is the asynchrony between the signals generated by the two light-sensing devices. The results collected with oscillatory movements performed at different frequencies indicate that for some VR systems, the end-to-end delay might not be constant but could vary as a function of the oscillation frequency.

1 Introduction

Virtual and augmented reality systems create the perception of a virtual element. This element is either the entire environment for virtual reality systems or specific objects of the environment for augmented reality systems (both virtual and augmented reality systems will be termed VR systems). VR systems based on the presentation of visual information (the ones that will be treated here) capture the user's head position to generate a depiction of the virtual element from the user's vantage point. In some cases, other parts of the user's body (e.g., the user's hand) or objects of the physical environment (e.g., a tool) are also captured. This is done by attaching a tracked element to the body part or to the object. One of the major limitations of today's VR technology is that the depiction of the virtual element is necessarily delayed with respect to the time at which the position of tracked element was obtained. As a consequence of this delay, the depiction no longer corresponds to the user's vantage point. End-to-end delay is defined here as the time required for the display to be updated according to the position of the tracked object.

One of the major challenges with the current state of the art technology is to have a VR system with low end-to-end delay. End-to-end delay can induce sim-

Presence, *Vol. 19, No. 6, December 2010,* 569–584

*Correspondence to max@tuebingen.mpg.de.

ulator sickness due to sensory mismatch between ocular and vestibular information (Kennedy, Lane, Berbaum, & Lilienthal, 1989; Ehrlich, 1997), it can reduce the subjective sense of presence (Held & Durlach, 1991), it can change the pattern of behavior such that users make more errors during speeded reaching, grasping, or object tracking (Ellis, Adelstein, Baumeler, Jense, & Jacoby, 1999; Teather, Pavlovych, Stuerzlinger, & MacKenzie, 2009; Ware & Balakrishnan, 1994; Watson, Walker, Ribarsky, & Spaulding, 1998), and can change the way multisensory information is combined into a percept (Di Luca, Knörlein, Ernst, & Harders, in press). The first step to try to reduce the end-to-end delay of a VR system is to understand where it is coming from. Here, the system will be divided into three major components.

- **Tracking**. The tracker captures a geometric property in the physical world, such as the position and orientation of the HMD or of the user's hand.
- **Processing.** The graphic system handles the interaction between the tracking data and the simulation of the laws of projection to produce the image of the virtual element.
- **Displaying**. The image is displayed.

All three of these components contribute to the end-to-end delay. If each of them were to introduce a constant delay, the VR system as a whole would reproduce the change in the virtual element after a specific interval of time. However, some of the components of the VR system (e.g., the tracking component) contain filters that make the amount of end-to-end delay dependent on the movement. The components could also exhibit a delay that depends on the particular simulation used (i.e., the graphic system might slow down with increasing complexity of the virtual element). Thus, there can be variable delay of the output signal (the rendering of the virtual element) depending on the characteristics of the input signal (the position of the tracked element).

To assess the end-to-end delay of a VR system it is important to establish whether such a delay is constant or whether it depends on the movement of the tracked element. To model this dependency, one could treat the VR system as an unknown nonlinear system and determine its transfer function: the function that relates the input signal to the output signal for signals at all possible frequencies. The problem can be simplified by treating the VR system as a linear system whose transfer function can be modeled as the gain and phase shift for each of the frequencies composing the input signal. Moreover, since the goal of this analysis is limited to finding out the origin of the end-to-end delay, to reduce the complexity of the problem even more it is possible to disregard gain and simply consider the phase shift between input and output. In other words, the phase shift at each frequency composing input and output signals should be measured. For the purposes of the measurement method proposed here, human-generated oscillatory movements of the tracked object with a predominant frequency component are used to create the input signal. By sampling the frequency range considered (ideally this range should exceed the range of frequencies produced during normal utilization of the system) it is possible to assess whether the VR system as a whole produces constant or frequency-dependent end-to-end delay. Moreover, by selectively modifying each of the components of the VR system, it is possible to test how the transfer function changes and to identify the contribution of each component of the system.

1.1 Related Work

Different methods have been proposed to perform measurements of end-to-end delay. Several of these methods utilize video cameras that record both a tracked object and the display that presents the rendered virtual object. The displacement of the tracked object with respect to its virtual counterpart is due to both the velocity of the tracked object and the end-to-end delay of the VR system. Since physical and virtual events are recorded synchronously, using this arrangement one can be assured that there is no differential delay added by the processing of the video stream in the measurement. Different research groups have measured the end-to-end delay of VR systems using this method by employing a tracker attached to a pendulum (Liang, Shaw, & Green, 1991; Steed, 2008), a chain that could be moved in a linear trajectory (Ware & Balakrishnan, 1994), a turntable (Swindells, Dill, & Booth, 2000; Kijima, Kitabayashi,

& Hayakawa, 2007), or a tracker moved by the user (He, Liu, Pape, Dawe, & Sandin, 2000). By filming or taking snapshots of the tracker and its virtual counterpart, it was possible to measure the image slip and thus estimate delay. Miller and Bishop (2002) adopted the approach of simultaneously recording the motion of the user's head and the displayed scene using two synchronized high-speed cameras. One camera was pointed at the scene (displayed with a projector) and a second camera filmed the user wearing the HMD. The two video streams were analyzed to find the amount of image motion. Delay was measured as the time elapsed between the stop of the user's head and the stop of the virtual scene.

The pros and cons of using cameras to measure timing differences between physical and virtual movements are listed as follows. The advantage of employing a camera is mainly that the measurement is relatively easy to perform once the setup has been built, and one can be confident that the recording of the movements is synchronous. However, this method requires specific hardware that can synchronously capture two streams of video at high refresh frequency or it requires an arrangement of the setup to film both movements using one camera (e.g., Steed, 2008). Moreover, in most cases, the tracked object and the simulation used in the measurement are not necessarily the same as those used in the final utilization of the system (e.g., as in the cases where the tracked object is moved mechanically). Furthermore, in many cases, the estimates of end-to-end delay have not been related to the type of movements performed (the works reported so far do not analyze the effect of movements on the amount of end-to-end delay of the VR system). In particular, the frequency of movement performed is not considered and the average value of delay is usually reported.

Another approach to measuring delay is to employ devices other than cameras. Typically, a light-sensing device is used to record events either in the physical or in the virtual environment, or in both. In Mine's method (1993), which is frequently employed for its simplicity (Akatsuka & Bekey, 1998; Liang, Shaw, & Green, 1991; Olano, Cohen, Mine, & Bishop, 1995; Teather et al., 2009; Ware & Balakrishnan, 1994), one light-sensing device is used to register when a pendulum reaches the lowermost point. A tracker is attached to the pendulum and its position is rendered in VR. Another light-sensing device registers the lowermost point of the virtual counterpart of the tracker. Despite the inherent simplicity, one problem of this method is that the rendered scene and the tracked object during measurement could differ from normal utilization and, most importantly, in the studies that employed it only one type of movement has been used for the recordings. Pendulums of a given length, for example, always oscillate at the same frequency (i.e., Teather et al., 2009 reported that the oscillation frequency was 0.8 Hz) and the authors do not report whether they changed length for different measurements. Adelstein, Johnston, and Ellis (1996) noted this limitation and thus used a motor-actuated arm to oscillate a tracker at different frequencies. They compared the physical trajectory of the tracker with the one registered by the tracking device using a Bode-plot analysis to assess the transfer function of the device. Although this type of measures accurately captures the frequency-dependent characteristic of the delay, the assessment is limited to the tracking device and does not measure the end-to-end delay of a VR system (i.e., graphic processing and display are not measured by this method). Other limitations are that a motorized device is required (human-generated movements are more irregular and thus can affect tracker performance) and that the delay in the recording of physical and tracked position might be different. The method described here is very similar to the one of Adelstein et al. as it allows measuring end-to-end delay at different frequencies. However, movement is generated by hand and both physical and virtual trajectories are synchronously captured using the same type of device.

1.2 Method Description

While it is relatively easy to acquire position in physical space and track an object's trajectory, to perform a measurement of the end-to-end delay of a VR system (that includes tracking, processing, and displaying) it is important to find a way to synchronously acquire position in physical space and in virtual space. For

this purpose, two light-sensing devices and two luminance gradients are employed (images containing a smooth transition from white to black). Each light-sensing device measures the brightness at one point of a gradient. The real gradient is obtained by displaying a static image on a display. The virtual gradient is obtained by placing a textured rectangular surface in the virtual scene. The signal produced by the sensor is proportional to the position along the gradient, thus the recording captures one component of the position of the object that it is attached to. By having two aligned gradients in real and in virtual space, it is possible to use the same method to synchronously acquire the two positions along the gradients. This method was devised by Bruce Kay (personal communication) and has been employed in Knörlein, Di Luca, and Harders (2009) and in Di Luca et al. (in press).

Henceforth, the display of the VR system will be called "VR display" to differentiate it from the "flat panel display" used to show the static gradient. Depending on the conditions, it is possible to avoid the use of a flat panel display by displaying the fixed real gradient juxtaposed with the moving virtual gradient on the VR display. This arrangement should be preferred as the characteristics of the signals captured by the light-sensing devices will be most similar. The only case where the use of a flat panel display is necessary is when the VR system employs an HMD.

The gradients can be oriented in any direction, but here we will consider the case of a horizontal movement orthogonal to the viewing direction as shown in Figure 1. Movements, however, do not have to be performed exclusively in the horizontal direction: there can be vertical movements as well. On the other hand, movement away from the gradients should be kept to a minimum so as to capture only a small area of the screen. Alternately, it is possible to use an occlusion screen to decrease the viewing angle of the sensing devices. In this case, it is possible to also perform rotations around an axis orthogonal to the gradient.

Note that the addition of the gradient might occlude some part of the VR scene such that there could be a decrease in the computational load. There are different approaches to address the issue. The most trivial one is to use blending and make the surface with the gradient slightly transparent. Another method is to add the surface with the virtual gradient texture only after having rendered the scene, having cleared the buffers, or having manually turned off the depth test. If raytracing is used, the transparency might not be sufficient to bring back the computational load to normal levels. In all other cases, the transparency method should be preferred because clearing the buffers or changing depth test might create unwanted side effects in terms of delay.

1.3 Types of Measurement

To apply the method to measure end-to-end delay of VR systems, a taxonomic scheme is defined which depends on the object being tracked. Three possible scenarios are conceived, where the position of the *display*, the *head*, or an *object* is acquired by the VR system and the corresponding measurement of end-to-end delay is performed (see Figure 1).

Display movement for end-to-end delay measurements can be used in VR systems where the display that shows the virtual scene can move. The position of the VR display is tracked so that the view of the scene changes with its position and orientation. Usually, this is obtained by mounting the display on the head (HMD), but also setups that use other types of moveable displays fall in this category. To perform display end-to-end delay measurements, both light-sensing devices are attached to the display (see Figure 1, top row). One sensor points to the flat panel display with the static image of the gradient (it captures the position of the VR display across the real gradient). The other sensor points to the VR display (it is pointing "into" the virtual world where the virtual gradient is, so it captures the relative position between the camera used for the rendering and the virtual gradient). In Figure 2, an example of such an arrangement is given for a VR setup employing an HMD. Here, by moving the HMD, the light that is captured by the sensing device pointing to the real gradient changes. The image of the virtual scene on the VR display is also updated and the image of the virtual gradient on the VR display moves in a direction opposite to the movement of the VR display itself. As a consequence,

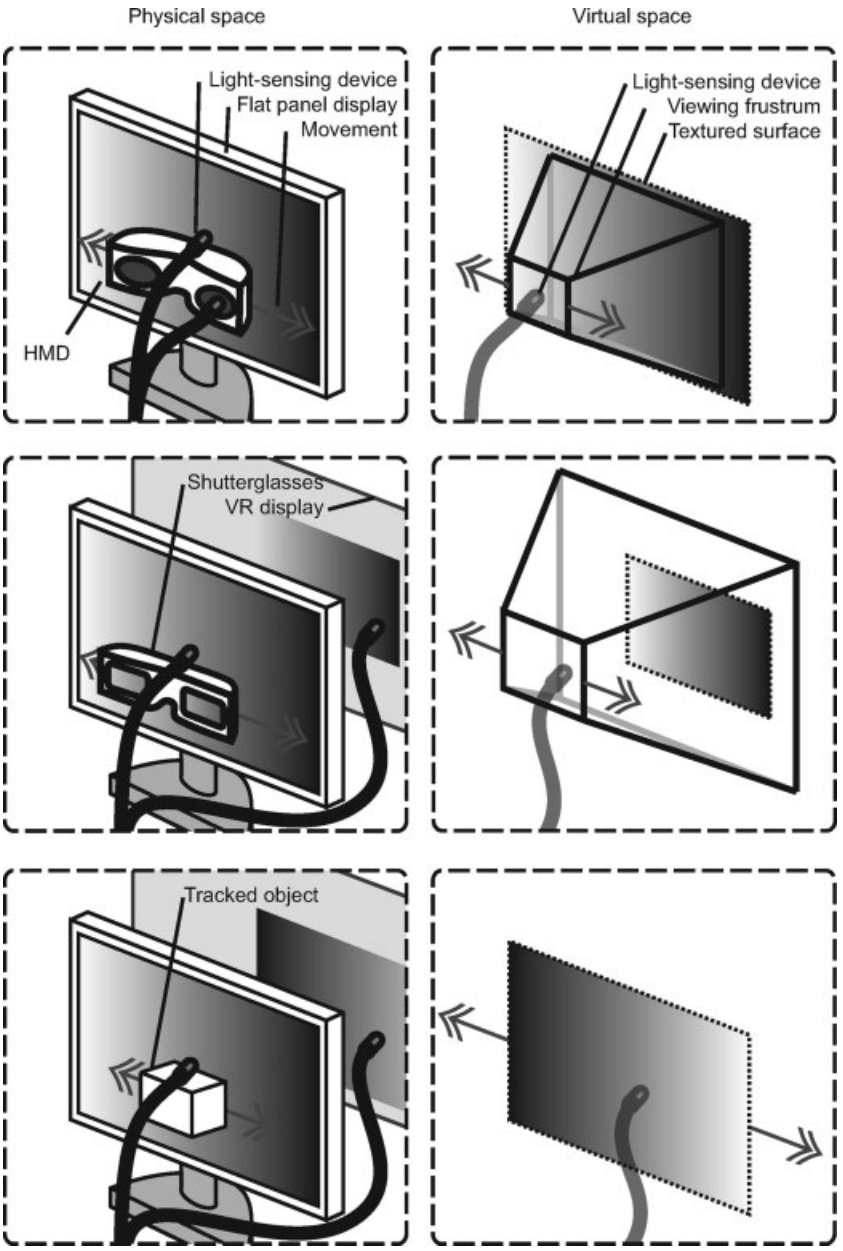

Figure 1. *Arrangement of the materials in real and virtual space to measure the three types of end-to-end delay. (Top) display; (middle) head; (bottom) object.*

the light captured by the sensing device pointing to the virtual gradient will change, but such a change is deferred by the end-to-end delay. Note that the two gradients should be oriented in the same way (with corresponding orientation of the dark and bright parts) to produce a similar change in light for the two sensing devices. By performing repetitive movements with the VR display in front of the gradients, the repetitive signals captured by the two sensing devices can be compared to obtain an estimate of the end-to-end delay.

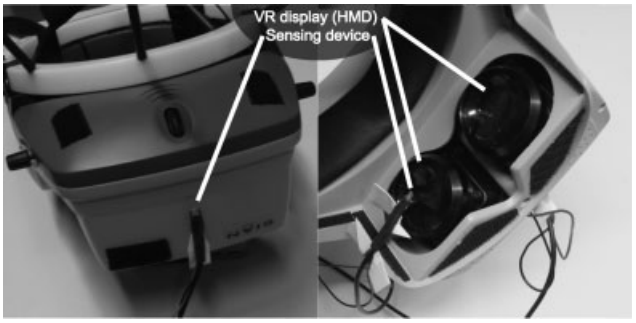

Figure 2. *Position of the two light-sensing devices on an HMD to perform an end-to-end delay measurement for display movements. (Left) Sensing device attached to the outside casing of the HMD and pointing to the flat panel display showing the real gradient. (Right) View from the bottom of the HMD showing the sensing device attached to the VR display and pointing to the virtual gradient.*

Head movement for end-to-end delay measurements can be used in VR systems where the view of the scene changes as the user's head moves, but where the VR display's location is fixed. Typical examples are CAVE systems and desktop VR systems where participants wear shutterglasses whose position is tracked. To measure end-to-end delay in these cases, as shown in Figure 1, middle row, one sensor is attached to the tracked shutterglasses and points to the flat panel display (thus capturing the position of the tracked head across the real gradient). The other sensor is attached to the VR display (thus capturing the position of the virtual counterpart of the user's head across the virtual gradient). To perform this measurement, the flat panel display can be positioned facing either the VR display or the same direction as the VR display (see Figure 3 for a comparison). The gradients here should be oriented to have the same orientation (see Figure 1, middle row). Another possibility is to position the flat panel display orthogonal to the VR display to capture movements in depth (to move away from the VR display). In this case, the virtual gradient has to be produced along the depth dimension. This can be done, for example, by making the color of a polygon proportional to the distance (or alternatively one can render a series of parallel polygons using near-plane clipping to make the closest ones disappear with forward movements).

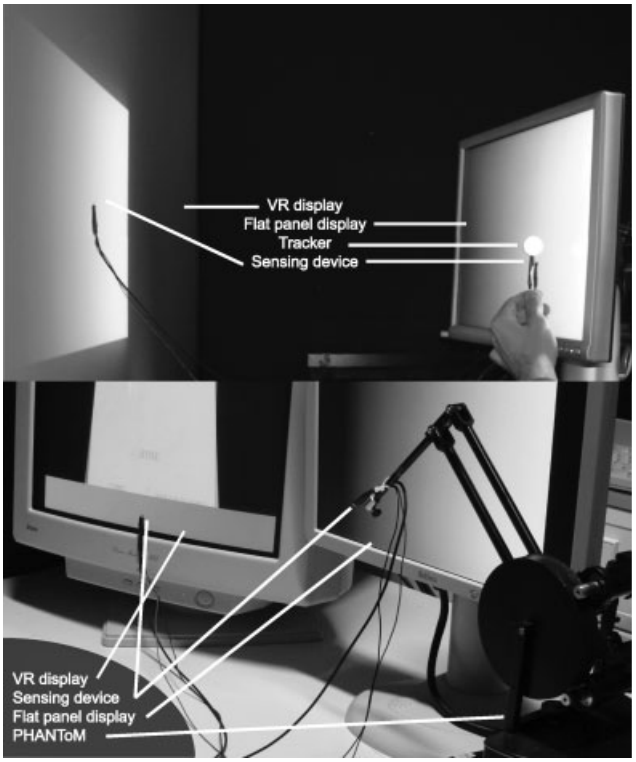

Figure 3. *Position of the two light-sensing devices used to perform an end-to-end delay measurement for a moving object. (Top) Arrangement for the measurement in a VR system using a back-projection screen where the flat panel display is facing the VR display. (Bottom) Arrangement for the measurement of end-to-end delay in the Table-top Virtual Workbench setup, where a PHANToM is used to track a participant's finger (see Section 2.3 for details). The flat panel is facing the same direction as the VR display.*

Object movement for end-to-end delay measurements can be used in addition to other measurements for systems that track either the head or the display. This type of measurement is necessarily employed in VR systems where there is a fixed view of the VR scene displayed on a fixed VR screen (only the view of the virtual element corresponding to a tracked object, not of the entire scene, changes in response to movement in the real world). To do this measurement, the position of the virtual gradient is not fixed in the VR scene (as it was for the other two types of measurements), but the gradient is positioned in front of the virtual element that corresponds to the tracked object (see Figure 1, bottom row).

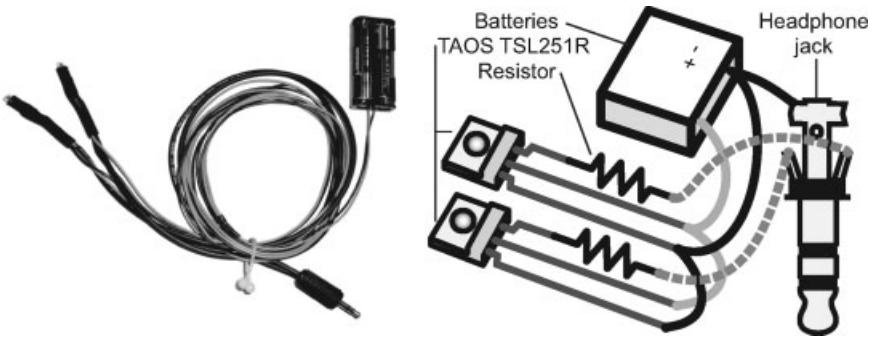

Figure 4. *(Left) Prototype circuit used for the measurements. (Right) Electric scheme of the circuit.*

One sensor is attached to the tracked object and points at the flat panel display (it captures the position of the object across the real gradient). The other sensor is attached to the VR screen (it captures the relative position between the virtual element and the camera). As in the case of the head movement measurements, the flat panel display can be oriented either in front of the VR display or facing the same direction (see Figure 3). As opposed to what happens for the other two types of measurements, however, the gradients should be oriented in opposite directions to produce the same change of luminance for a motion of the tracked object (see Figure 1).

2 Materials

The sensing device is made up of the following: two photodiode plus amplifier ICs TAOS TSL251R; two 10 KΩ resistors; two 1.5 V AA batteries with holder; electric wire 0.2 mm diameter red and black (1 m); coaxial audio cable (1 m); headphone jack (3.5 mm stereo plug); headphone extension cable (at least 2 m). Additional materials needed are: computer with two-channel synchronous A/D converter (in this study an audio card with a stereo line-in input is used for this purpose); flat panel display on which an image with a black to white horizontal gradient is displayed (characteristics of the flat panel display, i.e., refresh frequency, luminance, etc., should be similar to the VR display); VR scene with one slightly-transparent rectangular polygon with a black to white gradient texture (size and position roughly correspond to the flat panel display).

Since photodiodes are semiconductors that produce current when illuminated, it is possible to plug two photodiodes with near-visible spectrum response into the audio card line-in using a 3.5 mm plug and detect changes of light (phototransistors would not work since they do not provide sufficient voltage). The OSRAM BPW21 has been successfully used multiple times, including for the results on the VENLab (discussed shortly). In some cases, however, the brightness of the HMD screen is not sufficient to create a measurable signal. For this reason, it is better to use two identical diodes plus amplifier circuits such as the TAOS TSL251R. Figure 4 shows the circuit schematic and the picture of a prototype of the device used.

The circuit described above can be attached to the line-in audio input of a computer audio card using a headphone extension cable if necessary. For these measurements we used an Apple Macbook Pro, which is among the few laptop computers to have a stereo line-in input. The audio card is then used as an A/D converter that synchronously samples the two channels. The only major limitation in the use of an audio card is that it has a high pass filter with a cutoff at 20 Hz. The ability to record DC signals would allow the mapping between positions along the gradient and brightness level. For this reason, A/D converters without such filters should be preferred. This limit, however, is not a problem in practice as the displays have refresh frequencies typically above 60 Hz, which are not stopped by the filter. The

position along the gradient acts as an amplitude modulation of the refresh frequency, so that position can also be captured using an audio card.

2.1 Signal Analysis

Since the proposed method is based on capturing the position along the two parallel gradients, there can be only two sources of spatial misalignment: offset and different extension of the gradients. Due to the nature of the gradient, the only location offset that creates a change in the signal is along the direction of brightness change. In this case, the two recordings would differ by a constant factor k (as one sensing device captures more light than the other). Thus, without end-to-end delay, the signal captured by one sensing device s' would be related to the signal s produced by the other sensing device according to $s' = s + k$. On the other hand, a difference in the extension of the gradients produce signals with different modulation amplitude m in response to a certain movement. In this case, without end-to-end delay we would find that $s' = m\,s$. The effects of spatial misalignment are similar to the ones created by differences in brightness and contrast for the two displays, which would also create a relation between the signals such that $s' = m\,s + k$. Another possible difference between the signals is that the monitor γ deviates from unity so that one of the signals would become $s' = m\,s^{\gamma} + k$.

To avoid that such effects could influence the estimate of end-to-end delay, the recorded signals are processed using the same filters. First the signals are low-passed (about 10 Hz) to remove the discontinuous illumination (i.e., the refresh frequency of the display) and obtain the modulating signal related to position. Then, the signals are windowed (using a Gaussian window) to blend out the beginning and ending portions of the recording. The fast Fourier transform is computed to determine the frequency spectrum and signals are filtered using a band-pass filter centered on the frequency corresponding to the maximal frequency component. Cross-correlation is then computed to find the temporal shift (in discrete intervals corresponding to the sampling frequency of the signal) that maximizes the correlation coefficient.

Overall, the goal of the analysis performed is to extract the position signals and obtain an estimate of the most dominant frequency component and of the delay between the signals (the time shift that minimizes the difference between the two signals at the dominant frequency). This analysis is performed on every recorded pair of signals. The reader can obtain MATLAB code that performs this analysis (Di Luca, 2010).

2.2 Measurements Performed

The method proposed has been employed to test whether end-to-end delay is frequency dependent. For this, several setups have been analyzed: the TrackingLab, the Table-top Virtual Workbench, and the VENLab. The goal was not to provide a complete characterization of each setup, but to test the effectiveness of the proposed method. Display movements were used to record end-to-end delay in the TrackingLab and VENLab (Version 1). Object movements were used for the Workbench and for the VENLab (Version 2). Twenty recordings of 11 s each were performed for the TrackingLab and the Workbench setups; 12 recordings of 10 s each were performed for the VENLab setup (Version 1), and five recordings of 10 s each were performed for each of the tracked objects for the VENLab setup (Version 2). The length of recording should depend on the frequency of the oscillation, as it is necessary to capture the signal generated by several oscillations to obtain an estimate of end-to-end delay. In the current experiments we used 10–11 s, but when the frequency of oscillation is low (i.e., lower than 1 Hz) the recording could be lengthened to increase precision. The amplitude and frequency of the repetitive movement should be varied only between different recordings and should be kept constant in each one of them. This is because if there is a frequency dependent end-to-end delay, performing a single recording containing a wide range of movement frequencies would make this dependency undetectable. For this reason, only horizontal oscillatory movements spanning approximately 15 cm (for the TrackingLab and the VENLab) or 5 cm (for the Table-top Virtual Workbench) were performed to obtain a single movement frequency for each recording. Since the movement was not

constrained, the trajectories contained some component of movement also in the vertical and depth dimensions with respect to the flat panel display but they were intentionally kept to a minimum.

2.3 Setups

2.3.1 TrackingLab Version 1. The setup belongs to the Max Planck Institute for Biological Cybernetics (www.cyberneum.de) and consists of a 12 m × 12 m hall with an optical tracking system used to create a virtual environment by tracking the user's head and displaying images on a tethered HMD. The tracking system consists of 16 Vicon MX13 motion capture cameras (1.3 megapixels each, with a temporal resolution up to 484 Hz), which are mounted on the sidewalls and connected to a Vicon IQ 2.5. Five spherical markers are attached to the HMD with carbon rods, so that the position of the user's head can be tracked. The graphic processing is performed by a Dell XPS 420 equipped with an Intel Core2 Duo Processor e6850 3 GHz processor, 3 GB RAM, and an Nvidia Quadro FX 4600. The virtual environment created using Virtools 4.1.0.64 comprised only a textured floor and the gradient. The visualization is obtained using an NVIS nVisor SX with Mobile VCU with a resolution of 1280 × 1024 and refresh note of 60 Hz.

2.3.2 TrackingLab Version 2. In this configuration, backpack-mounted laptops are used for the rendering, allowing multiple users to share the same tracking space (Dell XPS Gen 2 laptop with an Intel Centrino 2.26 GHz, 1 GB RAM, and an Nvidia GForce2Go 6800 Ultra). The laptops receive tracker data through a WiFi connection. The HMDs are an eMagin Z800 3D Visor with a resolution of 800 × 600 and refresh rate of 60 Hz. A total of 12 objects are tracked contemporarily: two helmets with five spherical markers attached to carbon rods to capture head position, two backpacks with five markers each, four shoes with five markers, and four wristbands with four markers. For this version, the tracker is set to the default configuration (in particular, tracking data are sent only after 10 update cycles; see details in the result section). The VR environment comprises a fully furnished room and two full-body avatars.

2.3.3 Table-top Virtual Workbench. The setup belongs to the Max Planck Institute for Biological Cybernetics, was built by Marc Ernst and Volker Franz, and is described in detail in Ernst and Banks (2002). It consists of a computer monitor mounted slanting downward and viewed through a mirror. When observers look into the mirror, virtual objects appear to float above the table and the observer can virtually touch them with two fingers using a force feedback device. Two SensAble PHANToMs are placed below the mirror to provide haptic feedback. The tracking is also performed by the two PHANToMs to which (usually) the index finger and thumb are taped and can move within a space of about 30 cm × 30 cm × 30 cm. The PHANToMs update the tracked position of the fingers at 1000 Hz. Since the observer cannot see his or her fingers, two spheres are rendered at the tracked finger positions on the display. The graphic processing is performed by an SGI IP30 Octane workstation equipped with two 360 MHz R12 processors, 786 MB RAM, and an SGI Odyssey graphics card. The software is a custom C program that makes use of OpenGL and SensAble GHOST API. The visualization is performed on a Sony GDMF500R, the resolution is set to 1280 × 1024 and a refresh rate of 120 Hz, but the graphic update is 60 Hz since a pair of RealD CrystalEye shutterglasses is used to render stereo images.

2.3.4 VENLab. The setup at Brown University consists of a 12 m × 12 m room with a tracking system mounted on the ceiling and a tethered HMD (www.cog.brown.edu/research/ven_lab/index.html). The measurements are performed with two photodiodes (and not the photodiode-amplifier circuits), which limits the range of movements. The tracking system is a hybrid Inertial/Ultrasonic 6-DOF tracking system, Intersense IS-900. Two-hundred-fifty-five ultrasonic emitters are mounted on the ceiling and four microphones plus the inertial system are fixed on top of the HMD. A SGI Onyx2 with an Infinite Reality Engine using Sense8's World Tool Kit software performs the graphic processing. The visualization is obtained using a Rockwell Col-

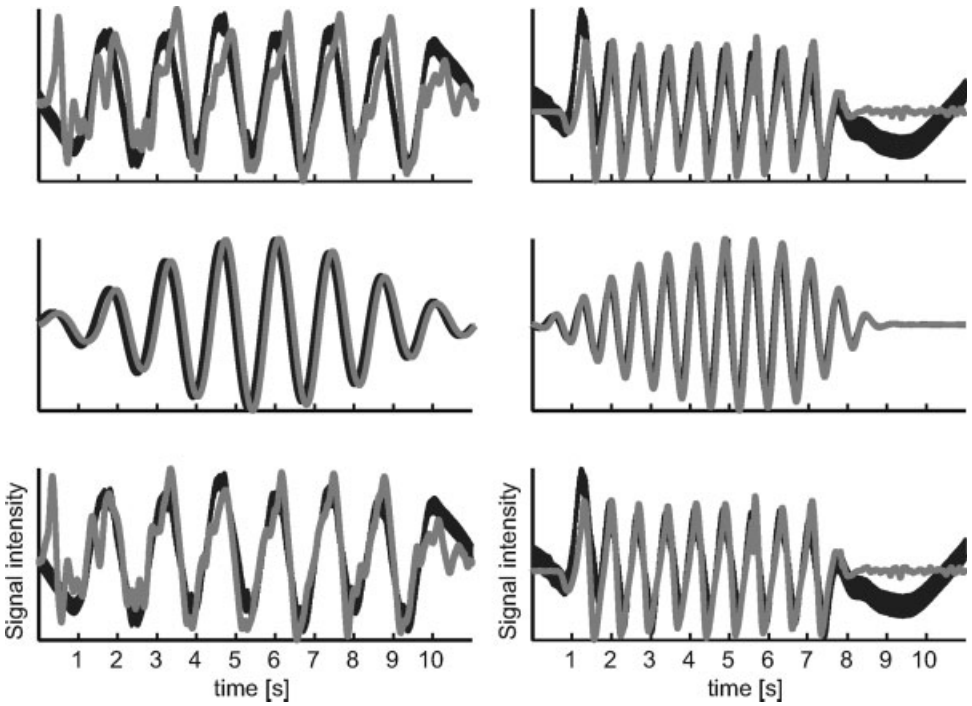

Figure 5. *Each column shows the analysis of a recorded pair of signals. The black line corresponds to the signal recorded from the real gradient and the gray line corresponds to the signal recorded from the virtual gradient. (Top) Recorded signals from which high frequencies have been removed. The values of the peak frequency in the spectrum are 0.7 Hz and 1.4 Hz respectively for the two signals. (Middle) Signals that have been windowed and filtered using a band-pass centered on the peak frequency in order to calculate end-to-end delay using cross-correlation. The estimated end-to-end delays obtained from cross correlation are 51 ms and 34 ms. (Bottom) Signals realigned to compensate for the estimated end-to-end delay.*

lins Kaiser Electro-Optics Proview SR-80A capable of a resolution of 640 × 480 and a refresh of 60 Hz. Note that the tracker (new IS-900 and software) and the HMD (Rockwell-Collins ProView SR80A, 1280 × 1024) as well as the graphics system in the VENLab have been updated since these data were collected.

3 Results

Two example recordings and the results of the signal realignment are shown in Figure 5. From the 20 recordings performed on the TrackingLab (Version 1) and Tabletop Workbench setups, it was possible to obtain an estimate of end-to-end delay and frequency only for 16 and 17 of them, as the remaining recordings contained recording artifacts (i.e., reaching the border of one of the gradients with the corresponding light-sensing devices). For the VENLab setup, the estimate was possible for 10 of the 12 recordings performed. The values obtained for these recordings are plotted in Figure 6 both in terms of temporal delay and in terms of phase delay between the input and output signals.

The pattern of end-to-end delay as a function of frequency of oscillation that is obtained with the three setups is markedly different. There is a constant end-to-end delay for the Table-top Virtual Workbench, which is 43.5 ms ± 5.1 ms and corresponds to 15.65 deg/Hz of variation in phase delay per Hz. On the other hand, Figure 6 shows that both the TrackingLab (Version 1) and, to a lesser degree, the VENLab setups have an end-to-end delay that changes as a function of the movement frequency. Data indicate that the TrackingLab setup

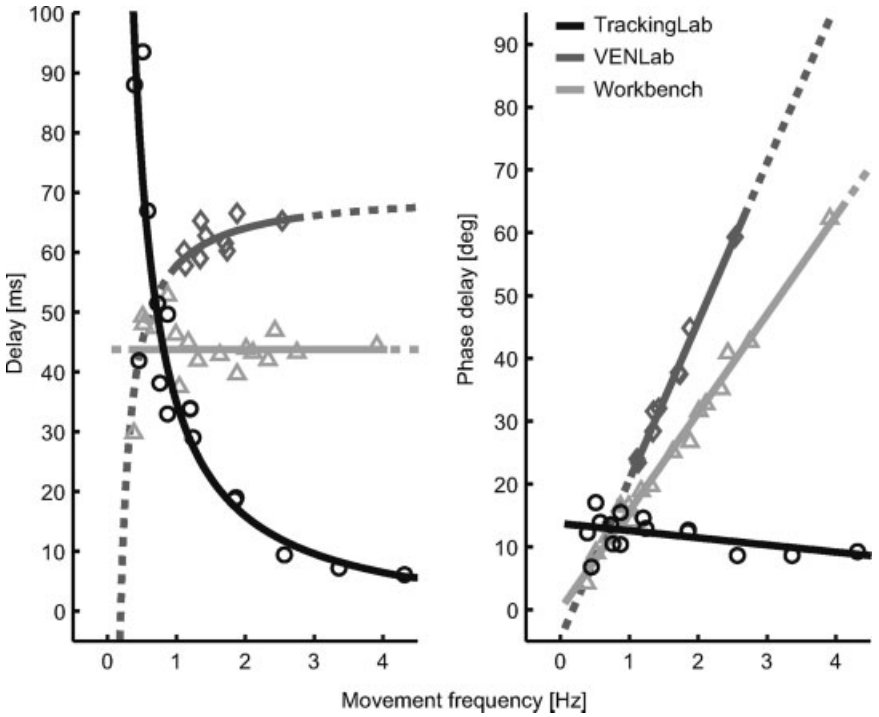

Figure 6. *Measurements obtained as a function of the frequency of movement in terms of time delay (left) and phase delay (right) for the three setups (TrackingLab Version 1, VENLab, and Workbench). The lines represent linear fits to the phase delay data.*

(Version 1) has a constant phase delay of $11.8^\circ \pm 2.8^\circ$ at different frequencies. Also for the VENLab setup there is an indication that the end-to-end delay depends on the type of movement, but the change of end-to-end delay is smaller and the pattern differs from the one of the TrackingLab (higher end-to-end delays are registered for the VENLab with faster movement frequencies).

To confirm that the frequency-dependent end-to-end delay found for some of the setups is not an artifact of the analysis performed and to isolate the cause of such a pattern of results, the tracking component of the TrackingLab setup was tested with additional recordings in configuration Version 2. Five measurements of 10 s each were performed with different objects. The results obtained with wristband, shoe, and helmet are compared in Figure 7 (five additional measurements were performed with the wristband). Overall the end-to-end delay of the TrackingLab obtained with this configuration is higher than the one in Version 1. Different factors contributed to increase the end-to-end delay: wireless transmission of the tracking data, lower performance of the computer used for the rendering, increase in the latency after which the tracking information was transmitted by the tracker, and contemporary presence of multiple objects to be tracked. Most importantly, however, the data collected show that there is a difference in the patterns of end-to-end delay recorded with different objects. Similar to the VENLab setup, end-to-end delay of the TrackingLab setup recorded with the wristband decreased at low frequencies of movement, while for other objects end-to-end delay remained constant across the spectrum of frequencies. The difference in the pattern of results obtained confirms that frequency-dependent end-to-end delay is not an artifact of the analysis performed (otherwise it should be present for every set of recordings). Moreover, since the only difference across the sets of recordings was the different tracked object, this result indicates that the cause of the frequency-

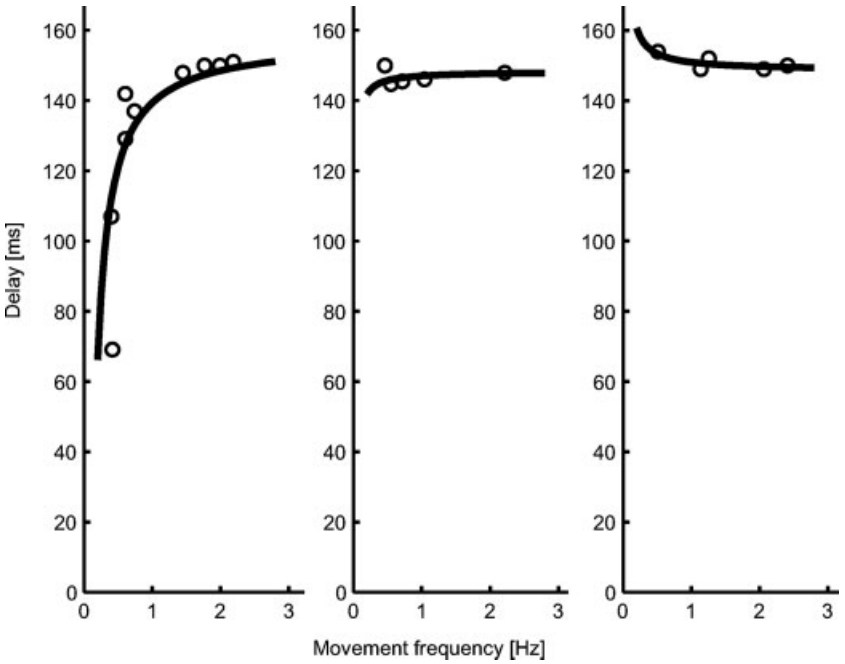

Figure 7. *Measurements of end-to-end delay obtained with different tracked objects in the TrackingLab setup (Version 2): (left) wristband, (middle) shoe, and (right) helmet. The lines represent linear fits to the phase delay data.*

dependent end-to-end delay in the TrackingLab is the tracking component of the system.

To test whether the proposed method produces reliable results, end-to-end delay was recorded for different parameters of the tracking device for the TrackingLab setup (Version 1). Here, the configuration of the tracking component was modified so that one of the parameters forced end-to-end delay in the tracking data and captured position and orientation of the HMD was sent only after a certain number of update cycles (10, 20, 30, and 100 update cycles with a frequency of update of about 145 Hz were used). This modification normally allows for a more precise capture of position in offline applications (i.e., motion capture and recording). To check whether this method can produce precise measurements with a limited number of recordings, only three measurements lasting 10 s each were performed at different frequencies of oscillation. The results shown in Figure 8 indicate that the precision of the single measurements performed is not very high; the average of such measurements, however, accurately captures the change in additional end-to-end delay. Thus, in normal circumstances, multiple measurements should be performed on the same system to obtain a precise estimate of end-to-end delay.

4 Discussion

A new method to measure end-to-end delay of VR systems has been proposed and tested. This method gives an estimate of end-to-end delay that includes all components of a VR system across a range of repetitive human-generated movements. To use this method, the following steps should be performed:

- Positioning of two luminance gradients in physical and virtual space so that they are colocated.
- Positioning of two light-sensing devices according to one of the schemes in Figure 1, so that each points to one gradient.

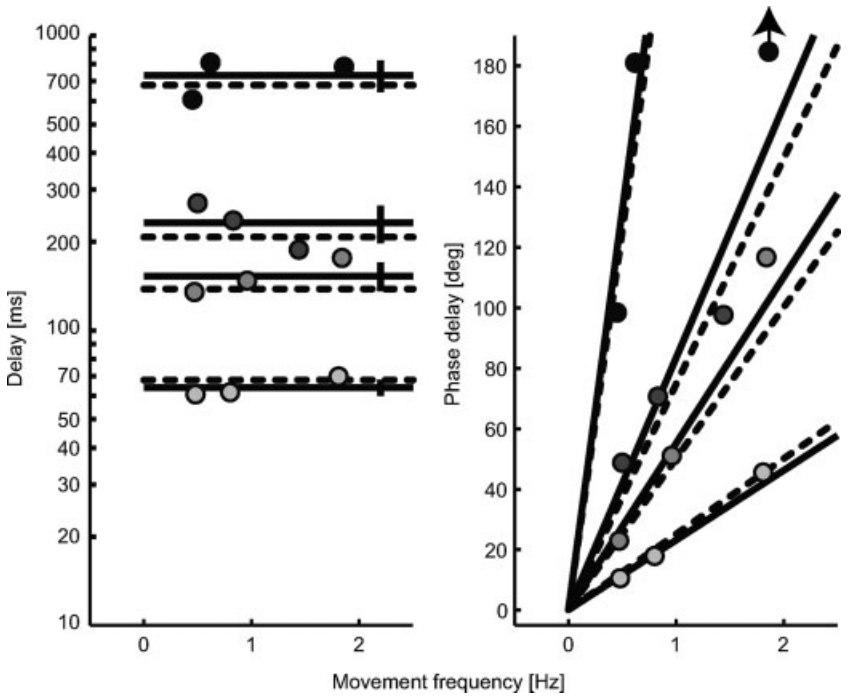

Figure 8. *Measurement of end-to-end delay (left) and corresponding phase delay (right) with delay added by the tracking system. The different colors correspond to measures with 10, 20, 30, and 100 update-cycle delay. The dashed lines represent the delay forced by the tracking device (about 70, 140, 210, 700 ms), the continuous lines are linear fits to the data, constrained to pass through the origin of the phase delay graph.*

- Recording of the signals generated by the two light-sensing devices while the experimenter moves the tracked object repetitively across the gradients. Recording should be repeated several times and movement frequency should span the spectrum of interest.
- Analysis of each pair of recorded signals to determine the frequency of repetition of the movement and the delay that maximizes the similarity between the two signals.
- From multiple estimates of frequency and end-to-end delay it is possible to determine whether there is a frequency-dependent component of the delay.

This method synchronously captures both physical and virtual position using the same mechanism, without using complex additional devices, or relying on discrete events. Rather, the method presented uses easily acquired hardware, is quick to set up, and requires minor software and hardware modification to the VR system. This method, moreover, can measure end-to-end delay with any repetitive movements of the object tracked.

Due to this characteristic, multiple measurements can be performed by systematically varying the movement, so that the end-to-end delay of the VR system can be related to some parameter of the trajectory of the tracked object. In this study, the proposed method has been employed to test whether end-to-end delay varies with frequency of repetition of the movement. The results obtained are in agreement with the ones by Adelstein et al. (1996) that—in some VR systems and for some configurations of the tracking device—end-to-end delay has a frequency-dependent characteristic. For some data sets, the end-to-end delay increases with frequency, while for others the delay stays constant or decreases. Adelstein et al. have found the same range of results

while testing different tracking hardware, but the amount of variation was smaller than the one registered here.

Variable end-to-end delay might seem surprising if one thinks of VR as a piece of software that generates an image at every graphic update. The end-to-end delay in this case would be constant and should not depend on frequency of movement. However, as described in the introduction, a VR system has many components, some of which are dynamic elements with their own transfer function. These components can be considered time-invariant linear systems whose transfer functions can be described by a gain and a phase delay at each frequency composing the input signal. Time delay in a frequency domain analysis adds phase delay proportional to the input frequency as $phase[\mathrm{deg}] = 360 \times freq[\mathrm{Hz}] \times delay[\mathrm{s}]$. This relationship captures very well the pattern of results obtained with the Workbench setup. In other words, the time delay in this setup is constant across movement frequencies. The data collected for the other two setups also show a phase shift that can be approximated by a linear relation of phase with frequency. However, there are deviations from constant time delay captured by the slope of the line. These deviations lead instead to linear fits with an intercept that is different from zero. The magnitude of the intercept is most pronounced for the data collected with the TrackingLab setup Version 1 (Figure 6), for which the phase shift is almost constant across the frequency range. Moreover, the data in Figure 7 indicate that the transfer function can change depending on the object tracked: even though the delay is equal at high frequency, there is a change in the intercept of the fitted line at 0. These two findings help in identifying the component that is responsible for the frequency-dependent delay, which in the TrackingLab is the tracking system (note that for the VENLab, the data collected do not prove which component is responsible). The tracking system is a priori the most likely component to have a frequency-dependent delay, as it often contains smoothing and predictive filters (see Adelstein et al., 1996; Azuma & Bishop, 1995).

Once the end-to-end delay of the VR system has been characterized, it is possible to evaluate whether the setup is performing satisfactorily. One rule of thumb is to use the perceptual detectability of end-to-end delay as a threshold. This value is not universal but should be estimated on the VR system at hand (i.e., see Adelstein, Lee, & Ellis, 2003; Di Luca et al., in press; Ellis, Mania, Adelstein, & Hill, 2004; Mania, Adelstein, Ellis, & Hill, 2004). If the performance is insufficient, one should identify which of the VR processing steps the end-to-end delay is mostly attributable to (tracking, processing, or displaying). As was done for one of the setups tested, it is possible to modify each of the three components of the VR system and measure the end-to-end delay of the rest of the system using the proposed method as was done for the results in Figure 7. If the frequency-dependent characteristic is gone or if the delay is considerably reduced, then the modified component is responsible for it. Another way of assessing the origin of end-to-end delay is if two of the components making up the VR system could be substituted with corresponding components with negligible delay. In this way, it would be possible to directly measure the end-to-end delay and characteristics of the remaining component. For example, the position of the camera used to generate the image can be linked to the position of a tracking device with low latency, such as a high-performance mouse or joystick (e.g., Teather et al., 2009). Thus, by placing the light-sensing device on the low latency tracking device so that it captures the light of the real gradient, it is possible to obtain the measure of end-to-end delay without the original tracking component. Also, it is possible to substitute the displaying component with one having a negligible delay (i.e., a high refresh frequency CRT monitor with the light-sensing device attached to the top-left corner). On the other hand, substituting the processing component with one with a negligible delay is not straightforward, so the easiest solution would be to substitute both tracking and displaying and measure the delay of the processing stage alone. Alternately, it is possible to test whether the processing delay decreases significantly by changing the displayed scene (i.e., by not displaying it and showing only the virtual gradient, or by generating the view only for one eye).

But once the transfer function of a VR system has been identified, how can the effect of end-to-end delay be mitigated? Predictive filtering has been used fre-

quently as a way to reduce end-to-end delay (Azuma & Bishop, 1995; see also Adelstein et al., 1996). Depending on the particular implementation and on the amount of end-to-end delay, the compensation obtained with these methods could provide a benefit for the whole or only for a limited range of the frequency spectrum (see, i.e., Azuma & Bishop, 1995). Such methods, however, could cause perceptible artifacts (Adelstein et al., 1996). To balance cost and benefits it is essential to have a good estimate of the end-to-end delay and of the performance of the prediction algorithm.

Acknowledgments

I am grateful to all people who helped me with the article, but in particular: Bruce Kay (University of Connecticut) who had the idea to use gradients rather than discrete events; Patrick Foo (University of North Carolina) and William H. Warren (Brown University) for help in establishing the methodology and in the measurements on the VENLab. From the Max Planck Institute for Biological Cybernetics I am grateful to Betty Mohler, Joachim Tesch, and Trevor Dodds for help with the measurements on the TrackingLab; and to the technicians of the Elektronische Werkstatt for the sensing device and technical support.

This work was supported by EU grant IST-2006-27141 "ImmerSence" and EP 7–ICT–2009–248587 "THE."

References

Adelstein, B. D., Johnston, E. R., & Ellis, S. R. (1996). Dynamic response of electromagnetic spatial displacement trackers. *Presence: Teleoperators and Virtual Environments, 5*(3), 302–318.

Adelstein, B. D., Lee, T. G., & Ellis, S. R. (2003). Head tracking latency in virtual environments: Psychophysics and a model. *Proceedings of the Human Factors and Ergonomics Society*, 2083–2087.

Akatsuka, Y., & Bekey, G. (1998). Compensation for end to end delays in a VR system. *Proceedings of IEEE 1998 Virtual Reality Annual International Symposium,* 156–159.

Azuma, R. T., & Bishop, G. (1995). A frequency-domain analysis of head-motion prediction. *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques,* 401–408.

Di Luca, M. (2010). MATLAB code to measure end-to-end delay of VR. Available at http://www.kyb.tuebingen.mpg.de/bu/people/max/

Di Luca, M., Knörlein, B., Ernst, M. O., & Harders, M. (in press). Effects of multimodal latencies and loading-unloading on compliance perception. *Brain Research Bulletin,* 1–15.

Ehrlich, J. A. (1997). Simulator sickness and HMD configurations. *Proceedings of SPIE, 3206*, 170–178.

Ellis, S. R., Mania, K., Adelstein, B. D., & Hill, M. (2004). Generalizeability of latency detection in a variety of virtual environments. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 48*, 2632–2636.

Ellis, S. R., Adelstein, B. D., Baumeler, S., Jense, G. J., & Jacoby, R. H. (1999). Sensor spatial distortion, visual latency, and update rate effects on 3D tracking in virtual environments. *Proceedings of the IEEE Virtual Reality,* 218–221.

Ernst, M. O. & Banks, M. S. (2002). Humans integrate visual and haptic information in a statistically optimal fashion. *Nature, 415*, 429–433.

He, D., Liu, F., Pape, D., Dawe, G., & Sandin, D. (2000). Video-based measurement of system latency. *International Immersive Projection Technology Workshop.*

Held, R., & Durlach, N. (1991). Telepresence, time delay and adaptation. In S. R. Ellis (Ed.), *Pictorial communication in virtual and real environments* (pp. 109–112). Bristol, PA: Taylor & Francis.

Kennedy, R. S., Lane, N. E., Berbaum, K. S., & Lilienthal, M. G. (1989). Simulator sickness in US Navy flight simulators. *Aviation, Space and Environmental Medicine, 60*, 10–16.

Kijima, R., Kitabayashi, K., & Hayakawa, Y. (2007). Time delay measurement and lag compensation with reflex HMD. *Proceedings of the IEEE Virtual Reality Conference.*

Knörlein, B., Di Luca, M., & Harders, M. (2009). Influence of visual and haptic delays on stiffness perception in augmented reality. *Proceedings of the 8th IEEE International Symposium on Mixed and Augmented Reality,* 49–52.

Liang, J., Shaw, C., & Green, M. (1991). On temporal-spatial realism in the virtual reality environment. *Proceedings of the 4th Annual ACM Symposium on User Interface Software and Technology,* 19–25.

Mania, K., Adelstein, B., Ellis, S., & Hill, M. (2004). Perceptual sensitivity to head tracking latency in virtual environments with varying degrees of scene complexity. *Proceedings*

of the 1st Symposium on Applied Perception in Graphics and Visualization, 39–47.

Miller, D., & Bishop, G. (2002). Latency meter: A device to measure end-to-end latency of VE systems. *Proceedings of SPIE—Stereoscopic Displays and Virtual Reality Systems.*

Mine, M. (1993). Characterization of end-to-end delays in head-mounted display systems. *Technical Report of the University of North Carolina* (TR93–001).

Olano, M., Cohen, J., Mine, M., & Bishop, G. (1995). Combatting rendering latency. *Proceedings of the 1995 Symposium on Interactive 3D Graphics.*

Steed, A. (2008). A simple method for estimating the latency of interactive, real-time graphics simulations. *Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology,* 123–129.

Swindells, C., Dill, J., & Booth, K. (2000). System lag tests for augmented and virtual environments. *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology,* 161–170.

Teather, R., Pavlovych, A., Stuerzlinger, W., & MacKenzie, I. S. (2009). Effects of tracking technology, latency, and spatial jitter on object movement. *Proceedings of the IEEE Symposium on 3D User Interfaces,* 229–230.

Ware, C., & Balakrishnan, R. (1994). Reaching for objects in VR displays: Lag and frame rate. *ACM Transactions on Computer-Human Interaction, 1*, 331–356.

Watson, B., Walker, N., Ribarsky, W., & Spaulding, V. (1998). Effects of variation in system responsiveness on user performance in virtual environments. *Human Factors, 40*, 403–404.