30

Tracking for Training in Virtual Environments

Estimating the Pose of People and Devices for Simulation and Assessment

Greg Welch and Larry Davis

1 Introduction

Estimating or *tracking* human and device motion over time is a central requirement for most Virtual Environment (VE) based training systems. In some cases it is sufficient to know a trainee's head or torso location (2D or 3D position only) within the training environment. Other cases require the full body *pose*—the position *and* orientation. Still other cases require complete body *posture*—the positions, orientations, and/or configurations of the trainee's arms, hands, legs, feet, as well as hand held devices (e.g., surgical instruments or weapons). Some times this information must be known with precision and accuracy to better than a millimeter, some times less spatial and temporal resolution is needed.

Primary uses for motion tracking for *training* include real-time and on-line simulation associated with a "live" training activity, and on- or off-line assessment of performance or behavior. "Live" training is most often associated with VE tracking where, for example, a military trainee performing a room-clearing exercise might wear a head-mounted display while moving around a virtual room looking for virtual enemies. In this case, at a minimum the trainees' heads would need to be tracked for the purpose of rendering the proper HMD imagery as they are moving around. Most likely their weapon would also need to be tracked to render it in the HMD imagery, and additionally perhaps their hands or other limbs would be tracked, so that they too could be properly rendered in the HMD imagery. Given the room-clearing scenario, one might want to know how efficiently the trainees moved during the exercise, where they were looking, where their weapons were pointing, etc.

In this chapter we look at tracking scenarios, technologies, and issues related to training in VE training systems. We explore the fundamental aspects of tracking only to the degree it is useful for considering, choosing, and using tracking systems for training. For further information about the fundamental technologies and methods used in tracking systems, we encourage the reader to refer to the many excellent existing survey articles (Ferrin, 1991; Meyer et al., 1992; Bhatnagar, 1993; Durlach and Mavor, 1994; Allen et al., 2001; Welch and Foxlin, 2002). In addition, Foxlin's chapter in Kay Stanney's *Handbook of Virtual Environments* (2002) is an excellent source of information about the requirements for tracking and the underlying fundamental technologies (Stanney, 2002, pp. 163–210). Beyond discussing the fundamental technologies, Allen, Bishop and Welch (2001) discuss the most common source/sensor configurations, and both Allen et al. (2001) and Foxlin's chapter in (Stanney, 2002) discuss the most common approaches and algorithms for estimating pose from the source/sensor measurements.

The remainder of this chapter is organized as follows. In Section 2 we describe the tracking considerations relevant to the most common scenarios related to training. While we do not intend to provide a complete tracking survey in this chapter, in Section 3 we describe some of the tracking technologies available to today for training. In Section 4 we discuss what we feel are the most important fundamental issues that one should consider when purchasing, installing, and using commercial tracking systems. Finally in Section 5 we speculate just a little about where research and development are heading.

Note that some of the material from this chapter was reproduced or adapted (with ACM copyright permission) from the course notes for ACM SIGGRAPH 2001 Course 11, "Tracking: Beyond 15 Minutes of Thought," by Allen, Bishop, and Welch (2001). While the course notes were written by all three individuals, the relevant material was originally written by Welch.

2 Tracking Scenarios

When choosing or using a motion tracking system for training purposes one needs to consider many factors. Here we explore three particular issues: what, where, and when to track. Considering these issues in advance of choosing a tracking system should help narrow the choices, for example given the technologies covered in

Section 3. It should also help to calibrate expectations for what might be possible in terms of precision, accuracy, robustness, and overall suitability, while also allowing the developers to focus on the relevant issues described in Section 4.

2.1 What to Track

The primary consideration is *what* needs to be tracked, and for what purpose. While this might at first seem obvious, one tracking solution will rarely fit all circumstances. Review of the tracking scenario can reveal unrealistic expectations (practical limitations) or liberating opportunities to determine degrees of freedom, accuracy and precision, to accommodate the practical limitations of the available tracking technology.

In his excellent *Taxonomy of Usability Characteristics in Virtual Environments*, Gabbard distinguishes between "VE User Interface Input Mechanisms" and "Tracking User Location and Orientation" (Gabbard and Hix, 1997, pp. 24–25). Although user interface mechanisms could potentially be included in a training application, here we will concentrate primarily on trainee pose and posture.

In VE, one needs to consider weight and bulk of tracking components placed on a user, however, for training these potential distractions and biases are even more of a concern, as ideally the desired response is evoked in exactly the same situation as the real event. If the training environment relies on user-worn components that the trainees have to knowingly contend with, it might affect their performance and hence their training. The impact depends on how noticeable the component is, and/or how it forces him or her to adjust their behavior. For example, for a Marine doing live-fire training exercises, a full lycra body suit instrumented with retroreflective spheres or inertial sensors could provide a wealth of information about the Marine's dynamic posture, but the Marines is likely to be very conscious of the suit, even if somehow integrated into his or her normal camouflage clothing. GPS and other devices can add weight, which corresponds to unusual forces on the body, which might be noticed. How noticeable a body-worn component is depends on the *relative* forces—something that is very noticeable on the hand might be less so on the back/torso.

Knowing what parts of the body need to be tracked will help determine the necessary degrees of freedom, accuracy, resolution, etc. For example, head tracking for head-mounted display based VR can be very demanding in particular in terms of delay-induced error (Section 4.2). Because our heads are relatively heavy, and attached to the mass of the torsos, people cannot translate their heads very fast. However as pointed out by Ron Azuma, "At a moderate head or object rotation rate of 50 degrees per second, 100 milliseconds (ms) of latency causes 5 degrees of angular error. At a rapid rate of 300 degrees per second, keeping angular errors below 0.5 degrees requires a combined latency of under 2 ms!" (Azuma, 1993, p. 50). A more typical 50 milliseconds of delay corresponds to about 15 degree of error for such rotation. However if one intends to use a room-mounted display system (projected imagery or flat panels), concerns about latency-induced rotational error are typically reduced dramatically. This is because such head rotation typically causes relatively small eye translation with respect to the fixed displays.

People can rotate their wrists at angular rates that are roughly comparable to head rotations, but they can translate their hands much fast than their heads, by rotating rapidly about the wrist or elbow. (Hands have much less mass than heads!) Typical arm and wrist motion can occur in as little as 1/2 second, with typical "fast" wrist tangential motion occurring at three meters per second (Atkeson and Hollerbach, 1985). Such motion corresponds to approximately one to ten centimeters of translation throughout the sequence of 100 measurements used for a single estimate. For systems that attempt sub-millimeter accuracies, even slow motion occurring during a sequence of sequential measurements impacts the accuracy of the estimates. For example, in a multiple-measurement system with 30 millisecond total measurement time, motion of only three centimeters per second corresponds to approximately one millimeter of target translation throughout the sequence of sensor measurements acquired for one estimate.

Finally, it is critical to consider *how many* individuals one needs to track simultaneously, for real-time graphics or training/behavioral analysis. The primary concern is one of the *sociability* of the tracking system as defined in Meyer et al. (1992). That is, how well does the approach/system support multiple simultaneous users? For example, an optical system could be more prone to occlusions from other trainees than magnetic or inertial systems. Even if the medium itself is relatively unaffected by multiple nearby users, as is the case for example with inertial sensors, one has to look at how one gets the data off the devices and processed simultaneously in real time. It is often hard enough to do this with one user, much less two or more. Sociability is something to investigate if you intend to track multiple trainees.

2.2 Where to Track

Beyond what one is tracking (Section 2.1) one needs to consider *where* the tracking needs to be done. Some applications might require only small-scale tracking where the user does not walk around. For example, if one is looking at training a task involving manual dexterity of the hand or fingers, such as surgical suturing, one does not need wide-area tracking but instead might get by with a couple of "glove" devices. If the pose of the hand (back or palm) is of interest over a small area, one might be able to use conventional single-sensor magnetic systems such as those made by Ascension or Polhemus, or inertial hybrids such as those made by InterSense.

For six degree of freedom head tracking, the training task might only involve what some call "fish tank" VR, whereby the user stands or sits in front of a cathode ray tube monitor or flat panel display, viewing some imagery that needs to have head-motion parallax, etc. In such cases concerns about optical or acoustic occlusions are likely to be lessened, allowing consideration of vision (camera) based tracking, etc. If the allowable volume of motion is restricted one might be able to use mechanical tracking such as the Shooting Star Technology ADL-1.

If one needs 6 degree of freedom head tracking over a room or lab-sized space one needs to be looking at wide-area systems such as the HiBall^{TM} by 3rdTech, the IS-900 by InterSense, or a cellular magnetic system such as the Ascension Flock of Birds®. With an increase in working volume comes a likely increase in number of trainees. If multiple trainees need to be supported in a large space, one needs to consider the sociability of the candidate systems as mentioned in Section 2.1 and in (Meyer et al., 1992).

Perhaps the most difficult tracking challenges are related to unusually large spaces, in particular outdoors. Outdoor environments can present exceptional challenges in terms of the sheer scale of the working volume (and corresponding difficulties with sensor signals); difficulties in dealing with uncontrollable environmental factors such as the too little or too much light, sound, etc.; and even something as seemingly mundane as getting signals off the body-worn sensors of the individual trainees. The problem becomes one not of just signal strength and data bandwidth if the trainees will be distributed over a large outdoor area, but potentially one of *timing*—latency, synchronization, etc. Over very large areas, e.g., a live-fire desert training environment, certain technologies become impractical or impossible. For example, an active magnetic system could not be made to work over several kilometers. In such cases *self-tracking* approaches such as those that make use of inertial devices become more attractive, as their accuracy and sensitivity does not necessarily depend on external infrastructure. On the other hand, there is no inertial tracking system that can function with reasonable bounded error in an unaided fashion over a large area. As such the approach would need to combine the inertial sensing with other approaches such as GPS (which itself is limited to meters of accuracy) or even vision/camera-based approaches.

2.3 When to Track

Finally, beyond what (or why) and where to track, one needs to consider *when* to track. By "when" we really mean when will the sensor data (whatever it is) be processed to generate pose or posture estimates. For example, if one needs to create computer generated imagery for the trainee(s), one will need some form of *on-line* (active during the training) and *real-time* (fast enough to keep up) estimation. In such cases, temporal issues such as latency (see Section 4.2) might be of primary concern. If one is only interested in post-exercise pose or posture analysis, then as long as the sensor data can be collected in real time, and accurately time-stamped if synchronization is needed, then the sensor data can be processed off line after the exercise to estimate the pose/posture that is then analyzed.

While the latter case (post-training analysis) might sound easier, in fact waiting to process sensor data until later in time (off line) can mean that one needs to transmit and/or store tremendous amounts of data during the exercise. The pose/posture estimation from the raw sensor data, in effect provides a form of compression of the sensor data, as many readings are (typically) combined into single estimates at a reduced rate.

On the other hand, analyzing sensor data in an off-line fashion (after the exercise) means that one can effectively “look into the future” when filtering the data. The benefits of such non-causal filtering can be tremendous if, for example, the data has structured errors in it, gaps, etc. Systems that operate in an on-line fashion, cannot (by definition) look ahead in time and therefore can only generate pose/posture estimates based on past measurements. This can make such systems susceptible to data dropout as well as unexpected changes in target (trainee) dynamics, such as transitions from still to rapid motion.

Note that it should be possible to take a hybrid approach, where some processing is done on line, in real time (perhaps to both compress the sensor data and provide some on-line feedback or analysis to the trainers) and further pose/posture refinement and behavioral analysis (for example) is done later, off line.

3 Today’s Technologies

Once one has determined what is to be tracked and the reasons for tracking it, the remaining decision is the type of tracking technology to use. There are many factors to influence this decision including the operating principle of the tracker, the required performance, and the cost.

In this section, we provide guidance in selecting commercial, off-the-shelf (COTS) tracking systems. We first discuss the types of trackers, categorized by principle of operation, and give examples of COTS systems within each category.

We then provide a list of tracker characteristics to consider when deciding upon a particular tracking system. Finally, we discuss choices that must be made when considering how one plans to interface with the desired tracking system.

The tracking system taxonomy found in (Welch and Foxlin, 2002) is used to classify the systems according to different operating principles. In addition, the specific tracking systems highlighted are limited to systems available for retail purchase at the time of writing. Some of the more popular trackers that have been discontinued by their manufacturers, such as the Boom3C from Fakespace or the ADL-1 from Shooting Star Technology, are often available through online auction sites. For more detailed discussion regarding operating principles and tracking in general, the reader may refer to (Welch and Foxlin, 2002) and (Foxlin, 2002).

3.1 Mechanical

Mechanical tracking systems use physical links to determine the pose (position and orientation with respect to a frame of reference) of a tracked object. Mechanical tracking systems have the advantages of high update rates (number of pose reports per second) and high accuracy. However, the object being tracked is tethered to the tracker, limiting the range of motion.

Mechanical tracking technology is used for digitizers as well as tracking hand motion and motion capture/pose determination. The Microscribe G2LX from Immersion and the FaroArm from Faro Technologies are examples of digitizers, which are tracking systems used to create digital representations of real objects. The Cyberglove, also from Immersion, is a popular device for measuring hand movements. It uses changes in electrical resistance to indicate the amount of bending of the fingers. Another popular option is the X-IST DataGlove from No DNA that includes conductive bend sensors and piezoelectric pressure sensors. Two mechanical tracking systems used for motion capture are the Gypsy 6 from Animazoo (a mechanical system that uses inertial sensors) and the ShapeTape and ShapeWrapIII systems from Measurand that use fiber optic bend sensing.

3.2 Inertial

Intertial trackers use the Earth's gravitational field to determine the pose of tracked objects. Inertial trackers can be very small, have low latency, and consume small amounts of power. Their drawback is that they suffer from drift (a gradual loss of measurement accuracy). This tendency is often mitigated by using inertial systems as part of hybrid tracking systems.

A popular inertial tracker is the Inertia Cube series from InterSense. The Inertia Cube 3 combines prediction algorithms with accelerometers and gyros to provide 360 degrees of rotational measurement. Other examples of inertial tracking systems include the 3D Bird from Ascension (180 degrees in elevation, 360 degrees in azimuth and attitude) and the MTx from Xsens (360 degrees in all directions).

3.3 Acoustic

Acoustic trackers use ultrasonic sound (near 40 kHz) to determine the pose of objects. The time of flight differences from multiple sources are measured and position is determined based upon the characteristics of sound traveling in air. However, multi-path reflection of the emitted sound can severely degrade tracking performance as well as occlusions between the emitter and receiver. Acoustic tends to suffer in outdoor conditions as well as near walls where air currents and noise can cause interference. Acoustic tracking is also used as part of hybrid tracking systems. Resolutions tend to be several millimeters and accuracies can be difficult to maintain if conditions can't be very controlled.

An example of acoustic trackers is the Hexamite HX11. The HX11 tracks the location of pulsed, ultrasonic emitters with chains of ultrasonic receivers. In theory, the system has no limit to the tracking coverage area.

3.4 Magnetic

Magnetic tracking systems use electromagnetic field differences to determine position and orientation. They offer good update rates and rugged performance. Magnetic systems are very susceptible to external magnetic fields, as well as induced fields from nearby metal objects or electrical equipment.

Two classic VE tracking systems are magnetic trackers. They are the Polhemus Fastrak and the Ascension Flock of Birds. The Fastrak uses AC magnetic fields to determine position and orientation, while the Flock of Birds uses a pulsed, DC magnetic field. Recent demonstrations of the Polhemus Patriot wireless tracking system have shown considerable improvement regarding robustness to metallic interference. The Patriot is expandable and can track up to four objects.

3.5 Optical

Optical tracking systems use the properties of light to determine the pose of tracked objects. Markers that emit light (active trackers) or reflect light (passive trackers) are used to determine pose. The arrangement of the light sources and sensors provide a sub-classification for optical trackers. If the sensors are mounted on the object to be tracked, the approach is sometimes called "inside-out." If the sensors are fixed, and the markers are attached to the object to be tracked, the approach is sometimes called "outside-in." Optical trackers provide the highest accuracy of any tracking type typically being sub millimeter, and they provide data in the 120 to over 1000 frames per second reducing motion blur artifacts. They suffer from line-of-sight issues, meaning the sensors or emitters can be blocked. Optical trackers that use infrared light often have difficulties in bright lights or direct sunlight. An important feature of an optical tracker is its field of view, meaning the angle through which its sensors can detect targets.

The HiBall tracking system from 3rd Tech is an inside-out tracker capable of unlimited tracking coverage in theory. Another inside-out tracker is the LaserBird from Ascension that has an infrared light source that sweeps through the tracked area.

Outside-in optical trackers include the DynaSight from Origin Instruments and the Certus from Northern Digital. Both of these systems use active infrared emitters to determine the pose of tracked objects. The DMAS from Motion Imaging is another outside-in system but it is a markerless tracking system, tracking objects through analysis of video frames.

Many optical tracking sytems are used for motion capture in addition to pose tracking. Many are passive trackers that detect reflections from special markers in the tracked area. Examples of these systems include the OptiTrack from Natural Point, the PPT from WorldViz, the Vicon MX, the Impulse from PhaseSpace, the Eagle from Motion Analysis, and the ProReflex from Qualisys

3.6 Radio Frequency and Ultra-Wide Band

Radio frequency trackers use differences in radio signals to determine position in an environment in meter or sub-meter resolutions. These differences include signal strength, signal content, and time-of-flight. Ultra-Wide Band (UWB) communications use similar principles but operate in a higher frequency range (2 GHz–7.5 GHz)

The Ubisense system uses a pulsed UWB signal to determine the 3D location of tags within the tracked area. Up to 1000 tags can be located simultaneously by the system and simultaneous RF communication occurs to dynamically change update rates. In this area, there have also been recent announcements from Thales on an indoor/outdoor RF tracking system and from AeroScout regarding a Wi-Fi based Active RFID tracking system.

3.7 Hybrid

As the name suggests, hybrid trackers combine multiple types of operating principles to provide increased robustness of pose measurements. Two examples of hybrid systems from InterSense are the IS-900 and the IS-1200. The IS-900 combines acoustic and inertial measurements and the IS-1200 combines optical and inertial measurements. Another hybrid system is the HyBird from Ascension. It combines inertial and optical tracking methods for use in cockpit applications.

3.8 Factors to Consider when Evaluating Tracking Systems

After considering the factors intrinsic to the technology used by the tracker, a tracking system should be evaluated based upon its performance characteristics. These characteristics include the following:

1. Degrees of Freedom: 3 (orientation or position tracking only) or 6 (position and orientation tracking)

2. Accuracy: The absolute difference between the real position of the tracked object and the position reported by the tracker.

3. Resolution: The minimum change in the position of the tracked object that can be detected by the tracker.

4. Jitter: The instantaneous change in position from frame to frame, as reported by the tracker when the tracked object is stationary.
5. Drift: The gradual change in position (or bias) reported by the tracker over time.
6. Latency: The amount of delay between when the tracked object moves and when the data corresponding to the movement is transmitted.
7. Update Rate: The number of measurements that the tracker makes each second. This number may decrease as additional objects are tracked.
8. Range: In principle, the maximum distance that a single object can be tracked. To increase the range, additional sensors may be required
9. Maximum Tracked Objects: The maximum number of objects that can be tracked simultaneously.
10. Operating Principle: Does the application environment enable the choice of tracking technology, e.g., using a magnetic tracker near a generator)?
11. Untethered Operation: Can the tracked objects move freely or are they constrained by a wire, connector, etc.?
12. Price

In Figure 1, a table comparing many of these factors for the preceding tracking systems is presented. While this table is a starting point for investigations or discussions in choosing a tracking system, the reader is cautioned to consider the issues presented in Section 4.

PLACE FIGURE 1 ABOUT HERE

3.9 Interfacing with the Tracker

A final aspect to consider is the challenge of interfacing hardware and software for communication with the tracking system. In the case of hardware interfaces, many tracking systems ship with RS-232 connectors, also known as serial or DB-9 connectors. If the tracker of choice has a serial port connection, the hardware interface will be easiest with a computer with a serial port on the motherboard. If a computer with a serial port is unavailable (as is the case with most laptops), a serial port expansion card can be added or a USB-to-Serial converter can be used. USB-to-Serial converters are inexpensive and do not require the addition of internal electronics, but they may pose problems communicating with the tracker.

The environment where the tracker will be used must also be considered when choosing the software interface. A sample application is included with the tracking system to allow measurement out-of-the-box, but integration with existing applications requires more consideration. Typically, tracking systems ship with a C/C++ Application Programming Interface (API) for customized software integration. In addition, many tracking products also have software interfaces available in virtual world building applications and in pre-existing device interface libraries. Vizard, from WorldViz, Inc. (`http://www.worldviz.com/`), is a commercially available world-building application that is easy to learn. VR Juggler (`http://www.vrjuggler.org/`) is an open source software platform for virtual environment application development. At a more basic level, the Virtual Reality Peripheral Network (`http://www.cs.unc.edu/Research/vrpn/`) is a public domain set of libraries for distributed tracking/interface device connections that can be easily incorporated in custom applications. All three software packages contain ready-to-use interfaces for many of the tracking systems mentioned in this section and the capability to customize interfaces to accommodate other types of trackers.

4 Fundamental Usage Issues

For VE displays one's overall goal is perfect, continuous registration and/or rigidity. For motion capture one's overall goal is accuracy that is sufficient enough to do training-related behavioral analysis. In either case, tracking is hardly perfect. It cannot be. But it can often be made "good enough" if one chooses carefully to try and address the fundamental sources of error as much as possible.

There are several sources of error in estimates from tracking and motion capture systems. Whether looking at head tracking or hand tracking, the basic principles are the same. There are of course many causes of visual error in interactive computer graphics systems. There are many people who would argue that various errors originating in the tracking system dominate all other sources. In his 1995

Ph.D. dissertation thoroughly analyzing the sources of error in an Augmented Reality system for computer-aided surgery, Holloway stated

> “Clearly, the head tracker is the major cause of registration error in AR [augmented reality] systems. The errors come as a result of errors in aligning the tracker origin with respect to the World CS (which may be avoidable), measurement errors in both calibrated and multibranched trackers, and delay in propagating the information reported by the tracker through the system in a timely fashion.” (Holloway, 1995, p. 135)

Holloways dissertation offers a very thorough look at the sources of error in the entire VR pipeline, including the stages associated with tracking. It is a valuable resource for those interested in a rigorous mathematical analysis. Chapter 8 of the dissertation discusses some methods for combating the problems introduced by tracker error, in particular delay.

For a person designing, calibrating, or using a tracking or motion capture system, it is useful to have some insight into where errors come from. As Michael Deering notes in his 1992 SIGGRAPH paper, “...the visual effect of many of the errors is frustratingly similar” (Deering, 1992). This is especially true for tracking errors. We have seen people build VR applications with obvious head tracker transformation errors, and yet people had great difficulty figuring out what part of the long sequence of transforms was wrong, if it was a static calibration error, or a simple sign error.

Yet even when all of the transforms are of the correct form, the units of translation and orientation match, and all the signs are correct, there are still unavoidable errors in motion tracking, errors that confound even the most experienced of practitioners of interactive computer graphics. No matter what the approach, the process of pose estimation can be thought of as a sequence of events and operations. The sequence begins with the user motion, and typically ends with a pose estimate arriving at the host computer, ready to be consumed by the application. Clearly by the time a pose estimate arrives at the host computer it is already "late"—and you still have to render an image and wait for it to be displayed! Section 4.4 offers some hope for addressing the long delays and in some sense "catching up" with the user motion, but that doesnt mean that we dont want to minimize the delay, and to understand how all of the various errors affect the outcome.

The sources of error in tracking and motion capture can generally be divided into two primary classes: *spatial* and *temporal* errors. We refer to issues and errors that arise when estimating the pose of an immobile target as *spatial issues.* (Note that spatial issues include measurement noise, which is generally a function of time, but statistically stationary.) We refer to issues and errors that arise when tracking a moving object as *temporal issues.* These issues include errors that arise from the inevitable sources of delay in the tracking pipeline (delay-induced error).

4.1 Spatial Issues

For an immobile sensor (static motion), we can further divide the measurement errors into two types: *repeatable* and *non-repeatable*. Some trackers (for example, magnetic ones) have systematic, repeatable distortions of their measurement volume, which cause them to give erroneous data; we will call this effect static field distortion. The fact that these measurement errors are repeatable means that they can be measured and corrected as long as they remain unchanged between this calibration procedure and run time. See (Livingston and State, 1997) for an example of how this can be done.

One also needs to consider the *non-repeatable* errors made by the tracker for an immobile sensor. Some amount of noise in the sensor inputs is inevitable with any measurement system, and this measurement noise typically leads to random noise or jitter in the pose estimates. By our definition, this type of error is not repeatable and therefore not correctable *a priori* via calibration. Moreover, jitter in the trackers outputs limits the degree to which the tracker can be calibrated. The amount of jitter is often proportional to the distance between the sensor(s) and the source(s), and may become relatively large near the edge of the trackers working volume.

While these effects are true for many source/sensor combinations, let us consider the effects related to image-forming digital cameras. There are two reasons we choose to look at cameras. For one, camera geometry should be relatively easy for most readers to understand. In addition, cameras are increasingly being used in tracking and motion capture systems, probably partly because of decreasing costs, increasing resolutions, and increasing image processing capabilities in computers in general (device and CPU bandwidth, and computation power).

Cameras effectively measure the number of photons arriving at each photo cell, over the period that the shutter is open. Those photons might have originated from an active tracking source such as an LED, or they might originate from an ambient light source and be reflected by a passive tracking target or marker. In either case, there are two related issues that are useful to consider: the size, or cross section, of the target in the camera (the resolution of the target) and the amount of light reaching the camera (the brightness and/or contrast).

Most readers will be familiar with the notion that as a target being imaged by a camera gets farther away, its image gets smaller in the camera. Specifically, as shown in Figure 2, as the distance d to a target increases, the angle θ that the object spans in the camera's field of view decreases proportionally. The effect is twofold. First, the smaller the angle θ, the fewer camera pixels cover the target. Correspondingly, as the distance at which one is attempting to image a target increases, so increase the projection or size of each pixel at that distance. In other words, as distance increases, one's ability to resolve fixed size objects in the world decreases.

PLACE FIGURE 2 ABOUT HERE

While the purely geometric relationship between distance and size (or resolution) is important, in the case of cameras (and similarly for magnetic devices) one also needs to consider the decrease in light that reaches the camera with increasing distance. This affects the brightness of any light emanating from or reflected by the target, as measured by the camera. As with a camera, as the distance d to a target increases, the angle θ that the object spans in the camera's field of view decreases proportionally. However as indicated in Figure 3, the brightness decreases at a rate proportional to the *square* of the distance.

PLACE FIGURE 3 ABOUT HERE

This quadratic reduction is because the photons propagate away from the light source (or reflective patch) in a particular direction, covering some solid angle. For the sake of illustration, consider an omnidirectional light source, where the light propagates equally in all directions in a spherical manner. In this case, because the area of the surface of the wave front increases proportionally to the square of the distance, the density of photons on the surface of the wave front (density per surface area) decrease proportionally to the square of the distance. For a fixed size reflective patch on a tracking target for example, this means that the number of photons hitting the target decreases proportionally to the square of the distance.

Beyond distance to the target, it is usually the case that the angle between the normal direction (e.g., the surface normal) of the target and the camera and/or light source plays a role in the number of photons reaching the surface or camera. Figure 4 provides a simply illustration of this. In some cases, for example the brightness is proportional to $\cos\alpha$. In that case, if the surface is see “straight on” then there is no attenuation of the light because $\cos(0) = 1$. At the other extreme, if the camera is seeing the surface from an extreme angle, e.g., 90 degrees, then there would be extreme attenuation of the light because $\cos(90 \text{ degrees}) = 0$.

PLACE FIGURE 4 ABOUT HERE

Finally, one needs to consider that in a passive system, e.g., vision-based tracking systems with passive markers, the photons have to travel from the light source to the target as in Figure 3, and then from the target back to the camera (each photo cell) as in Figure 2. The effect is that the density of photons can, in some cases, decrease proportionally to the *square of the square* of the distance. In other words, the brightness is proportional to $1/d^4$.

For image-based systems, any reduction in brightness corresponds to a reduction in contrast (the ratio of brightest to darkest signal in the image), which corresponds to a reduction in the *effective* resolution at that distance. This effect can be illustrated or measured by the *modulation transfer function.* The shape of the modulation transfer function indicates the magnitudes of various spatial frequencies measured, compared to the spatial frequencies inherent in the scene. In general, the less light there is, the more difficult it is to resolve something. Thus the amount of light impacts the resolution of the system.

In some cases one can precisely control the light, in others one cannot. For example, some optical motion capture systems attempt to address modulation transfer functions issues by using infrared lights that are located very near the cameras—often times in a ring around each camera. If the timing of the lighting can be controlled precisely, one can "pump" a lot of photons into the scene just when the camera shutter is open, thus increasing the signal without unduly flooding the scene with infrared light.

With optical and other systems, one might also use differential signaling to improve the contrast and hence the signal-noise ratio. The idea is to take one image with the lights on (bright) and one with the lights off (dim). When one subtracts the two, the reflective objects should dominate the result, while other bright sources should be eliminated (subtracted out). There are practical limits on how much one can achieve with this approach, and there are temporal concerns as one must capture *two* sequential images to do the differencing. (The target might well be moving during the capture time!)

4.2 Temporal Issues

Beyond the spatial concerns covered in the previous section, there are several temporal concerns related to tracking. The problems include the rate at which discrete measurements are made of a moving target (any medium), the duration of each low-level sample of a device (e.g., how long a camera shutter is open), and the delay or latency from the time the measurement is made to the time the effect is "seen" by the remainder of the system (graphics subsystem, display, etc.).

4.2.1 Delay-Induced Error

Any measurement of a non-repeating, time-varying phenomenon is valid (at best) at the instant the sample occurs—or over the brief interval it occurs, and then becomes “stale” with the passage of time until the next measurement. The age of the data is thus one factor in its accuracy. Any delay between the time the measurement is made and the time that measurement is manifested by the system in a pose estimate contributes to the age and therefore the inaccuracy of that measurement. The older the tracker data is, the more likely that the displayed image will be misaligned with the real world.

We feel that concerns related to *dynamic error* (including *dynamic tracker error* and *delay-induced error* from above) deserve distinct discussion. This class of error is often less obvious when it occurs, and when one does recognize it, it is difficult to know where to look to minimize the effects. Further, it is literally impossible to reduce the delay to zero. One typically has to contend with overall system delays on the order of 10–100 milliseconds. See (Meehan et al., 2003) for one example of the effects such delays can have.

4.2.2 First-Order Dynamic Error

Probably the most obvious effect here is the overall *dynamic error* caused by continued user motion after a tracker cycle (sample, estimate, produce) has started. If the users head is rotating with an angular velocity of $\dot{\theta}$ and translating with a linear velocity of $\dot{x}$ then simple first-order models for the delay-induced orientation and translation error are given by

PLACE IMAGE FILE welch-eqn-1-2-trimmed.jpg

EXACTLY HERE

where Δt is the sum of the total motion delay Δt_m for the tracking system as described below, as well as Δt_g, the delay through the remainder of the graphics pipeline—including rendering and image generation, video synchronization delay, frame synchronization delay, and internal display delay. The *video synchronization delay* is the amount of time spent waiting for a frame buffer to swap—on average 1/2 the frame time. (*Synchronization delay* in general is described more below.) The *internal display delay* is any delay added by the display device beyond the normal frame delay. For example, some LCD and DLP devices buffer images internally in a non-intuitive manner as they convert adjustable display resolution from the input to a fixed pattern of pixels on the screen, sometimes introducing several video frames latency. The delay must be measured on a per-device basis if it is important.

4.2.3 Motion-Induced Measurement Noise

Clearly the placement of sources and sensors can affect the signal quality as described above in Section 4.1. But there are often other *internal* (a.k.a. *intrinsic*) parameters that need to be specified. For example for cameras one needs to specify the focus and aperture settings, gains, frame rates, and shutter/exposure times. In particular here we want to point out the potential for motion-induced noise during a camera exposure, a magnetic current measurement, an acoustic phase measurement, etc. In a nutshell, just as the target motion is an issue for multiple measurements, it is often an issue for even a *single* measurement.

Without loss of generality, let us assume a regular camera update rate of $1/dt$. Each cycle can be divided into sampling (exposure) time τ_s, processing time τ_p, and idle time τ_i. The three times sum to the overall update period, i.e. $dt = \tau_s + \tau_p + \tau_i$. Because cameras integrate light over the non-zero shutter time τ_s, estimating camera motion or dynamic scene structure using feature or color matching always involves a tradeoff between maximizing the *signal* and minimizing any motion-induced *noise*. If the shutter time is too short the dynamic range or contrast in the image will be too low, reducing the *effective* resolution, increasing the measurement uncertainty, and negatively impacting the final motion or structure estimates. Conversely, if the shutter time is too long the measurements will be corrupted by scene or camera motion (blur), again reducing the effective resolution, increasing the measurement uncertainty, and negatively impacting the final estimates. See for example Figure 5, which illustrates the amount of motion in the image planes of various cameras, under a changing scene. This issue is discussed more in (Welch et al., 2007).

PLACE FIGURE 5 ABOUT HERE

4.2.4 Sensor Sample Rate

Per Shannon's sampling theorem (Jacobs, 1993) the measurement or sampling rate r_{ss} should be at least twice the true target motion bandwidth, or an estimator may track an alias of the true motion. Given that common arm and head motion bandwidth specifications range from 2 to 20 Hz (Fischer et al., 1990; Foxlin, 1993; Neilson, 1972), the sampling rate should ideally be greater than 40 Hz. Furthermore, the estimation rate r_e should be as high as possible so that slight (expected and acceptable) estimation error can be discriminated from the unusual error that might be observed during times of significant target dynamics.

4.2.5 Synchronization Delay

While other latencies certainly do exist in the typical VE system (Mine, 1993; Durlach and Mavor, 1994; Wloka, 1995) tracker latency is unique in that it determines how much time elapses before the first possible opportunity to respond to user motion. When the user moves, we want to know as soon as possible. Within the tracking system pipeline of events (and throughout the rendering pipeline) there are both fixed latencies associated with well-defined tasks such as executing functions to compute the pose, and variable latencies associated with the synchronization between well-defined asynchronous tasks. The latter is often called *synchronization delay*, although sometimes also *phase delay* or *rendezvous delay*. See for example Figure 6.

PLACE FIGURE 6 ABOUT HERE

In the example of Figure 6, measurements and pose estimates occur at regular but *different* rates. Inevitably, any measurement will sit for some time before being used in to compute a pose estimate. At best, the measurement will be read immediately *after* it is made. At worst the measurement will be read just *before* it is replaced with a newer measurement. On average the delay would be 1/2 the measurement rate.

Figure 7 presents a more involved example, a sequence of inter-tracker events and the corresponding delays. Consider an instantaneous step-like user motion as depicted in Figure 7. The sequence of events begins at t_m , the instant the user begins to move. In this example the sensors are sampled at a regular rate $r_{ss} = 1/\tau_{ss}$, such as would typically be the case with video or a high-speed A/D conversion. On average there will be $\Delta t_{ss} = \tau_{ss}/2$ seconds of sample synchronization delay before any sample is used for pose estimation. Because the pose estimate computations are repeated asynchronously at the regular rate of $r_e = 1/\tau_e$ there will be an average of $\Delta t_e = \tau_e/2$ seconds of estimation synchronization delay, after which time the estimation will take τ_e seconds. Assuming a client-server architecture such as (Taylor, 2006) the final estimate will be written to a server communications buffer where it is being read at a rate of $r_{srb} = 1/\tau_{srb}$, and will therefore wait an average of $\Delta t_{srb} = \tau_{srb}/2$ seconds before being read and transmitted over the network to the client. The network transmission itself will take τ_{net}, and the final client read-buffer synchronization delay will take $\Delta t_{crb} = \tau_{crb}/2$ seconds, where $\tau_{crb} = 1/r_{crb}$ (the client read-buffer rate). The total (average) motion delay in this example is then

PLACE IMAGE FILE welch-eqn-3-trimmed.jpg

EXACTLY HERE

where r_{ss} is the sensor sample rate, r_e is the estimate rate, $\tau_e = 1/r_e$, r_{srb} is the server read-buffer rate, τ_{net} is the network transmission time, and r_{rcb} is the client read-buffer rate.

Note that this bound does not include any latency inherently added by pose estimate computations that also implement some form of filtering.

PLACE FIGURE 7 ABOUT HERE

4.3 Total Tracker Error

Summing the *static measurement error* from Section 4.1 and the dynamic error given by Section 4.2 we get a total error of

PLACE IMAGE FILE welch-eqn-4-5-trimmed.jpg

EXACTLY HERE

where Δt_m is from Equation (3), and includes the remainder of the graphics pipeline delay as described in Section 4.2.2. Clearly the final rotation and translation error is sensitive to both the user motion velocity, and the total delay of the tracker and graphics pipeline.

4.4 Motion Prediction

When trackers are used to implement VE or AR systems, end-to-end delays of the total system will result in perceived swimming of the virtual world whenever the users head moves. The delay causes the virtual objects to appear to follow the users head motion with a velocity dependent error. The sequence of events in a head-mounted display system goes something like that shown in Figure 8.

PLACE FIGURE 8 ABOUT HERE

The interval from t_0 to t_5 is on the order of 30 ms in the fastest systems and upwards to 200 ms in the slowest. If the user is moving during this interval the image finally displayed at t_5 will not be appropriate for the users new position. We are displaying images appropriate for where the user *was* rather than for where he or she *is*.

The most important step in combating this swimming is to reduce the end-to-end delay. This process can be taken only so far though. Each of the steps takes *some* time and this time is not likely to be reduced to negligible simply by accelerating the hardware.

After the avoidable delays have been eliminated one can attempt to mitigate the effect of the unavoidable delays by using *motion prediction.* The goal is to extrapolate the users past motion to predict where he or she will be looking at the time the new image is ready. As Azuma and Bishop point out (1995) this is akin to driving a car by looking only the rear-view mirror. To keep the car on the road, the driver must predict where the road will go, based solely on the view of the past and knowledge of roads in general. The difficulty of this task depends on how fast the car is going and on the shape of the road. If the road is straight and remains so, then the task is easy. If the road twists and turns unpredictably, the task will be impossible.

Motion predictors attempt to extract information from past measurements to predict future measurements. Most methods, at their core, attempt to estimate the local derivatives so that a Taylor series can be evaluated to estimate the future value. Several available commercial systems offer or support motion prediction. The differences among methods are mostly in the type and amount of smoothing applied to the data in estimating the derivatives.

The simplest approach simply extends a line through the previous two measurements to the time of the prediction. This approach will be very sensitive to noise in the measurements. More sophisticated approaches will take weighted combinations of several previous measurements. This will reduce sensitivity to noise but will incur a delay in responding to rapid changes. All methods based solely on past measurements of position and orientation will face a trade off between noise and responsiveness.

Performance of the predictor can be improved considerably if direct measurements of the derivatives of motion are available from inertial sensors. As described earlier, linear accelerometers and rate gyros provide estimates of the derivatives of motion with high bandwidth and good accuracy. Direct measurements are superior to differentiating the position and orientation estimates because they are less noisy and are not delayed.

Azuma and Bishop demonstrated prediction using inertial sensors that reduced swimming in an augmented reality system by a factor of 5 to 10 with end-to-end delay of 80 ms (Azuma and Bishop, 1994). Further, Azuma and Bishop (1995) show that error in predictions based on derivatives and simple models of motion are related to the square of the product of the prediction interval and the bandwidth of the motion sequence. *Doubling* the prediction interval for the same sort in input will *quadruple* the error.

5 Looking Ahead

Given the rapid pace of technology advances, and active work in the fields, we expect that by the time this handbook is in print, some of the information will be outdated. This aging process will continue, as is the case with almost any technology-related book. This is one reason we have attempted to wrap discussion of today's technologies (Section 3) in the context of the fundamental circumstances and issues that are likely to continue to be relevant for the foreseeable future.

Any attempt to look ahead into the future faces even more difficult challenges. And yet we want to attempt to share with the reader what appear to be some emerging trends and potential opportunities. Our hope is that the combination of the previous material and this brief speculation will combine to help make the reader a better consumer of the available technologies, and perhaps a better tracking systems engineer when needed.

We would claim that many of the fundamental challenges related to head and hand tracking *indoors* for one or two users have been addressed to a point where very interesting VR work is being done without major issues related to tracking. It is arguably not the dominant problem it once was, for circumstances involving a *few* people. The dominant research challenges are largely related to the competing desires for increased performance and reduced infrastructure. These challenges continue to be tackled in corporate and university labs.

However there remains the significant challenge of real-time, on-line head, hand, and full-body tracking for *teams* of individuals, as might arise in team training applications. The major issue is that of the sociability of the current approaches, as defined in (Meyer et al., 1992). As far as we know, all current commercial and research systems will begin to run into problems with more than a few co-located collaborating trainees. The issues include source or sensor bandwidth (can't flash or image fast enough), processing speeds, signal synchronization, and signal interference between/by nearby users. In fact we think there exists an interesting conflict between team training desires and the sociability shortcomings of today's centralized tracking systems: as trainees get closer to each other the tracking information will likely become more critical, and yet that's precisely when the likelihood of interference from each other increases. It seems that to accommodate teams of co-located collaborating trainees researchers might have to re-think the entire single-user centralized approach.

Perhaps the most exciting (to us) area of ongoing research related to tracking for training in virtual environments is related to tracking *outdoors*, as would be needed for military training exercises for example. The excitement in this area comes in two forms. First is the growing crossover between the computer graphics and computer vision communities. Some examples include work by Tobias Höllerer et al. at the University of California at Santa Barbara, Ulrich Neumann et al. at the University of Southern California, and Didier Stricker et al. at Fraunhofer-Gesellschaft. With simultaneous ongoing advances in computer vision algorithms and cameras, this synergy promises to provide some exciting capabilities in the not-so-distant future.

On the topic of cameras, the second form of excitement related to tracking outdoors is in the continually improving technologies that can be used outdoors. This includes rapid improvements to cameras, shrinking and more stable inertial sensors, improved GPS including differential signaling and pseudolites. Happily these improvements continue to be spurred on by other commercial demands.

On a final note, we think that perhaps acoustic/audio sensors are currently undervalued, and might find new favor in addressing both team-training needs and outdoor tracking. (Clearly the InterSense acoustic hybrids are one shining counter example.) With respect to team training, small acoustic devices could provide a complementary absolute reference for inertial sensors in a body-relative tracking scheme. The work by Vlasic et al. (2007) is a good example of this. With respect to tracking outdoors, we find it interesting that blind people use internalized models of environmental noise as an absolute reference for estimating their location. This includes sounds from traffic indicating the road, and sounds from air conditioning units indicating a building. Just as researchers have realized that the human combination of vision and inertial (vestibular) sensing is valuable, we might also recognize the added value of environmental sounds as yet another source of absolute geo-spatial references.

References

Allen, Bonnie, D., Bishop, G., and Welch, G. (2001). Tracking: Beyond 15 minutes of thought: Siggraph 2001 course 11. In *Computer Graphics*, Annual Conference on Computer Graphics & Interactive Techniques. ACM Press, Addison-Wesley, Los Angeles, CA, USA (August 12-17), siggraph 2001 course pack edition.

Atkeson, C. G. and Hollerbach, J. M. (1985). Kinematic features of unrestrained vertical arm movements. *J Neurosci*, 5(9):2318–2330.

Azuma, R. T. (1993). Tracking requirements for augmented reality. *Communications of the ACM*, 36(7):50–51.

Azuma, R. T. and Bishop, G. (1994). Improving static and dynamic registration in an optical see-through hmd. In *Computer Graphics*, Annual Conference on Computer Graphics & Interactive Techniques, pages 197–204. ACM Press, Addison-Wesley, Orlando, FL USA, siggraph 94 conference proceedings edition.

Azuma, R. T. and Bishop, G. (1995). A frequency-domain analysis of head-motion prediction. In *Computer Graphics*, Annual Conference on Computer Graphics & Interactive Techniques, pages 401–408. ACM Press, Addison-Wesley, Los Angeles, CA, siggraph 94 conference proceedings edition.

Bhatnagar, D. K. (1993). Position trackers for head mounted display systems: A survey. Technical Report TR93-010, University of North Carolina at Chapel Hill.

Deering, M. (1992). High resolution virtual reality. In *SIGGRAPH '92: Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, pages 195–202, New York, NY, USA. ACM.

Durlach, N. I. and Mavor, A. S., editors (1994). *National Research Council Report on Virtual Reality: Scientific and Technological Challenges.* National Academy Press, Washington, DC, USA.

Ferrin, F. J. (1991). Survey of helmet tracking technologies. *Proceedings of SPIE*, 1456:86–94.

Fischer, P., Daniel, R., and Siva, K. (1990). Specification and design of input devices for teleoperation. In *Proceedings of the IEEE Conference on Robotics and Automation*, pages 540–545, Cincinnati, OH.

Foxlin, E. (1993). *Inertial Head-Tracking.* M.s. thesis, Massachusetts Institute of Technology. by Eric Foxlin, before he changed his name from Fuchs. Thesis (M.S.)–Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, 1993. Supervised by Nathanial I. Durlach.

Foxlin, E. (2002). *Motion Tracking Requirements and Technologies*, chapter 30, pages 163–210. Lawrence Erlbaum Associates, Mahwah, NJ.

Gabbard, J. L. and Hix, D. (1997). A taxonomy of usability characteristics in virtual environments. Technical report, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Holloway, R. L. (1995). *Registration Errors in Augmented Reality Systems.* Ph.d. dissertation, University of North Carolina at Chapel Hill.

Jacobs, O. (1993). *Introduction to Control Theory.* Oxford University Press, second edition.

Livingston, M. A. and State, A. (1997). Magnetic tracker calibration for improved augmented reality registration. *Presence: Teleoperators and Virtual Environments*, 6(5):532–546.

Meehan, M., Razzaque, S., Whitton, M. C., and Frederick P. Brooks, J. (2003). Effect of latency on presence in stressful virtual environments. In *VR '03: Proceedings of the IEEE Virtual Reality 2003*, page 141, Washington, DC, USA. IEEE Computer Society.

Meyer, K., Applewhite, H., and Biocca, F. (1992). A survey of position trackers. *Presence, a publication of the Center for Research in Journalism and Mass Communication*, 1(2):173–200.

Mine, M. R. (1993). Characterization of end-to-end delays in head-mounted display systems. Technical report, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA.

Neilson, P. (1972). Speed of response or bandwidth of voluntary system controlling elbow position in intact man. *Medical and Biological Engineering*, 10(4):450–459.

Stanney, K. M., editor (2002). *Handbook of Virtual Environments: Design, Implementation, and Applications.* Lawrence Erlbaum Associates, Mahwah, NJ.

Taylor, R. (2006). Virtual reality peripheral network. `http://www.cs.unc.edu/Research/vrpn/`.

Vlasic, D., Adelsberger, R., Vannucci, G., Barnwell, J., Gross, M., Matusik, W., and Popović, J. (2007). Practical motion capture in everyday surroundings. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, page 35, New York, NY, USA. ACM.

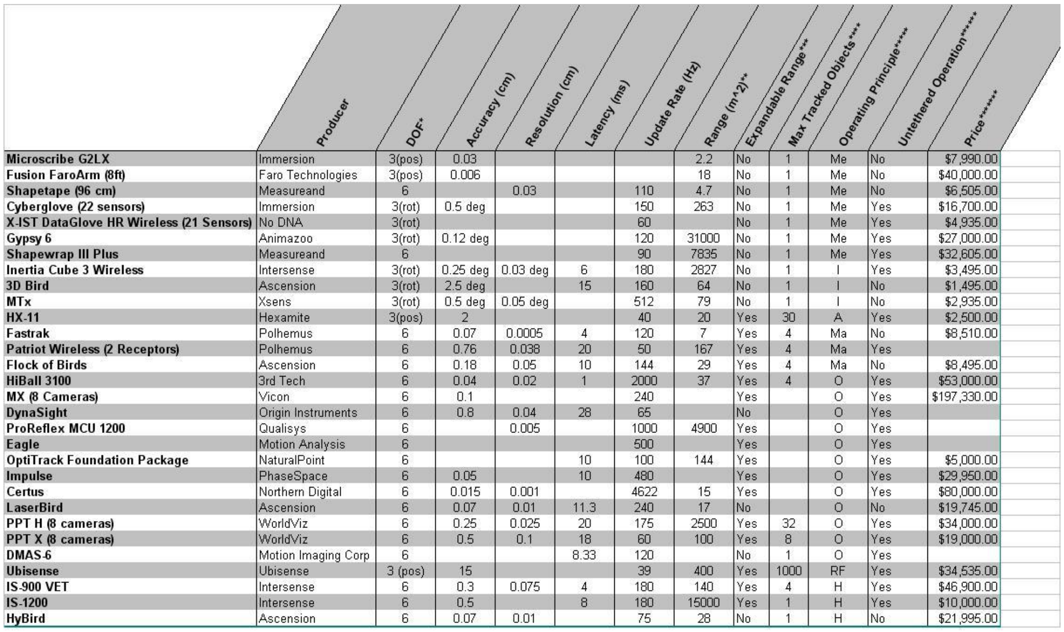
Welch, G., Allen, B. D., Ilie, A., and Bishop, G. (2007). Measurement sample time optimization for human motion tracking/capture systems. In Zachmann, G., editor, *Proc. IEEE VR 2007 Workshop on 'Trends and Issues in Tracking for Virtual Environments'*, Charlotte, NC, USA. IEEE, Shaker Verlag, Aachen, Germany.

Welch, G. and Foxlin, E. (2002). Motion tracking: No silver bullet, but a respectable arsenal. *IEEE Computer Graphics Applications*, 22(6):24–38.

Wloka, Mathhhias, M. (1995). Lag in multiprocessor virtual-reality. *Presence: Teleoperators and Virtual Environments*, 4(1):50–63.

Chapter Figures

Image file: davis-table-tracking-systems.jpg



	Producer	DOF*	Accuracy (cm)	Resolution (cm)	Latency (ms)	Update Rate (Hz)	Range (m^2)**	Expandable Range***	Max Tracked Objects****	Operating Principle*****	Untethered Operation******	Price*******
Microscribe G2LX	Immersion	3(pos)	0.03				2.2	No	1	Me	No	$7,990.00
Fusion FaroArm (8ft)	Faro Technologies	3(pos)	0.006				18	No	1	Me	No	$40,000.00
Shapetape (96 cm)	Measureand	6		0.03		110	4.7	No	1	Me	No	$6,505.00
Cyberglove (22 sensors)	Immersion	3(rot)	0.5 deg			150	263	No	1	Me	Yes	$16,700.00
X-IST DataGlove HR Wireless (21 Sensors)	No DNA	3(rot)				60		No	1	Me	Yes	$4,935.00
Gypsy 6	Animazoo	3(rot)	0.12 deg			120	31000	No	1	Me	Yes	$27,000.00
Shapewrap III Plus	Measureand	6				90	7835	No	1	Me	Yes	$32,605.00
Inertia Cube 3 Wireless	Intersense	3(rot)	0.25 deg	0.03 deg	6	180	2827	No	1	I	Yes	$3,495.00
3D Bird	Ascension	3(rot)	2.5 deg		15	160	64	No	1	I	No	$1,495.00
MTx	Xsens	3(rot)	0.5 deg	0.05 deg		512	79	No	1	I	No	$2,935.00
HX-11	Hexamite	3(pos)	2			40	20	Yes	30	A	Yes	$2,500.00
Fastrak	Polhemus	6	0.07	0.0005	4	120	7	Yes	4	Ma	No	$8,510.00
Patriot Wireless (2 Receptors)	Polhemus	6	0.76	0.038	20	50	167	Yes	4	Ma	Yes	
Flock of Birds	Ascension	6	0.18	0.05	10	144	29	Yes	4	Ma	No	$8,495.00
HiBall 3100	3rd Tech	6	0.04	0.02	1	2000	37	Yes	4	O	Yes	$53,000.00
MX (8 Cameras)	Vicon	6	0.1			240		Yes		O	Yes	$197,330.00
DynaSight	Origin Instruments	6	0.8	0.04	28	65		No		O	Yes	
ProReflex MCU 1200	Qualisys	6		0.005		1000	4900	Yes		O	Yes	
Eagle	Motion Analysis	6				500		Yes		O	Yes	
OptiTrack Foundation Package	NaturalPoint	6			10	100	144	Yes		O	Yes	$5,000.00
Impulse	PhaseSpace	6	0.05		10	480		Yes		O	Yes	$29,950.00
Certus	Northern Digital	6	0.015	0.001		4622	15	Yes		O	Yes	$80,000.00
LaserBird	Ascension	6	0.07	0.01	11.3	240	17	No		O	No	$19,745.00
PPT H (8 cameras)	WorldViz	6	0.25	0.025	20	175	2500	Yes	32	O	Yes	$34,000.00
PPT X (8 cameras)	WorldViz	6	0.5	0.1	18	60	100	Yes	8	O	Yes	$19,000.00
DMAS-6	Motion Imaging Corp	6			8.33	120		No	1	O	Yes	
Ubisense	Ubisense	3 (pos)	15			39	400	Yes	1000	RF	Yes	$34,535.00
IS-900 VET	Intersense	6	0.3	0.075	4	180	140	Yes	4	H	Yes	$46,900.00
IS-1200	Intersense	6	0.5		8	180	15000	Yes	1	H	Yes	$10,000.00
HyBird	Ascension	6	0.07	0.01		75	28	No	1	H	No	$21,995.00

Figure 1: Comparison of commercial off-the-shelf tracking systems.

Image file: welch-camera.jpg

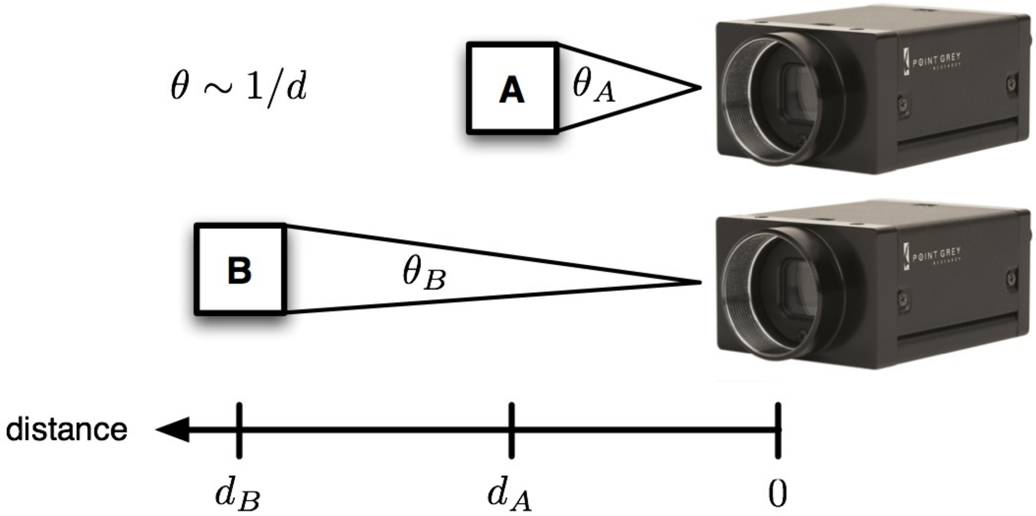


Figure 2: Relationship between distance and size of a target as imaged in the camera.

Image file: welch-light.jpg

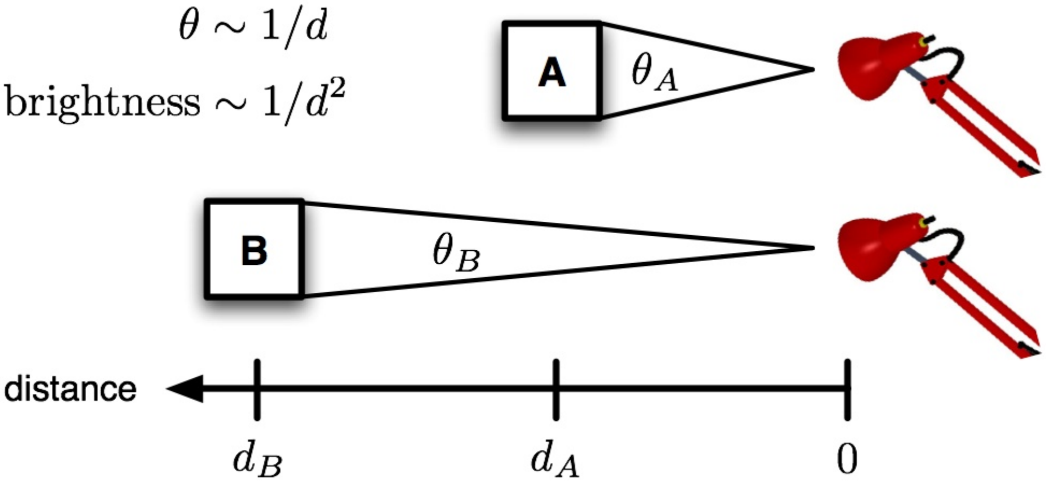


Figure 3: Relationship between distance and the amount of light (brightness) reaching a target.

Image file: welch-angle.jpg

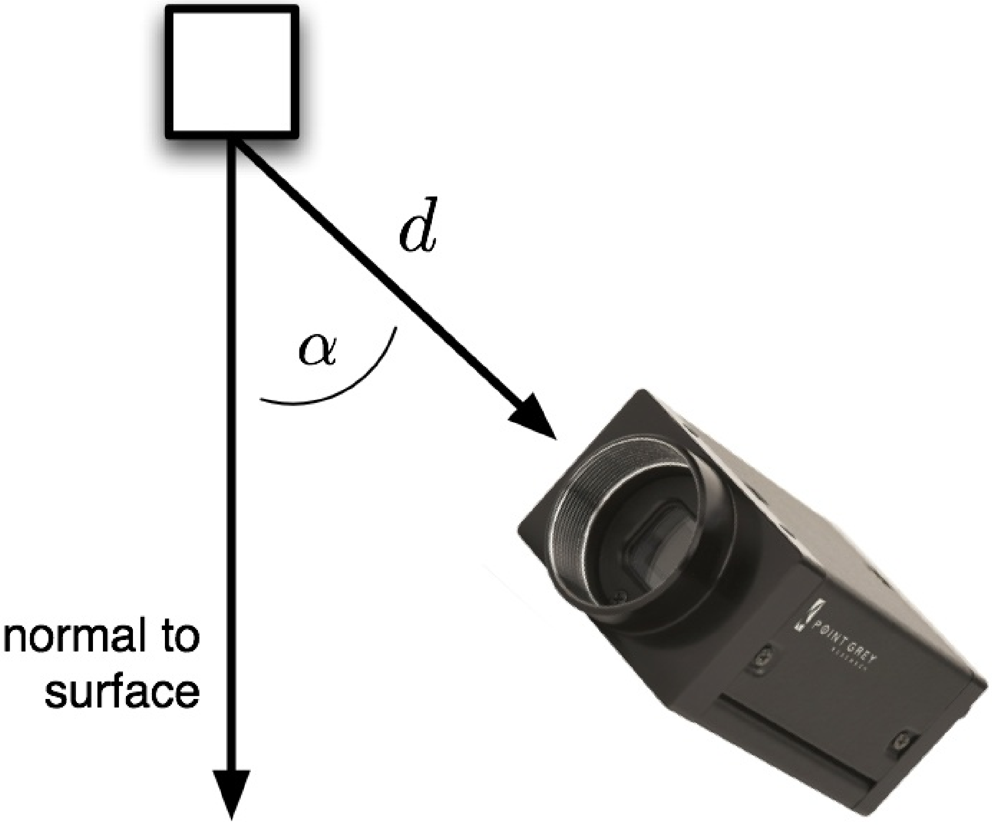


Figure 4: Relationship between normal-camera angle and the amount of light (brightness) reaching a target.

Image file: welch-eqn-1-2-trimmed.jpg

$$\varepsilon_{\mathrm{dyn},\theta} = \dot{\theta}\Delta t \tag{1}$$

$$\varepsilon_{\mathrm{dyn},x} = \dot{x}\Delta t \tag{2}$$

Image file: welch-eqn-3-trimmed.jpg

$$\begin{aligned}
\Delta t_m &= t_{m'} - t_m \\
&= \Delta t_{ss} + \Delta t_e + \tau_e + \Delta t_{srb} + \tau_{net} + \Delta t_{crb} \\
&= \frac{1}{2r_{ss}} + \frac{1}{2r_e} + \tau_e + \frac{1}{2r_{srb}} + \tau_{net} + \frac{1}{2r_{crb}} \qquad (3)
\end{aligned}$$

Image file: welch-eqn-4-5-trimmed.jpg

$$
\begin{aligned}
\varepsilon_\theta &\approx \varepsilon_{\mathrm{stat},\theta} + \varepsilon_{\mathrm{sa},\theta} + \dot{\theta}(\Delta t_m + \Delta t_g) \quad (4)\\
\varepsilon_x &\approx \varepsilon_{\mathrm{stat},x} + \varepsilon_{\mathrm{sa},x} + \dot{x}(\Delta t_m + \Delta t_g) \quad (5)
\end{aligned}
$$

Image file: welch-urbanscape.jpg

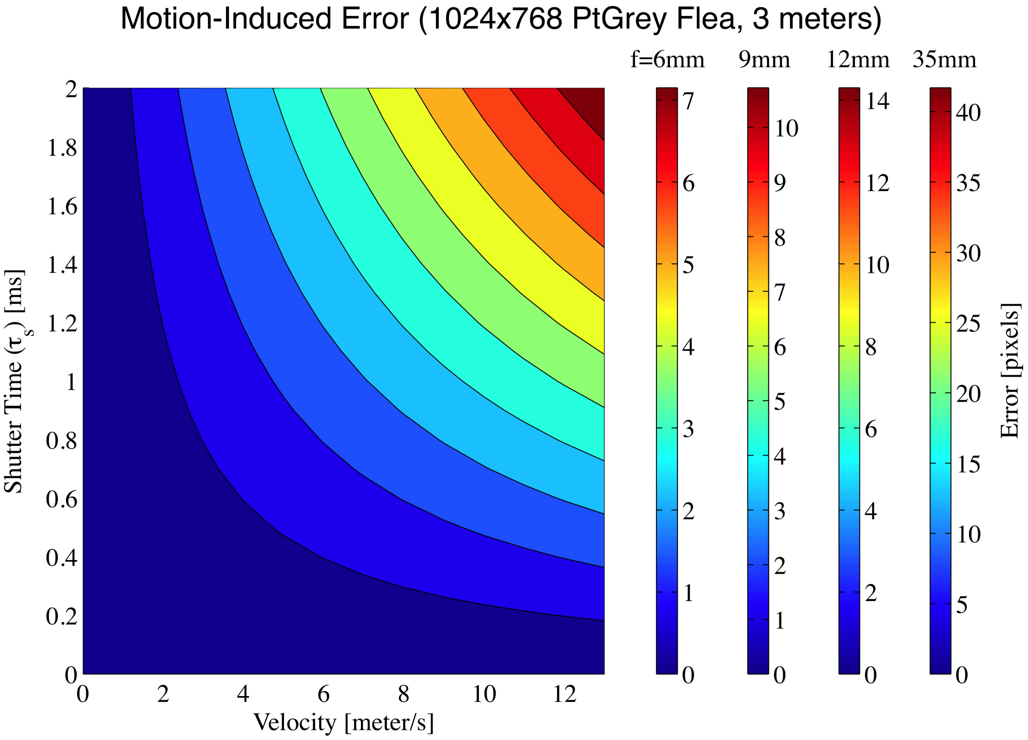


Figure 5: An example of motion-induced measurement error in a camera while imaging a dynamic environment (moving target or camera).

Image file: welch-sync-delay.jpg

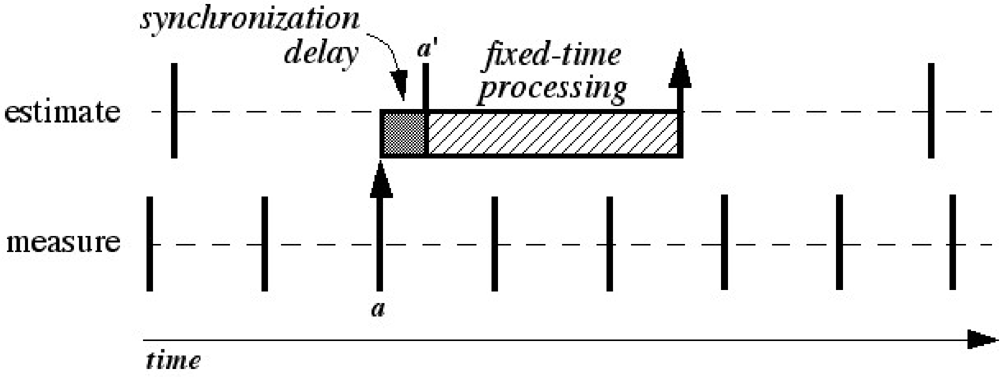


Figure 6: When a measurement is taken at time a, but not used to estimate the pose until time a', the intervening time is called *synchronization delay.*

Image file: welch-timeline.jpg

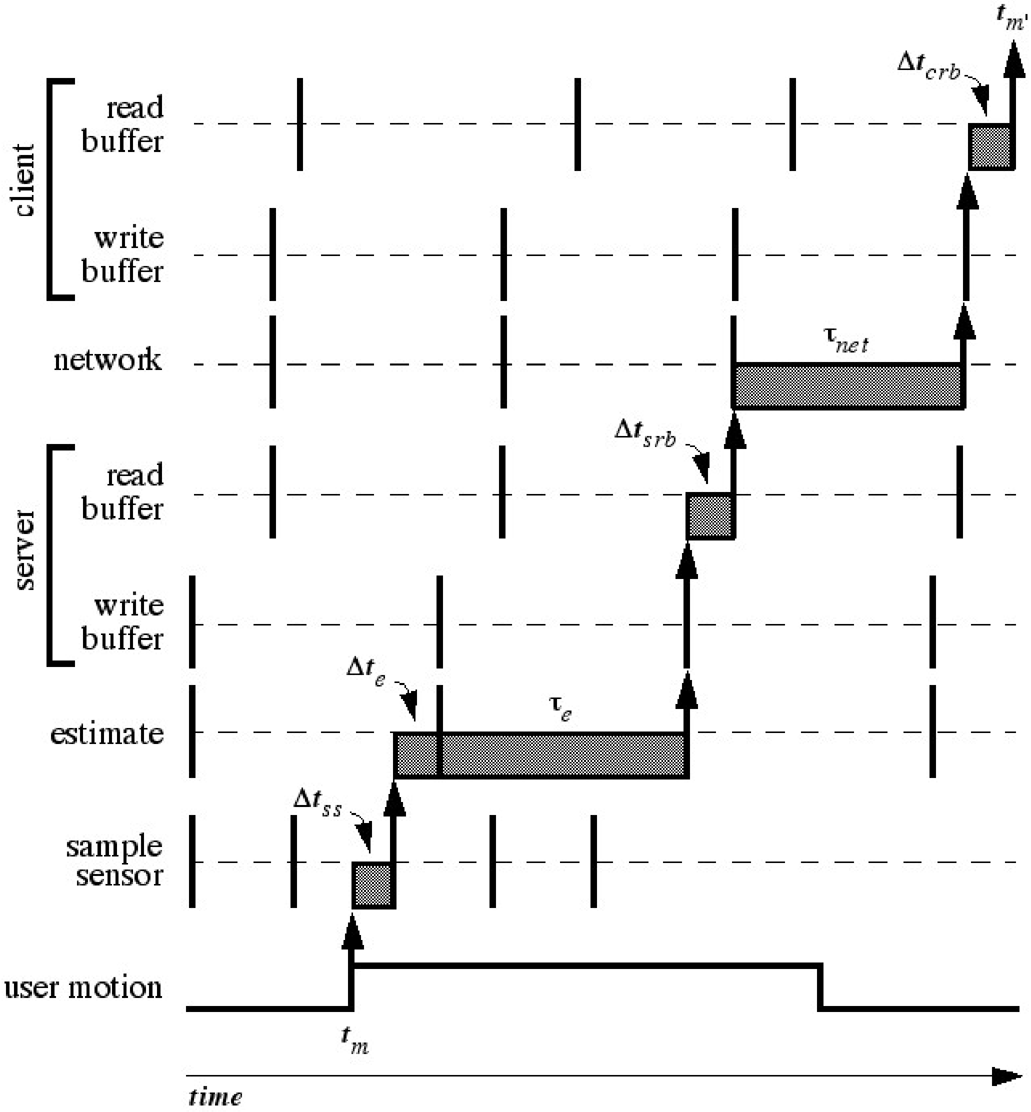
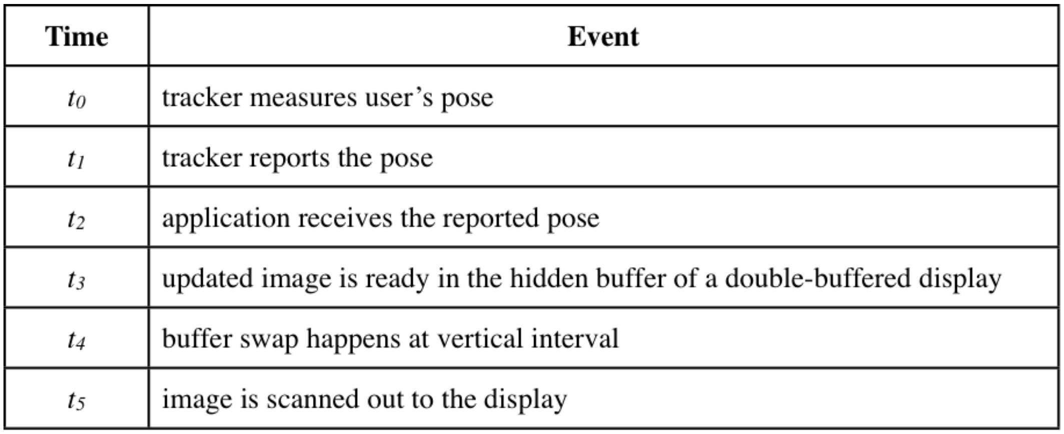


Figure 7: An example sequence of total tracker-related events and delays.

Image file: welch-table-prediction.jpg



Time	Event
t_0	tracker measures user's pose
t_1	tracker reports the pose
t_2	application receives the reported pose
t_3	updated image is ready in the hidden buffer of a double-buffered display
t_4	buffer swap happens at vertical interval
t_5	image is scanned out to the display

Figure 8: Time series of events in a head-mounted display system.