Extended draft version of Chapter 8 in Handbook of Virtual Environment Technology, Kay Stanney, Ed., Lawrence Erlbaum Associates, 2002

Chapter 7. Motion Tracking Requirements and Technologies

Eric Foxlin
InterSense Inc.

7.1. Introduction

The science of motion tracking is fascinating because of its highly interdisciplinary nature and wide range of applications. This chapter will attempt to capture the interdisciplinary approach by organizing the subject differently from the several excellent review articles already available (Meyer et al, 1992; Ferrin, 1991; Bhatnagar, 1993; National Research Council, 1995). These reviews tend to break trackers into several technology categories, and evaluate the merits of each technology by inferring from commonalties amongst the performance and ergonomics of existing trackers in that class. This survey will instead focus on what capabilities are required for various applications, and what methods can be used to realize these capabilities.

Separate examinations of tracking methods are given from the point of view of the physicist, who seeks to design new and better sensors, and the mathematician, who seeks to take whatever measurements are available from the physicist's sensors and calculate the best possible estimate (or prediction) of the object's motion. This dual taxonomy is necessary in order to support a new emphasis on *hybrid* tracking techniques. To simply append hybrid tracking as a new technology in addition to the usual 4 or 5 technologies would be a disservice - there are a combinatorial number of different hybrids possible, and each may behave quite differently. By categorizing physical sensing principles according to the type of observation they yield, and understanding how multiple observations can be mathematically blended together while accounting for the quality of each individual observation, a rationale is provided for the design and evaluation of these many potential hybrid systems. The founding philosophy for InterSense was that in most difficult tracking applications one can obtain better results at lower cost and weight by fusing measurements from a larger or more varied set of mid-quality sensors, rather than a smaller or more homogeneous set of high-precision sensors. This bias will come through clearly in the emphasis and organization of this chapter.

The organization of this chapter basically parallels the development cycle of a tracking system. The rest of Section 7.1 and Section 7.2 discuss several categories of motion tracking applications that will be considered, and the required tracking fidelity and ergonomics associated with each of these categories. This corresponds to the identification of a market need and the development of a specification for a tracker that will meet this need. Sections 7.3 and 7.4 discuss the physics and math that are used to design a tracking system to meet this specification. In Section 7.5, consideration is given to the engineering trade-offs that are involved in the implementation of the design during the productization phase of development. Finally, Section 7.6 discusses the vital real-time systems integration issues that must be handled correctly when interfacing the motion tracker into a larger system. If this integration is not done well the application will perform poorly no-matter how good the tracking system is, and therefore a large effort must typically be devoted to this back-end "Applications Engineering" support to make sure the end-user can really benefit from the new motion tracking system.

7.1.1. Taxonomy of Applications for Motion Tracking

Motion tracking as defined above has a much wider range of applications than what will be discussed herein. For example, it would include on-board navigation systems for aircraft, missiles, space vehicles, ships, submarines, UAVs, mobile robots, land vehicles, smart bullets, etc. It would also include external radar-based tracking of these items, optical systems designed to track hockey pucks, baseballs and golf clubs, sonar for fish, and RF tags for keeping track of bird migrations, mobile phones, stolen cars, soldiers in the field, or doctors in a hospital. All of these applications are close relatives, and some of the approaches discussed herein are borrowed from them, but we will focus attention on *tracking human head, limbs, or hand-held objects for purposes of interacting with 3-D computer-generated displays or teleoperators*. Within these computer-generated environments, motion-tracking devices have four main functions:

- View control
- Locomotion/navigation
- Object selection/manipulation
- Avatar animation

Even with this seemingly narrow definition of tracking, there is a vast array of different types of applications, and it is useful to categorize them according to aspects that affect the tracking requirements. This taxonomy of applications will be the basis for the discussion in Section 7.2 of performance requirements for motion tracking systems, since the requirements are different for each of the application categories. Two of the parameters that most affect the tracking requirements are the type of visual display and the types of manual interactions that the application allows with virtual and/or real objects in the environment. Due to visual capture (a.k.a. the ventriloquism effect), the presence or quality of auditory display is not thought to affect tracking requirements if there is also a visual display in use. Therefore, the taxonomy of applications will be based only on a breakdown of possible visual display modes and manual interaction modes. We prepare for the analysis of application categories by enumerating the possibilities in these two arenas:

7.1.1.1 Visual Display Modes

There are many types of visual display devices used in virtual environment systems, with new ones being invented every year. First, there is the distinction between Head-Mounted Displays (HMDs) and the like, and Fixed-Surface Displays (FSDs). The FSD shall denote any basically stationary display surface from a desktop monitor to a Virtual Model Display (VMD) such as the ImmersaDesk™ or Virtual Workbench™, to a display wall, on up to a full Spatially Immersive Display (SID) like the CAVE™ or VisionDome™. A further distinction can be drawn between head-tracked FSDs and cinematic FSDs. In head-tracked FSDs, the virtual camera viewing parameters are controlled in real-time by the user's tracked head position to achieve a first-person perspective – he sees the virtual world through his own eyes. This is the paradigm we normally associate with virtual environments. In cinematic FSDs, the viewing parameters are either fixed, pre-programmed, or controlled manually using a keyboard, mouse, joystick, SpaceBall™ or other input device. This is the paradigm we normally associate with animation and "pre-VE" interactive 3D applications such as CAD. For the rest of the chapter we will use the term FSD to mean head-tracked FSD unless otherwise noted. Note that when used as virtual environments, these FSDs are usually provided with stereoscopic 3D capabilities, using shutter glasses, polarized glasses, autostereoscopic display techniques or any other method. Stereoscopy is not required, but it will be assumed in the following discussion. The term HMD-like display shall likewise be used in a broad sense, to include both true HMDs and handheld relatives (eg. virtual binoculars, palmtop or camcorder-style displays, telescope sights, etc.) or boom-mounted displays. The distinction will be based on the method of view control used by the image generator. If the display movement is tracked, and the view is rendered assuming the eye is a fixed distance straight behind the display screen (without separately tracking the head), then it is an HMD-like display. For an FSD, the display doesn't move, but the user's eyepoint must be tracked relative to the display surface in order to calculate a possibly skewed through-the-window perspective projection. A third alternative exists, labeled "No Visual Display". What we really mean is that the tracking is not used for view control of the visual display, as it is in HMDs and FSDs. One example is a purely auditory virtual environment using tracked headphones to produce spatialized sounds that augment the user's perception of the real world. In addition, there is the realm of avatar-animation applications, in which the tracking is used to animate a virtual surrogate body (or part of one), usually for others to view. Although slightly outside the above-defined scope of 3D interactive motion tracking, this application will be at least briefly discussed since 3rd-person avatars sometimes play a role in multi-player interactive virtual environments.

Secondly, displays may be categorized according to whether they occlude the user's view of all real world objects, or allow real objects, at least the user's own hands, to be seen together with the virtual ones. The standard opaque HMD or hand-held display is occluding, while the non-occluding HMD-like displays include either optical see-through or video see-through designs, as well as head-mounted projectors. The optical see-through HMDs (OST-HMDs) so far cannot occlude real objects, while the video see-through ones (VST-HMDs) do have the ability to place virtual objects in front of certain real objects and occlude them *if* the depths of these real objects are known. Since this capability is rarely used to occlude the user's hands, which are constantly moving, all the ST-HMDs are categorized as non-occluding, although technically the VST-HMD potentially has some partial occlusion some of the time. The FSD examples listed in the definition above are all of a direct-view type. Since the user's hands are in front of the display surface and visible, these are classified as non-occluding displays. There are also reflected-view

FSD systems, in which an angled mirror is placed between the user's head and hands to occlude the hands. The FSD surface is mounted above and reflected in such a way as to cause the virtual objects to appear to be located behind the mirror where the real hands are. These systems are well-suited for high-dexterity reach-in applications where you don't want to block a portion of the display with your hands, especially if force-feedback devices are required which would block even more of the display area.

Combining these two criteria, we will take the universe of visual display modes to be those in the following table:

Table 1 : Visual Display Modes

	HMD-like displays	FSDs	No visual display	
Occluding	**I.** Opaque HMD	**II.** Reflected-view FSD		
Non-occluding	**III.** See-through HMD	Direct-view FSD	**IV.** Headphones	**V.** Avatar animation

The heavy boxes with roman numerals indicate the five major categories in the taxonomy of applications we are leading up to. The reflected and direct-view FSDs are lumped together as a single major category because they both require essentially the same quality of head-tracking in the absence of manual interaction. The major categories of the taxonomy are thus related to the display modality, which in turn imposes a certain minimum set of requirements on the head-tracking performance. The distinction between the reflected and direct-view FSDs has more to do with the different interaction modes they support. The interaction modes, enumerated in the following section, will be used as the sub-categories of the taxonomy.

7.1.1.2 Manual Interaction Modes

In addition to head-tracking for view control in the various display modes just discussed, many applications require motion trackers to allow the user to interact with the virtual objects that inhabit the virtual or augmentative environment. This interaction may be effected using the fingertips, which are typically tracked using a bend-sensing glove in combination with a 6-DOF tracker on the wrist, or using hand-held tools, or the end-effector of a tele-operated robot, or occasionally the foot or some other body part. Hand-held tools may be either short devices such as a pen or forceps, longer tools, such as a golf club, or devices which project their point of action outwards an arbitrary distance along a ray, such as a laser pointer or rifle. For brevity we will use the term **manipulandum** to indicate any of these objects as they appear in real or virtual form.

We will focus attention on the use of absolute position-mode trackers for direct object manipulation, rather than relative or velocity-mode control commonly used in CAD-like applications, which are already well-studied in the human-computer interaction literature. Going even further, we will only consider the direct manipulation of objects within reach of the manipulanda, ie. 100% "naturalism". Certainly, the tracked manipulanda will also be used for many "magical" forms of extended range selection and manipulation, as well as for controlling travel in some virtual environments. The full gamut of natural and magical interaction techniques is covered in Chapter 15. However, as argued by Mine et al (1997), object manipulation tasks can be completed more efficiently with objects held in the hand rather than floating in space offset from the hands, because this allows the user to take advantage of proprioception and eye-hand coordination skills honed over a lifetime. Furthermore, this type of interaction will place a greater accuracy demand on the motion-tracking devices, and therefore determines the required performance.

The types of direct object manipulations possible will vary depending on the display mode. In order to enumerate all the manual interaction modes possible for each display mode, we first provide a basis for the space of interaction modes, by breaking it down according to three key dimensions as illustrated in the following table:

Table 2: Dimensions of Manual Interaction

Manipulandum Visibility	Object Visibility	Haptic Feedback
N: None R: Real VC: Virtual co-located with unseen real VD: Virtual displaced from real A: Real augmented with virtual	N: None R: Real V: Pure virtual VC: Virtual co-located with unseen real VD: Virtual displaced from real A: Real augmented with virtual	N: None R: Real object contact V: Virtual simulated

Manipulandum Visibility describes the way the user sees each manipulandum that he is controlling. *Object Visibility* describes the way he sees the objects in the world in general (when there is no manual interaction), or the object being manipulated with the manipulandum in particular. Object Haptic Feedback describes how the feeling of contact with the object is achieved, if at all.

Any particular interaction with an object can be classified by selecting one entry from each column of the table. This would imply that there are 90 different interaction modes we have to discuss. Fortunately, for any given display mode, only a few of these are realizable, distinct, and useful. In the next section, we complete our taxonomy by enumerating the useful manual interaction modes (sub-categories indicated with letters) under each display mode (major category indicated with Roman numeral). The modes that are most commonly used are in boldface. Under each display category, the modes are listed in roughly increasing order of tracking difficulty.

7.1.1.3 The Taxonomy

I. Opaque HMD-Like Display

A. **NM-VO-NH: Viewing virtual objects with no manual interaction.**
B. **VDM-VO-[NH|VH]: Displaced virtual manipulandum manipulates virtual object, possibly with virtual haptic feedback.**
C. NM-VCO-RH: Unseen real manipulandum touches virtual object and feels its real counterpart.
D. **VCM-VO-[NH|VH]: Co-located virtual manipulandum manipulates virtual object, possibly with virtual haptic feedback.**
E. VCM-VCO-RH: Co-located virtual manipulandum touches co-located virtual object while real manipulandum feels corresponding real object.

II. Fixed Surface Display (FSD)

A. NM-VO-NH: Viewing virtual objects with no manual interaction.
B. **VDM-VO-[NH|VH]: Displaced virtual manipulandum manipulates virtual object, possibly with virtual haptic feedback.**
C. VCM-VO-[NH|VH](reflected-view only): Co-located virtual manipulandum manipulates virtual object, possibly with virtual haptic feedback.
D. **VDM-VDO-RH: Displaced virtual manipulandum manipulates displaced virtual object, while real manipulandum contacts corresponding real object.**
E. VCM-VCO-RH(reflected-view only): Co-located virtual manipulandum manipulates co-located virtual object while real manipulandum feels corresponding real object.
F. [RM|AM]-VO-[NH|VH](direct-view only): Real or augmented manipulandum manipulates virtual object, possibly with virtual haptic feedback.
G. [RM|AM]-AO-RH(direct-view only): Real or augmented manipulandum manipulates augmented object (which could be another manipulandum such as transparent palette), with real haptic contact feedback.

III. See-Through HMD-Like Display

A. NM-VO-NH: Viewing floating virtual objects with no manual interaction.
B. VDM-VO-[NH|VH]: Displaced virtual manipulandum manipulates floating virtual object, possibly with virtual haptic feedback.
C. **NM-AO-NH: Viewing augmented objects with no manual interaction.**

D. **VDM-AO-[NH|VH]: Displaced virtual manipulandum manipulates augmented object, possibly with virtual haptic feedback.**
E. [RM|AM]-VO-[NH|VH]: Real or augmented manipulandum manipulates floating virtual object, possibly with virtual haptic feedback.
F. **[RM|AM]-[RO|AO]-RH: Real or augmented manipulandum manipulates real or augmented objects with real haptic contact feedback.**

IV. **Audio-Only Display**
A. NM-VO-NH: Listening to floating virtual audio sources in the distance (or blindfolded).
B. NM-VCO-NH: Listening to virtual audio sources that are co-located with real objects that are in view.

V. **Third-Person Avatar Display**
A. Head and/or hands only.
B. Upper body.
C. Full body.

7.1.1.4 Is It Complete?

Certainly there are some experimental or even fielded systems that don't fit neatly into one of the subcategories listed above. For example, how would you classify a system in which a user is immersed in a surround-screen display environment and carrying a pair of virtual binoculars which he may occasionally raise to look out into the virtual yonder? When not looking through the binoculars, the user is looking on an FSD, and his head would need to be tracked with an appropriate fidelity for application II-A. However, when looking through the binoculars he is using an opaque HMD-like display and the binoculars should be tracked with the fidelity associated with application I-A. Thus, to a certain extent, a non-traditional or hybrid VE system can be understood as a combination of operating modes belonging to different subcategories in the taxonomy, and motion tracking specifications can be guessed for each of the various elements in the hybrid system. However, even this analysis of operating modes does not necessarily produce modes that fit perfectly into the taxonomy. In the above example, the binoculars must certainly be tracked at least as well as a pair of binoculars used alone (typical I-A application), but there is an additional requirement for the magnified view inside the binoculars to correspond to the terrain on the FSD behind the binoculars. The two views cannot be seen simultaneously, so this requires less accuracy than application type III-C, but clearly much more accuracy than type I-A. The taxonomy is therefore put forth as a framework for the discussion of tracker performance requirements in the next section rather than a means for enumerating every possible VE system configuration. It also may need to be extended if new display paradigms or interaction methods are invented.

7.2. The Human Factors of Motion Tracking

In the previous section, the realm of VE-related motion tracking applications was divided up into five major categories according to the type of display used. The display type dictates the performance required from the head-tracker just to view objects without any manual interaction. Thus, for each major category, the first sub-category (A) is the use of the display device without manual interaction. When manual interactions are added in the other sub-categories, this adds requirements on the hand-held tracking devices, and sometimes additional requirements on the head-tracking device, but the fundamental head-tracking requirements from sub-category A must still be satisfied as well.

In deciding the quality of head or hand tracking required for an application, there are several possible metrics for "good enough":

1. User feels *Presence* in the virtual world.
2. *Perceptual Stability*: Fixed virtual objects appear stationary, even during head motion.
3. *No Simulator Sickness* occurs.
4. *Task Performance Unaffected* by any tracking artifacts.
5. *Tracking Artifacts Below Detection Threshold* of a user who is looking for them.

Clearly, all of these metrics are inter-related, and without additional clarification of the experimental conditions, they are not even well-defined binary threshold tests. At the current level of understanding in the field, it is not even obvious how to rank these tests from least to most stringent. For example, keeping all tracking artifacts below detection threshold would imply total realism if the rest of the VE system were also perfect. That is, the user would not be able to tell whether the scene was real or virtual. This would seem to be the most stringent metric possible for "good enough" tracking, but how can we be certain the system would not cause simulator sickness due to some subtle artifacts of which the user is not consciously aware?

Presence is the most subjective of the criteria listed. There is no standardized test for measuring presence, and most of the different experimental tests that have been used rank the degree of presence on a continuous scale. These tests are typically based on questionnaire responses or on the ability of the system to provoke visceral responses which can be detected either through physiological changes or involuntary behaviors. Some of these metrics, especially presence, are highly dependent on factors other than tracking performance. Some of the largest factors influencing sense of presence are display-related: FOV, resolution, vergence, accommodation and graphics realism. Without the existence of a display which matches the human visual system capabilities, it is impossible to test the effect of tracker characteristics alone on presence.

There are several types of tracking errors which may contribute in varying degrees to destroying the sense of presence or stability, degrading task performance, or causing sickness. Different authors or manufacturers have focused on different specifications, or defined them differently, and every type of tracker has its own complicated idiosyncrasies that would require a thick document to characterize in complete detail. However, the following six specifications can capture the essential aspects of tracking performance that affect human perception of the VE system while the tracked object is still (static) or moving (dynamic):

Static

Spatial Distortion: Repeatable errors in the time-averaged outputs at different poses in the working volume. This encompasses the effects of all sensor scale factor, misalignment and nonlinearity calibration residuals and repeatable environmental distortions. In general, represented by a 6-input, 6-output mapping function.

Jitter: The portion of the tracker output noise spectrum that causes the perception of image shaking when the tracker is actually still.

Stability or Creep: Variations in the tracker output when still that are too slow to be seen as motion, but which may be observed to cause a change in the mean output position of a stationary tracker over time. This may be caused by temperature drift or random processes effecting the sensors, or by repeatability errors if the tracker is power-cycled or moved and returned to the same pose. There is not a clearly defined distinction between jitter and creep, as they represent the high and low frequency portions of a continuous noise spectrum. A reasonable cut-off might be to consider any motion slower than a minute hand (0.1°/s) in orientation and slower than 1 mm/s in translation to be creep, with everything else called jitter. Creep itself might be broken down into repeatability and short-term and long-term in-run stability. Providing a complete power spectral density (PSD) or Allen Variance plot can convey a more complete picture of the jitter and short-term and long-term stability, without making any arbitrary distinctions.

Dynamic

Latency: The mean time delay from a motion passing through a certain pose until an image corresponding to that pose is displayed. It is possible to specify the latency of the tracker and other subsystems separately, but they don't simply add up (see Section 7.6). If the system transfer function has linear phase, which is a reasonable approximation of typical VE systems where most of the lag is due to transport delay, then the latency is a single number independent of frequency.

Latency Jitter: Any cycle-to-cycle variations in the latency. When moving, this will cause stepping, twitching, multiple image formation, or spatial jitter along the direction the image is moving. Again this is a system specification and is discussed more fully in Section 7.6.

Dynamic Error (other than latency): Any inaccuracies that occur during tracker motion that cannot be accounted for by latency or static inaccuracy (creep and spatial distortion). This might include overshoots generated by prediction algorithms, or any additional sensor error sources that are excited by motion.

These differ slightly from the traditional specifications of **resolution**, **static accuracy** and **dynamic accuracy**, but they provide a description of motion trackers that maps more readily to the psychophysics of viewing virtual environments. In tracking systems whose resolution is limited by noise rather than quantization, the classical resolution would be equivalent to the jitter plus perhaps some of the short-term creep, depending on the time period of averaging that was used to make resolution discriminations. Static accuracy would include the fixed spatial distortion as well as the long-term creep, and dynamic accuracy would include everything on the list. Qualitative specifications, such as environmental **robustness**, **range, line-of-sight** requirements and **multiple object tracking** capability are important as well, and must be considered in addition to the six error performance specifications listed above for determining the suitability of a tracker for an application.

Unfortunately, very little research has been completed so far about the effects of tracker errors on the five virtual environment quality metrics listed above. More has been written about the effects of display parameters, image realism, and update rates on presence, and indeed it is impossible to quantify the effects of the tracker parameters independent of these. Until someone can build a perfect display driven by a very fast image generator, experiments to evaluate the required tracking performance will not be simple. The sections below summarize the few results that have been published, and offer some additional speculation, which remains to be evaluated by experiments in the future.

7.2.1 Tracking with Opaque HMD-Like Displays (category I*)*

7.2.1.1 Head-Tracking (I.A)

Opaque HMD-like displays range from highly immersive wide-FOV HMDs to less immersive hand-held devices such as binoculars or arms-length flat-panel displays. The highly immersive ones by definition strive for a magnification or zoom ratio of 1. Narrow-FOV binocular style devices, on the other hand, generally have magnification much greater than 1, and "magic lens" type flat panel displays vary. The bulk of this discussion will focus on immersive HMDs, with some commentary at the end about the implications of narrower FOV or higher magnification. To carry it to an extreme, we will imagine an HMD with a FOV and resolution matching the human visual system, so that we can focus attention on the effects of tracking performance.

The tracker parameters that have been studied the most are latency and frame rate. Frame rate is usually limited by the image generator not the tracker, but it contributes to the overall latency and is therefore discussed in tandem. So and Griffin (1995) studied how lag in tracked HMD’s affected performance on a target tracking task. The task was to keep an HMD-fixed cross-hair reticle on a moving target that moves around randomly with an r.m.s. target velocity of 2, 3.5 or 5°/s. The mean radial error in tracking performance increased linearly with increasing latency when additional 40, 80, 120 and 160 ms delays were added on top of the 40-ms base latency. The mean radial error increased more (about 50% increase) for the faster moving targets than for the slower target (about 30%). This paper shows that delays as small as 80 ms can degrade task performance, although the effect of smaller delays could not be measured because of the inescapable base latency of 40 ms. Recently, a system with base latency of 27 ms has been built and used to test subjects’ ability to notice additional latency in increments of 16.7 ms (Ellis, Young, Adelstein & Ehlrich, 1999). The psychometric functions for discrimination in a two-alternative forced choice experiment were found to be independent of the base latency; sensitivity to latency does not appear to follow a logarithmic Weber law as it does for most other stimuli. Subjects were just as able to discriminate a 16.7-ms increase in latency on top of a base latency of 27, 97, or 196 ms, from which we might extrapolate that they would also be able to discriminate a 16.7-ms latency from no latency. This is possibly bad news for those wishing to develop a virtual environment that users cannot tell apart from reality, but it does not necessarily imply that there is also decreased task performance at such low levels of latency. In a separate study (Ellis, Adelstein, Baumeler, Jense, & Jacoby, 1999) the relative importance of latency, update rate, and spatial distortion to a variety of metrics related to tracking task performance, perceptual stability, realism and simulator sickness was evaluated. The task involved manually tracking a target in a virtual world presented via an immersive HMD, so it is not certain that the results are entirely due to head-tracking performance. It was found that frame rate and latency had a significant impact on task performance and perceptual stability, with latency having a much more dramatic effect. However, this study only went down to a minimum latency of 80 ms, which was already known to degrade tracking performance, so it does not answer the question of whether the extremely small latency differences which were found to be subjectively noticeable are also detrimental to task performance.

Ellis, Baumeler, Jense, and Jacoby (1999) also evaluated the impact of spatial distortion on task performance and other metrics. The spatial distortion studied in these experiments consisted of an upward curling of about 15-30 cm in the corners of a 1.2 X 1.8 m working volume. Since the distortion affected both the head and hand trackers, the subject would only perceive the difference between two relatively nearby points, and this difference was changing relatively slowly. Not surprisingly, this gradual change in the head-to-hand transformation did not have a significant correlation with simulator sickness symptoms or major impact on performance, but it did create a small increase in the Cooper-Harper scale, which is highly correlated with perceptual stability and normalized r.m.s. tracking error. We can conjecture that distortions with higher spatial frequencies would produce greater impacts on all the metrics, since they would require more adaptation for a user moving around in the space.

No research has been reported yet on the effects of jitter on virtual environment users, although it seems obvious that jitter which is visible will reduce the illusion of presence, and may even contribute to simulator sickness if it is too extreme. The threshold of being detectable is not known, although it would be a very easy experiment to conduct. From experience, it seems that jitter of 0.05° r.m.s. in orientation and 1 mm r.m.s. in position is generally unnoticeable in an HMD with magnification of 1, but becomes fairly visible in a virtual binoculars simulator or virtual set camera tracker with 7X zoom. Note that when viewing distant virtual objects, tracker position jitter becomes irrelevant, and orientation jitter multiplied by zoom factor is all that matters. For viewing close objects, translational jitter becomes dominant. It is an interesting question whether we are sensitive to perceived jitter in world space or screen space (pixels) or some combination. If the former, than we might be psychologically more forgiving of an object jittering 2 pixels at 1 meter apparent depth (4 mm in world space) than an object jittering 1 pixel at 10 meters apparent depth (2 cm in world space). This again is an easy experiment. Other factors which affect the perception of jitter are display resolution and whether or not the graphics are anti-aliased.

Latency jitter has also been largely neglected in the many papers on the effect of latency on presence, simulator sickness and task performance. There is one paper that examines the effect of artificially imposed sinusoidally oscillating frame rate variations on task performance, using a task of grabbing a moving target object and placing it on a pedestal in a head and hand-tracked HMD virtual environment (Watson et al, 1997). The paper found that at "higher" frame rates of 20 fps, superimposing oscillating frame time variations of 10-20 ms amplitude on the 50 ms average frame time did not interfere significantly with task performance. However, at lower frame rates of 10 fps, varying the mean 100 ms frame time by +/- 60 ms did cause performance degradation if the oscillation was slow enough. This is probably because at slow oscillation frequencies, there would be quite a few frames in a row where the frame rate was only 6-8 fps. It should be noted that although the experiment introduced a high percentage of frame rate variation relative to the mean, the percentage of latency variation was much less because the system latency without the artificially introduced delays was already 213 ms.

Although latency variations for a fast frame rate system were not found to degrade task performance much, our subjective experience is that they cause a variety of very distressing artifacts such as image and object stepping, multiple image formation, twitching and jittering during head rotations. For a fixed latency system, the world appears to shift at the start and finish of a head rotation, but during the constant speed portion of the rotation the world appears stable, although displaced from its normal orientation by an angle proportional to the head speed times the latency. If there is latency jitter, then during the constant speed rotation, the world will jitter by an amount proportional to the latency jitter times the speed of rotation. Interaction with the refresh rate of the display device will cause additional effects such as multiple imaging that are explained in more detail in Section 7.3. More quantitative data is needed, but our experience with systems indicates that for very smooth apparent motion it is important to keep latency jitter below 1 ms.

7.2.1.2 Hand-Tracking (I.B-E)

The most typical interactive VE uses a head-tracker on the HMD for view control, and a tracking device held in the hand for object selection and manipulation. The hand or the device it holds cannot be seen directly, but a graphical representation is provided which is used to select and manipulate virtual objects. In mode I.B, the graphical manipulandum is displaced from the real one – perhaps the arm extension is multiplied to give the user longer reach, or the graphic is just displaced up and forward so the arms can rest comfortably in the lap instead of constantly reaching out into the workspace. For such applications, the absolute accuracy of the tracking is relatively unimportant. It will suffice that translating or rotating the real manipulandum will cause the virtual one to follow in a smooth and predictable manner. With practice the user's eye-hand sensori-motor loop adapts to the displacement, as discussed in Chapter 35 of this handbook. Good control and task performance can occur after adaptation is substantially complete, but the presence of significant latency interferes with this adaptation (Held, Efstathiou & Green, 1966). Also, the user will be relatively inefficient or even disoriented during the period of adaptation and may spend a lot of time trying to locate the visual icon of the manipulandum. After adaptation, there may be negative after-effects when returning to normal reality. To avoid all these problems, one may adopt mode I.D, which attempts to match the position of the visual representation to the actual position of the manipulandum so that the user can exploit natural proprioception without adaptation. This will clearly produce a very natural and easy-to-learn interface if the absolute accuracy of the tracking system is better than the accuracy with which proprioception can sense hand position with the eyes closed. Due to visual capture, the seen virtual manipulandum may be perceived as consistent with the felt real one even if the tracking errors are somewhat larger, but experiments need to be done to determine at what level of unpredictable tracking error this leads to noticeable sensory conflict and side-effects. Whether the virtual manipulandum is co-located with or displaced from the real one, latency in tracking the manipulandum probably has equal consequences. Held et al (1966), Ellis, Young, Adelstein and Ehrlich (1999), and Ware and Balakrishnan (1994) discuss the effects of this latency.

The use of virtual haptic feedback in modes I.B and I.D imposes no additional requirements on the tracking accuracy, since the moment of visual contact with the object is guaranteed to coincide with the onset of haptic feedback despite any tracking error. Modes I.C and I.E use real contact with the corresponding real objects to produce haptic feedback, and this feedback will only match up with the visual contact between the virtual manipulandum and virtual object if both the HMD and the manipulandum are tracked in 6-DOF with high absolute accuracy. With accurate tracking, these modes could lead to increased sense of presence as the user would be able to interact with real walls, doors and furniture whose felt positions correspond to their seen positions in a virtual environment walk-through. This level of tracking accuracy, sufficient for “visual-haptic registration”, is far greater than the accuracy required when only “visual-proprioceptive registration” is required (modes I.A, I.B and I.D), but is still less than the accuracy required for “visual-visual registration” which arises in certain modes of the category II and III displays.

7.2.2 Tracking with Fixed Surface Displays (category II*)*

7.2.2.1 Head-Tracking (II.A)

There is no experimental data yet, but theoretically head-tracking requirements for FSDs are far less demanding than for HMDs (Cruz-Neira, Sandin & DeFanti, 1993). The primary reason is that changing head-orientation, to first order, does not change the displayed scene. With HMDs, the most noticeable tracking artifact is orientation latency, because every time the head turns by any amount, the displays must be immediately updated with images corresponding to the new look direction. Even a very small orientation tracking latency causes the whole virtual world to first rotate with the head, then settle back to its expected position, resulting in a disturbing loss of perceptual stability. With an FSD, the appropriate images for all allowable look directions are already on the walls and can be seen immediately as soon as the head is turned towards them, without having to redraw anything. Head translation does require redrawing the screens to update the (off-axis) perspective projections, but these changes are slight for small head translations, so moderate translational tracking latency is not too noticeable unless you make unusually quick translations. Sensitivity to translational tracking errors is less problematic in an FSD than an HMD when viewing close objects, but worse for distant objects (Cruz-Neira et al, 1993).

Virtual environment FSD displays are almost always rendered in stereo, so the real tracking requirement is to track the 3-DOF position of each eyeball in order to render the left and right eye perspective views. Since it is easier to attach a sensor to the head than to each eyeball, this is usually accomplished by tracking the position and orientation of the head and using the orientation to calculate the positions of the two eyepoints as displacements from the head position sensor. There is therefore some sensitivity of the displayed images to head orientation tracking errors, but these errors only cause changes in the separation between the virtual eyepoints used to create the stereo. How much these temporary contractions in the virtual inter-ocular distance affect stereoscopic fusion or depth perception for a given quality of orientation tracking still needs to be determined by careful psychophysical experimentation.

7.2.2.2 Hand-Tracking (II.B-G)

While the head-tracking requirements in FSDs are comparatively lenient if one only wishes to use the display for visualization, most FSDs are used for interactive design activities in which tracking one or more manipulanda is essential, and here the requirements are essentially the same as in category I displays. With reflected-view FSDs, the real manipulandum is hidden behind the mirror and only a virtual representation can be seen, so the tracking requirements should be equivalent to the opaque HMD applications where the same conditions hold. In the direct-view arrangements such as the CAVE and Virtual Workbench, the virtual representation of the manipulandum is sometimes displaced from the real one so that it does not need to be accurately registered. In this case, the user is only looking directly at the virtual manipulandum and the same requirements for latency, accuracy and jitter should apply as in the previous cases. However, the real manipulandum may be visible in the peripheral vision, and this may make the user more able to discern latency of the virtual relative to the real. Much more critical however are modes II.F and II.G where an attempt is made to keep the virtual manipulandum precisely overlaid on the real manipulandum. Any errors in tracking are easily detected as visual misregistration, so the high tracking accuracy requirements of augmented reality (AR) apply. It should be noted that such attempts are also frustrated by the impossibility of focusing the eyes simultaneously on both the real manipulandum held at arm's length and virtual one displayed on the screen way behind it. Therefore most practitioners display the virtual manipulandum at least slightly displaced from the real one, which also eases the tracking requirements somewhat.

7.2.3 Tracking with See-Through HMD-Like Displays (category III*)*

7.2.3.1 Head-Tracking (III.A,C)

Mode III.C represents the most typical goal of augmented reality systems, to overlay virtual annotations precisely on top of real objects in order to guide the user who is performing manual operations on these objects. The system is only useful to the extent that the annotations are accurately registered with the objects, and thus absolute accuracy of the tracking system takes primary importance. This is in sharp contrast with the situation for opaque HMDs, where jitter, latency and latency jitter are crucial, but spatial distortion and creep are rarely noticed. For mode III.C applications, the most important tracker specifications are latency, spatial distortion and creep, because these cause visible misregistration. Jitter and latency jitter are also undesirable because they cause the annotations to shake or vibrate, but at least the entire world does not jitter as it does in I.A applications, so the risk of simulator sickness caused by jitter is lower. The causes of registration error have been discussed and analyzed in the AR literature (e.g. Drascic & Milgram, 1996; Holloway, 1997). The consensus is that latency is usually the worst offender, but once that is addressed, optical distortion caused by the HMD optics must be tackled before improving the tracking accuracy to millimeter levels can pay off.

Not all AR systems require every virtual object to be precisely registered on a real object. Mode III.A consists of displaying virtual objects that appear to be floating in mid-air within the user's view of the real world. This is useful for AR gaming, in which virtual beasts might jump in through the windows and attack the player, or for shared AR visualization, in which a 3-D model or dataset might hover above a table while multiple participants view it from different angles. Nothing has been written about the tracking performance requirements for this mode, but one would guess they will be slightly less demanding than for III.C applications since precise registration to the mm or even cm level may not be required. Thus a slight spatial distortion such as a systematic offset or nonlinearity may be less noticeable, but sensitivity to latency is probably nearly the same. The threshold for noticing latency in both III.C and III.A display modes is thought to be lower than for mode I.A immersive displays because there are real objects having zero latency visible for comparison. On the other hand, the unconscious effects of latency such as decreased presence or simulator sickness are probably worse in mode I.A because the whole world loses its perceived stability.

7.2.3.2 Hand-Tracking (III.B,D-F)

Tracked manipulanda are generally less discussed in the AR area, since the most common AR applications involve guiding a user to perform operations using a real tool on a real object. Since the object-tool interaction provides its own visual and haptic feedback, virtual graphics are usually used just to annotate the object than needs to be acted upon. In this case the primary purpose of tracking the tool would be to provide feedback to the AR application software about when and to what extent the action has been completed. For example, if the AR program prompts the user to turn a certain bolt 90°, it could determine, by tracking the wrench, when the bolt has been turned the correct amount, then automatically prompt the next action. To do this, the tool would need to be tracked with the same translational accuracy as the HMD in order to determine unambiguously when the actual tool has engaged the actual object. The orientation tracking requirement for the tool may be less stringent than for the HMD (which requires an orientation accuracy of about 0.1° to register an annotation to within 1 mm at arm's length). Mode III.F includes this situation in which a real manipulandum whose position is tracked manipulates a real object whose position is known. Either the manipulandum, the object, or both (or neither) may be augmented with graphics, but in all cases the tracking requirements are such that it would be possible to augment them both accurately enough that when the real objects engage, the augmentations do too.

Since it may be too expensive or complicated to track both the HMD and every needed tool with such high precision, some applications settle for the user's acknowledgement that an operation has been completed. The acknowledgement may be simply spoken or otherwise entered (tracking mode III.C) or a relatively simple hand tracker might be implemented to produce a virtual icon, displaced from the real manipulandum, which can be used to select and manipulate the annotations with VE-style interactivity (mode III.D).

7.2.4 Tracking for Audio-Only Displays (category IV*)*

The best 3D-spatialized sound is generated using headphones to deliver specifically processed audio streams to each ear. The sound delivered to each ear is generated by convolving the audio signal with the head-related transfer function (HRTF) which models the acoustic attenuation, time delay and spectral filtering that would occur if the sound were to propagate from the source to the current location of that ear. When headphones are used together with an HMD, the tracking requirements imposed by the visual display mode generally take precedence because visual acuity is higher than auditory localization acuity. However, in an FSD-presented virtual environment or a real environment with no visual display, the need to track head orientation is driven by the need to make the sounds appear to come from a certain stable direction or object in space even as the head rotates. It is thus reasonable to ask what quality of head-tracking would be required to fool the user into believing the sounds were really coming from a fixed source location even as he moves his head.

The directional resolution of binaural localization is best, about 1° in azimuth and 15° in elevation, directly in front of the head (see Chapter 4 for details). This implies that orientation tracking jitter and short-term stability of better than 1° is all that is required to make sure the sound source doesn't appear to wobble around while the user's head is still. Accuracy of 1° is certainly sufficient to make a sound appear to come from the direction of a certain object, and may be needed if the user is blindfolded or cannot look at the source. However, if the listener can see the alleged sound source then a phenomenon called visual capture or "the ventriloquism effect" makes the sound seem to emanate from the visual object, even if the auditory cues indicate a somewhat different direction. Auditory depth perception is even weaker, so it follows that modest resolution and accuracy for both orientation and position are sufficient for pure auditory displays. Less is known about the temporal response of the binaural localization system. Logically, if there is a system latency of Δt and the head pans at a constant angular rate ω, the apparent location of all sound sources would shift by angle $\omega\Delta t$ in the direction of the head rotation, just as visual objects do in an HMD. Therefore the orientation tracking latency requirement for perfect realism in an audio-only display may be just as critical as for an HMD. However, the ventriloquism effect may be able to compensate for this apparent auditory shift if it is only several degrees, so the detection threshold for latency in a spatialized audio system may actually be significantly higher than for an HMD. It is also unknown whether this auditory sensory conflict can contribute to simulator sickness, for example.

7.2.5 Tracking for Avatar Generation (category V*)*

Commercial full-body motion tracking systems are sold primarily for three application areas: 1) biomechanics & gait analysis, 2) motion capture, and 3) performance animation. Avatar generation in VE applications is most similar to performance animation, which differs from the motion capture done for film special effects or medical diagnostics primarily in that it must be done in real-time, and that motion that looks life-like is usually sufficient even if not completely accurate. In VE applications, body avatars may be presented to the person being tracked, or to 3rd –person viewers who may be other participants in a multi-user VE or non-participating trainers.

First person avatars are unnecessary in display modes II and III because users can see their own real bodies. In category I displays, users who look down and see no legs may feel disembodied and lose some of their sense of presence in the virtual world. If the VE involves manual interaction, then at least avatars for the manipulanda (hands or handheld tools) must be presented. Motion tracking requirements for these manipulanda have already been discussed with respect to the manual interaction modes of category I, II and III displays. In most systems, the avatars of these manipulanda are presented as isolated floating objects, leaving the user the feeling of having invisible or non-existent arms or even hands. This may not be important in a lot of applications, since the visual feedback from the manipulanda is enough to accomplish the interaction tasks. If it is psychologically desirable to have the hands and forearms that hold the tool visible, they can be drawn approximately most of the time without additional sensors, by using inverse kinematics (IK, Section 7.3.1.3). Nothing is really known about the required accuracy of representing these, but since the user's gaze and attention will usually be focused on the tip of the manipulandum, the forearm will only appear in the peripheral view and may not need to be drawn very accurately. On the other hand, if the user looks at an arm that is only tracked by IK and moves his elbow, he will see that the motion of the avatar does not follow correctly, which might defeat the purpose of rendering it. With the limited field-of-view available in current HMDs, the user will have great difficulty looking at his own torso so it is rarely important to provide full-body tracking in these applications. The legs may be rendered if desired by a crude algorithm such as placing the feet directly under the head when standing, or in front of the chair when seated. Occasionally an application will warrant actually tracking the legs.

Avatar display to 3rd-person viewers is more likely to provide a sufficient reason for full-body tracking. In a multi-participant simulation, a given player knows basically what he is doing himself - standing, squatting, or crawling for example - without visual feedback of his own body. Other players, however, have no idea, and this information may be important. Is the other player aiming a gun at me or holding up both hands in surrender? Unlike the precise head and hand tracking that is required for VE navigation and interaction, this information requirement is often qualitative in nature. Duplicating the precision tracking sensors that are used for the head and hands on all the other body segments may be more costly and cumbersome than necessary. Furthermore, line-of-sight problems, which are endemic to all of the high-accuracy tracking technologies, are multiplied when trying to put sensors all over the body. Ideas for new approaches to real-time full-body tracking which could potentially provide sufficiently lifelike motion with greater freedom and simplicity than today's prevalent optical and magnetic body-trackers are presented in Sections 7.3.1.2, 7.3.1.3, and 7.3.2.5.

7.3. The Physics of Motion Tracking

The previous two sections have focused on the first two stages of a motion tracking system lifecycle: identification of the type of application for which a tracking system is required, and analysis of the performance requirements for that application. This section should help with the next phase: choosing appropriate basic physical sensing technologies to measure the motion in the intended environment. Designing a sensor to make a specific measurement, for example a distance between two points, is a matter of selecting an appropriate physical principle to exploit, so this section will summarize the various physical laws and forces that are available, and how they may be used.

Classical physics views the world as a collection of particles that interact with each other through four fundamental forces: strong, weak, electromagnetic, and gravitational. The primordial branch of physics, mechanics, studies the motion of material bodies (particles or systems of particles such as rigid bodies) in response to these forces. The study of mechanics includes **kinematics**, which aims to describe the geometrically possible motions of objects without regard to the forces that cause those motions, and **dynamics**, which reveals the motions that will actually occur, given both the kinematic constraints and the existing forces and mass distributions.

For classical mechanics, which is adequate for describing the motion of human heads, limbs and handheld objects, all the motion follows from Isaac Newton's beautifully simple Laws of Motion. We will therefore start by looking at the kinematics (Section 7.3.1) and dynamics (Section 7.3.2) of human motion from a simple Newtonian perspective to see what tracking methods they may reveal. The analysis of the mechanics of large numbers of colliding molecules in air leads to **acoustics**, which we will examine next (Section 7.3.3) as another rich source of physical effects that have been used in motion tracking systems.

Of the four fundamental forces, the strong and weak forces only operate over extremely short distances inside the nuclei of atoms, and therefore have no obvious application in motion tracking sensors. A third fundamental force, gravitation, is indeed usable for motion tracking systems. Unlike the other three forces, gravity affects every particle with mass, and therefore plays a role in the dynamics of every system on the earth. Therefore, it is impossible to discuss the dynamics of motion in Section 7.3.2 without introducing the role of gravity, so the possible use of gravity in tracking systems is discussed there instead of creating a separate subsection. The final fundamental force, electromagnetism, has the richest variety of manifestations that are useful for remote sensing of object position and orientation. Section 7.3.4 discusses the proposed use of electric fields for tracking, and Section 7.3.5 surveys the use of magnetic fields, which have been the most common means of tracking until recently.

Sections 7.3.6 and 7.3.7 present techniques for exploiting electromagnetic wave effects in the lower and upper halves of the electromagnetic spectrum respectively.

7.3.1 Kinematics: Mechanical Tracking

In classical mechanics, a particle is represented as a point in space, with a constant mass m, and a time-varying position **r**(t) specified by three Cartesian coordinates x(t), y(t), and z(t). Tracking this particle would consist of reporting, whenever asked, the then-current three degrees-of-freedom of **r**(t), and is therefore called a 3-DOF (position only) tracking problem. In addition to its position **r**(t), the particle has a velocity **v**(t), which is the derivative of the position vector, **v**(t) = **r'**(t) = (x'(t), y'(t), z'(t)). According to Newton's Second Law, this velocity evolves according to **v'**(t) = **a**(t) = **F**(t)/m, where **F**(t) represents the sum of all the forces acting on the particle. In the absence of external forces, the velocity remains constant (Newton's First Law), and one could track the particle with no further measurements once one had obtained the position and velocity at any point in time. This is a complete statement of the kinematics of an unconstrained point particle, and captures the essence of the world's first motion tracking technique, **dead reckoning**. Sailor's slang for "deduced reckoning", dead reckoning was used by early mariners to calculate the vessel position by knowing the starting point, velocity, and elapsed time.

Your head is hopefully not a point particle. It is better approximated as a *rigid body*, which is modeled as a system of many particles constrained to maintain constant distances between one another. A rigid body requires 6-DOF to specify its pose (position and orientation), from which one can calculate the 3-DOF positions of all of the point particles that make up the body. The kinematics of a rigid body are somewhat more complex than a point particle, including an additional set of three nonlinear equations to relate the time derivatives of the three euler angles that describe the orientation to the angular velocity components. Where kinematics really gets interesting is in the study of multiple rigid bodies interacting through constraint equations, such as a robotic arm or a human body. Kinematics of such linkage systems can be applied to motion tracking in three different ways:

73.1.1 fixed-reference forward kinematics: mechanical linkages and pull-strings

Perhaps the most straightforward approach to head tracking is to make some kind of direct physical connection to the object being tracked, the displacement of which can be easily measured using potentiometers, optical encoders, rotary or linear variable differential transformers, or cable-extension transducers. The first HMD was tracked with such a mechanical contraption (Sutherland, 1968). Simplicity of mechanical design has dictated the two most common linkages for 6-DOF measurement: a single segment which extends telescopically, or two rigid segments jointed together at an "elbow." In either case, the arm is attached at one end with a 2-DOF "shoulder" joint to some fixed reference base, and at the other end is attached with a 3-DOF "wrist" to the object being tracked. The main causes of lag or inaccuracy are flexion of the linkages and transducer quality. The linkages can be made very rigid, particularly if they are short, and mechanical transducers such as potentiometers and optical encoders are available with extremely good precision and fast response.

The biggest problem with mechanical arm trackers is range. Making the arm segments longer lowers the mechanical resonance frequency, which may lead to unacceptable lag or oscillation. It also increases the inertia felt by the user. Even if one is willing to make such sacrifices to achieve larger range, the range of a two-segment arm is ultimately limited by ceiling clearance for the elbow as it folds upwards. As Meyer et al (1992) point out, they are not very "sociable", since multiple arms cannot be used easily in a shared space. Mechanical arm tracking remains an attractive alternative for situations where the infrastructure needs to be present anyway – boom supported displays, and force feedback devices are prime examples.

Another mechanical approach to head-tracking uses pull-string encoders mounted on the wall or ceiling, with the free ends of the pull-strings attached to the object being tracked. High-precision instrumented pull-strings, called cable-extension transducers, are available at low cost for industrial distance-measurement applications. There are two ways to measure head position using strings. The first method uses three pull-strings with the free ends of all three strings attached to a common point. The cartesian position of the point in space is found by trilateration. A second method would be to use a single pull-string to measure distance and also direction. The azimuth and elevation with which the string departs the reel box could be measured by running the string through a hole in a very light, low-friction joystick on its way out from the box. It should be noted that to get faster response requires a tighter string-retraction spring, which might result in an annoying tugging if the forces from the various pull-strings don't cancel each other out well. Like mechanical arms, pull-string arrangements have been used to provide force feedback (Ishii & Sato, 1994) and perhaps could also serve to help levitate a display.

7.3.1.2 moving-reference forward kinematics: bio-kinematic dead reckoning

The mechanical trackers in the last section are grounded to a fixed reference base, thus limiting the range and introducing a cumbersome mechanical linkage system. When tracking creatures with rigid skeletons, there is a mechanical linkage system already present that could be used for tracking purposes. Most casual users will not allow you to instrument their skeleton directly with joint angle encoders, but you can approximate the joint angle measurement using strapped-on goniometers. Goniometers (joint angle measurement devices) may consist of two rigid plastic parts that strap on, for example to the thigh and shin, with an instrumented hinge between them that measures the knee angle. Alternatively, they may dispense with the rigid attachments and use a flexible bend sensing material based on fiber optics or conductive foam that changes resistance when it bends. In fact, virtual goniometers with no physical connection can be implemented by attaching gravito-inertial sensors to each side of the joint, or a magnetic source to one side and sensor to the other, and calculating the difference angle between them. Molet, Boulic, and Thalmann (1999) attached magnetic sensors to each body segment, then drove an animation of a human using only the angular differences between the sensors. They cite several advantages over the usual technique of using the magnetically determined 6-DOF poses directly. In all cases, the tendency of the strapped on devices to slip and the inevitable movement of the flesh relative to the bones limit the accuracy. To achieve higher accuracy, some vendors have connected from one goniometer to the next using rigid rods, producing in effect an exoskeleton for tracking. This is more cumbersome than isolated goniometers at each joint, but allows freer movement than a grounded mechanical tracker.

Whether measured by goniometers, an exoskeleton, electromyography (Suryanarayanan & Reddy, 1997), or a body-suit, the sampled joint angles can be used to calculate the pose of each segment of the body, relative to a root node, normally at the pelvis, using simple forward kinematics calculations. The goniometer data completely determines the *shape* of the whole body, but the position and orientation of the body as a whole must be determined by other means, for instance by specifying the 6-DOF pose of the root node. This is often done by attaching a 6-DOF sensor to the root node, using magnetic, optical, acoustic, inertial, hybrid, or even grounded mechanical trackers, to provide an external reference pose for the root node with which to bootstrap the body-suit. Unfortunately, this root-node tracking system turns an otherwise self-contained tracker into a system that must make reference to the external environment, introducing the usual limitations such as range or line-of-sight. This type of system is therefore only used when full-body tracking is required for avatar generation, and it's main advantage over the external tracking system used alone is the capability to track the whole body.

It is theoretically possible to track the position and orientation of the whole body just using self-contained goniometry if certain assumptions are made. For example, if it could be assumed that at least one foot is flat on the floor at all times, then that foot could be treated as the root node while it is in contact with the floor (perhaps determined by pressure sensors on the soles of the shoes). As long as the contact is maintained, the position and orientation of that foot are known to remain constant at the same values they were set to when the foot first landed. During this time, the pelvis can be tracked relative to the grounded foot, and the moving foot relative to the pelvis, all using just forward kinematics. When the moving foot lands, it becomes the root node, and its current pose is maintained constant as the reference for the whole body until it loses that status. The upper body, or any portion of it, may be instrumented and tracked relative to the pelvis at the same time if required. Using only non-holonomic constraints on the feet, errors will gradually accumulate as a percentage of the distance traveled due to foot sliding or joint-angle measurement errors. The technique thus only provides **bio-kinematic dead reckoning**, much as an odometer and directional compass can provide dead reckoning for a wheeled vehicle. It should work much better than an ordinary pedometer, in that:

- it can keep track of height changes as the user climbs stairs
- it keeps track of direction changes and therefore position, not just total distance traveled
- each stride is individually measured, not just assumed to be the average stride for the user

If long term position accuracy must be maintained, then some method of correcting the accumulated position errors must be devised, perhaps based on map-correlation techniques or some simple light beams that are intercepted by photoelectric sensors on the ankles as the user walks through doorways or near landmarks.

We have seen no implementation of this concept and do not know if it would work adequately in practice. In normal walking, the foot does not remain flat on the floor, but rather rolls from the heel to the toe during the stride. Thus right off the bat the system described above would need to be augmented with some accurate foot roll sensors if it is to work with natural walking. The accuracy would depend critically on the quality of the ankle goniometers, since any error there would cause the whole body to be tilted incorrectly. It may be extendable to work with fixed seats, but would have difficulty with moveable office chairs, and would certainly fail if the user runs or jumps. An enhanced version called bio-dynamic navigation is introduced in Section 7.3.2 which in theory may be able to handle these cases better.

7.3.1.3 inverse kinematics

A final application of kinematic calculations, again to the problem of whole body tracking, is the use of Inverse Kinematics (IK) to deduce the pose of the whole body when only the poses of a few extremities, usually the head and hands, have been measured. Inverse kinematics problems are intrinsically a great deal more complex than forward kinematics, but a great deal of effort has been invested in developing algorithms to solve them, especially in the robotics community. In robotics, the desired pose for an end-effector (i.e. robotic hand) in order to accomplish a task is specified, and the robot must figure how to control all the individual joint angles to get to the end-effector into this pose. In a manipulator with redundant degrees of freedom, there may be a large number of different combinations of joint angles that result in the same end-effector pose, and the IK algorithm must solve for the combination which is optimal in terms of maximum speed, minimal work, or some other criterion. In the human motion capture problem, the human brain has already solved this problem and controlled the joint angles to get the hand to a desired position. A sensor measures the pose of the hand and uses an IK algorithm to try to guess what is the most likely combination of poses for the rest of the body segments resulting in the measured end-effector pose. Discomfort factors may become part of this calculation. The amount of “guesswork” can be reduced by using more sensors (for example by adding another sensor on the shoulder or the back), but the whole purpose of IK is to capture full-body motion using a minimal number of sensors. Badler et al (1993) used a fairly minimal set of four sensors, which happens to be the number provided by many commercial 6-DOF tracking systems, to control an animated figure called Jack by inverse kinematics. The sensors were placed on the head, the hands, and the waist. First, the spine was modeled as a single flexible rod with two axes of flexion and one of torsion, controlled by the waist and head sensors. Next, the arm positions were estimated by IK using the measured hand positions and shoulder positions consistent with the spine torsion. Finally, Jack’s center of gravity was calculated from the updated upper body configuration, and the legs were moved along animated stepping sequences to positions that could support the body mass.

7.3.2 Dynamics: Gravimetric & Inertial Tracking

7.3.2.1 gravitation and gravity

Dynamics is the branch of mechanics dealing with forces and the motions they induce. One force that acts on every object on or near the earth is gravity. Gravity is the sum of gravitation, the mass attraction specified by Newton’s Universal Law of Gravitation, and the centrifugal force due to the earth’s rotation. The mass attraction gravitation vector $\mathbf{g}_m$ pulls towards the center of the earth, but because the spin of the earth has caused it to flatten approximately 0.34% into an ellipsoid, $\mathbf{g}_m$ does not point exactly straight down relative to the surface of the ellipsoid. However, the local apparent gravity $\mathbf{g}_l$ to which a plumb bob aligns itself is the sum of the mass attraction gravitation and a centrifugal acceleration away from the spin axis of the earth, and it points nearly straight down relative to the surface of the ellipsoid which best describes the shape of the earth. In retrospect this is not surprising, since if the earth surface were not horizontal with respect to the local apparent gravity, it would eventually settle until it was. The geoid, which follows the mean sea level in the oceans and extrapolates what that sea level would be inside the continents based on a gravitational equipotential surface, is wavy but never deviates from the reference ellipsoid by more than 100 meters. The reference ellipsoid as defined by the World Geodetic System of 1984 (WGS-84) is universally used to express coordinates on earth using geodetic lattitude, longitude and altitude. Gravity can be approximated by the normal gravity field, which points exactly down relative to the WGS-84 ellipsoid and can be calculated using just the height and width of the ellipsoid, the mass and the angular rate of the earth. The normal gravity field varies smoothly from about 9.780 m/s^2 at the equator to about 9.832 m/s^2 at the poles. Any deviations of the actual local gravity from the normal gravity field, caused by the slight waviness of the geoid due to surface topography, are called gravity anomalies. The angular deviations of the gravity anomalies are up to about 30 arc seconds away from the vertical with magnitude variations up to 30 ppm.

7.3.2.2 accelerometers and gravimetric tilt sensing

An open-loop accelerometer is a proof mass constrained to move in one dimension, and restored to a center position in the housing by a spring. A pickoff measures the displacement of the proof mass from the center position, and this provides a reading of the non-gravitational acceleration acting on the accelerometer, traditionally called specific force, **f**. Many people are surprised to hear that an accelerometer is sensitive to all acceleration *except* gravity, because when you hook up an accelerometer and tip it up and down, its output swings +/- 1 g as if it were sensing the component of earth's gravity along its sensitive axis. In fact, the proof mass and housing are both equally affected by gravity, and an accelerometer in free-fall will read zero. The accelerometer placed on a table actually measures the upward restoring force exerted by the table to counteract the downward force of gravity. High-precision accelerometers that require a great deal of linearity are built using a closed loop servo-accelerometer design. Instead of a spring, an electromagnetic displacement pickoff and forcer are used to constantly rebalance the proof mass to its null position. The amount of current generated by the servo amplifier to maintain this null provides the output reading and is highly linear because the proof mass hardly moves at all. In recent years, such closed-loop servo-accelerometers have been implemented in silicon Micro-Electro-Mechanical-Systems (MEMS) technology, using capacitive, optical, or even quantum tunneling effects to detect the tiny (sometimes sub-atomic) displacements of the miniature silicon proof mass, and electrostatic restoring forces to rebalance it. MEMS accelerometers can already be produced with accuracies in the 10's of μg , and are beginning to replace the traditional quartz-flexure servo accelerometers in navigation applications. In addition, MEMS accelerometers with performance on the order of 1-10 mg are mass-produced for a few dollars each and widely deployed in commercial applications such as car air bag deployment.

As just described an accelerometer tipped in earth's gravity field provides a very sensitive reading of the component of gravity along its sensitive axis (but with the sign reversed). A high-quality servo accelerometer can have an accuracy of 10 μg . When in the horizontal position, this could measure a tilt angle of 0.0006°, an order of magnitude smaller than the vertical deflections due to gravity anomalies. Even the extremely low-cost MEMS accelerometers can measure pitch and roll of a static headset to within a degree or so. The so-called "sourceless" head-trackers that were included with consumer HMDs of the early 90's were based on this principle, using an inclinometer to measure pitch and roll, and a compass for yaw. The problem is that any inclinometer, whether it is made from two horizontal accelerometers, a mechanical pendulum or a fluid-filled bubble-level, must be able to sense the direction of the non-gravitational acceleration vector in order to determine tilt with respect to the gravity field. Any actual accelerations of the object horizontally will add vectorially to the vertical component that cancels gravity, producing a resultant that is no longer vertical. Even moderate head motions produce "slosh" errors of 5-10°. This is a fundamental limitation imposed by the physics of gravimetric tilt sensing, and cannot be overcome through improved sensor design. Judicious adjustment of damping, pendulosity and rotational inertia can decrease the frequency response to horizontal accelerations, but the low-frequency motions characteristic of head translation will always come through if the device has enough pendulosity to measure dynamic changes in tilt. A sourceless orientation-tracking alternative without slosh is presented in the next section.

7.3.2.3 gyroscopic orientation tracking

Despite their tremendous potential performance advantages, the use of gyroscopes came relatively late to human motion tracking (Foxlin, 1993). Before this, gyros were built with spinning wheels or lasers and were too large and expensive for human motion tracking, and the mechanical ones also produced distracting inertial reaction torques and noise. Motivated by the automotive market, the 1990's saw the commercial introduction of a new class of much smaller and cheaper gyros, now called Coriolis Vibratory Gyroscopes (CVGs). A CVG is a kind of mechanical gyro (as opposed to optical gyros such as the Ring Laser Gyroscope and the Fiber Optic Gyroscope), but it requires no spinning mass and therefore no bearings. Instead, a proof mass is made to oscillate at a fairly high frequency, usually in the tens of KHz, and a pickoff is provided to measure the secondary vibration mode caused by the Coriolis force $\mathbf{F} = \boldsymbol{\omega} \times \mathbf{v}$ which pushes the mass to vibrate in a direction perpendicular to the primary driven vibration. CVGs have been implemented with a wide variety of different geometries – vibrating rings, hemispherical shells, tuning forks, vibrating wheels, cylinders, triangular or rectangular prisms, etc – supported by a variety of clever suspensions. They have been made from quartz, ceramic, metal, and silicon elements, and the vibrations are caused and detected using piezoelectric, magnetic, electrostatic, or optical effects. An overview of the physics of CVGs is provided in Lynch (1998). In particular, the micromachined silicon CVGs have the potential in the long term to become extremely small and inexpensive, and achieve a high degree of integration possibly including three-axis sensing with electronics and processing on a single chip. Such developments will make MEMS gyroscopes even more attractive for wide-spread deployment in human motion tracking applications.

Some gyros measure orientation directly, as with vertical and directional spin-position gyros which maintain the spatial direction of the spin axis despite case motion by using isolation gimbals. Most modern gyros do not have gimbals, and they measure the angular rate of rotation around a sensitive axis. If three such rate gyros are assembled in a cluster with orthogonal sensitive axes, the three angular rate signals can be integrated together through a suitable three-dimensional numerical integration algorithm to keep track of the current orientation of the sensor assembly relative to where it started.

Regardless of the sensor technology or the orientation tracking mechanization, gyroscopic tracking provides a number of performance advantages that are crucially important to interactive graphics motion tracking applications. Because they are self-contained, inertial sensors can track with undiminished performance over an unlimited range without any line-of-sight or interference concerns. This is a significant advantage over all externally referenced tracking technologies, which cannot increase the working volume without degrading the signal-to-noise ratio and interference susceptibility. A second crucial advantage is extremely low jitter. Gyroscopes measure angular rate with very low noise, something like 0.001°/s for mid-quality gyroscopes. This produces no visible angular jitter, even at very high zoom ratios. Even with the very lowest-cost gyros, which currently have noise levels closer to 0.5°/s, the angular rate signal needs to be integrated, and in the process the noise is attenuated by the low-pass filtering effect of an integrator, so that the resulting angular jitter of about 0.02° is invisible in VE applications. A third important advantage of gyroscopic rotation tracking is speed. The outputs of gyroscopes can be sampled and used to update the orientation as often as desired. This can even be done right before a display refresh for final image shifting as described in Section 7. 6. Because of the low jitter of gyroscopically measured orientation, there is never a need to perform low-pass filtering for noise reduction, and the latency can be a fraction of a millisecond. What's more, the inertial angular rates measured by gyros can be used to perform prediction with several times greater accuracy than without them (Azuma, 1994). As indicated in Section 7.2.1.1, jitter and latency in *orientation* are the critical performance parameters for HMD tracking. Therefore, orientation tracking based on gyroscopic angular rate sensors should be used for any high-quality VE using HMDs, unless the tracking range is so small that an external tracking source can provide low enough jitter and latency in the intended environment.

The problem with tracking orientation using only gyros is drift. There are several causes of drift in a system that obtains orientation by integrating the outputs of angular rate gyros:

- **gyro bias**, $\delta\omega$, when integrated causes a steadily growing angular error $\phi(t) = \delta\omega \cdot t$
- **gyro white noise**, $\eta(t)$, when integrated leads to an angle random walk (Brownian Motion) process $\phi(t) = \int_0^t \eta(\tau)d\tau$ which has expected value zero because it is equally likely to wander in either direction, but a mean squared error growing linearly in time.
- **calibration errors** in the scale factors, alignments, and linearities of the gyros, produce measurement errors which look like temporary bias errors while turning, leading to the accumulation of additional drift proportional to the rate and duration of the motions.
- **gyro bias instability** means that even if the initial gyro bias is known or can be measured and removed, the bias will subsequently wander away, producing a residual bias that gets integrated to create a second-order random walk in angle. Bias stability is usually modeled as a random walk or Gauss-Markov process, and is often the critical parameter for orientation drift performance, since constant gyro bias and deterministic scale factor errors can usually be calibrated and compensated effectively.

The problem of gyro drift may be solved by using higher accuracy sensors and algorithms to keep the drift rate low, and requiring the user to return to a known position and restart after a certain period of time. Figure 1 illustrates the achievable drift performance for typical gyros of different grades. This simulation examines the effects of various gyro error sources over 20 minutes for an object that is not moving – scale factor calibration errors are thus not included and would lead to additional error accumulation as a function of the particular motion trajectory of the object. The simulation assumes initial gyro biases are known to 1/10th the hourly bias stability listed for each grade, i.e. 150°/hr, 1.5°/hr, and 0.0015°/hr for the commercial, tactical and navigation grade gyros respectively. Angle random walk caused by the gyro noise component has negligible effects over the extended simulation period, but it prevents the initial biases from being measured perfectly prior to the run, and the initial bias uncertainties are actually the dominant cause of drift for the first 36 seconds, after which they are surpassed by changing biases caused by the bias instability.

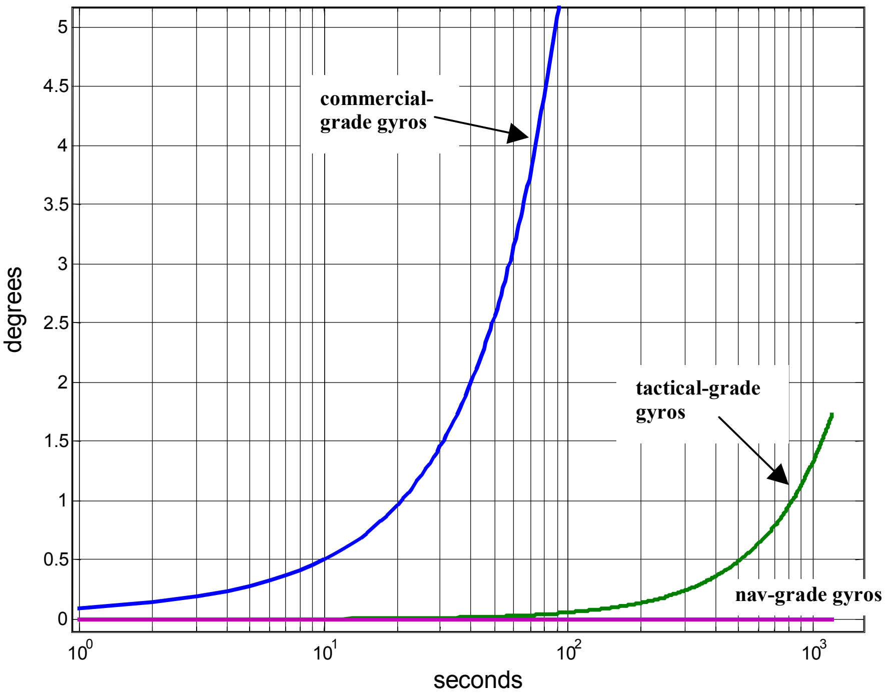

Figure 1 : Comparison of 1-σ random orientation drift performance of commercial ($1500°/\mathrm{hr}/\sqrt{\mathrm{hr}}$ bias stability), tactical ($15°/\mathrm{hr}/\sqrt{\mathrm{hr}}$ bias stability), and navigation-grade ($0.015°/\mathrm{hr}/\sqrt{\mathrm{hr}}$ bias stability) gyros over a 20 minute covariance simulation

An examination of Figure 1 leads to the conclusion that today's commercial grade gyros (devices used in automobiles, camcorders, model helicopters and low-cost head-trackers) can only be used for a minute or so before the drift becomes distracting and the user needs to reset the orientation tracker[1]. On the other hand, tactical-grade gyros used in short-range missile guidance are good enough for uninterrupted head-tracking for more than 20 minutes, which may be quite acceptable for many uses, and navigation-grade gyros are complete overkill if only orientation is needed. Unfortunately, the price ratio between tactical and commercial gyros roughly follows the 100-fold performance ratio, and they are also too large and heavy for head-mounted use. However, it is likely that MEMS gyroscopes over the coming decade will be gradually closing in on the performance of tactical grade gyros, and it may eventually become feasible to make a light and affordable head-orientation tracker out of gyros alone, requiring only occasional resets by the user.

Another solution is to correct the drift of the gyros using occasional measurements from another source. If only orientation needs to be tracked, it is possible to do this while still maintaining the advantages of a sourceless tracker (Foxlin, 1993, 1997). Gravimetric tilt sensing discussed in the previous section corrects drift in pitch and roll, and a magnetic compass can be used to correct drift in yaw if necessary. The problems with slosh in the gravimetric tilt sensors can be overcome because the gyros do not need to be corrected very often. Clever algorithms determine moments when the horizontal acceleration is likely to be negligible, and measurements at these times are introduced with higher weighting factors, for example using an adaptive Kalman Filtering approach (Foxlin, 1996). Because

[1] Head orientation trackers may significantly extend this by deadbanding the gyro outputs to zero when they are below a certain threshold. This eliminates the random drift when the head is still, but introduces "slippage" error accumulation whenever the head rotates below the angular rate threshold.

there may be sustained periods of motion when no corrective measurements can be used, a degree or more of orientation error may build up, and once calmness is restored the Kalman Filter may try to correct this rapidly enough to create some perceptible motion. To mask this, a perceptual enhancement algorithm holds back the correction until the user again begins to move, and applies it gradually proportional to the speed of head rotation.

Geomagnetic compassing provides a cheap and effectively sourceless way to correct drift in yaw – at least the source is provided by the earth and is present over most of its habitable surface. However, as described in Section 7.3.5.1, the accuracy of magnetic compasses in many environments is poor. Temporary magnetic disturbances can be detected and prevented from entering using something similar to the anti-slosh techniques used to screen the gravimetric tilt measurements. A degree of accuracy can be achieved which is suitable for many applications that just require the forward direction of a virtual world to remain consistent for a seated user. If the user is free to reorient towards the forward direction, or if high quality gyros are used and the duration of use is limited, than the use of a compass may not even be necessary. However in applications requiring registration accuracy, such as an outdoor AR application, magnetic compassing is not a suitable choice, and physics provides yet another option, still sourceless, called gyrocompassing. A mechanical gyrocompass makes use of the spin of the earth to cause the spin axis of a gyroscope to align itself towards true north through a damped gyroscopic precession. The same concept can be used with a stationary cluster of angular rate sensors to detect the direction of the earth's angular velocity vector, then project it onto a horizontal plane to find north. This requires gyros with sensitivity that is a small fraction of earth's 15°/hour rotation rate. To detect yaw with an accuracy of 0.2° would require gyros good to 0.05°/hour. Unfortunately, gyros of this caliber won't be small enough for comfortable use on an HMD for a great many years, if ever.

7.3.2.4 inertial position tracking

The previous section discussed the use of inertial angular rate sensors (gyros) for orientation tracking, which offers great advantages due to the self-contained, fast and noiseless measurements that can be made. In many applications it is also desirable to track position, and the aforementioned advantages would theoretically apply as well to a 6-DOF tracking system built with gyros to determine orientation and accelerometers to measure changes in position. In fact, this combination of sensors has been used successfully for Inertial Navigation Systems (INS) in ships, airplanes and spacecraft starting in the 1950's. In this section we review the basic operating principles of inertial navigation, discuss the differences between human-scale inertial tracking and geographic-scale inertial navigation, and analyze the present and future options for use of pure or aided inertial tracking in human-machine interaction.

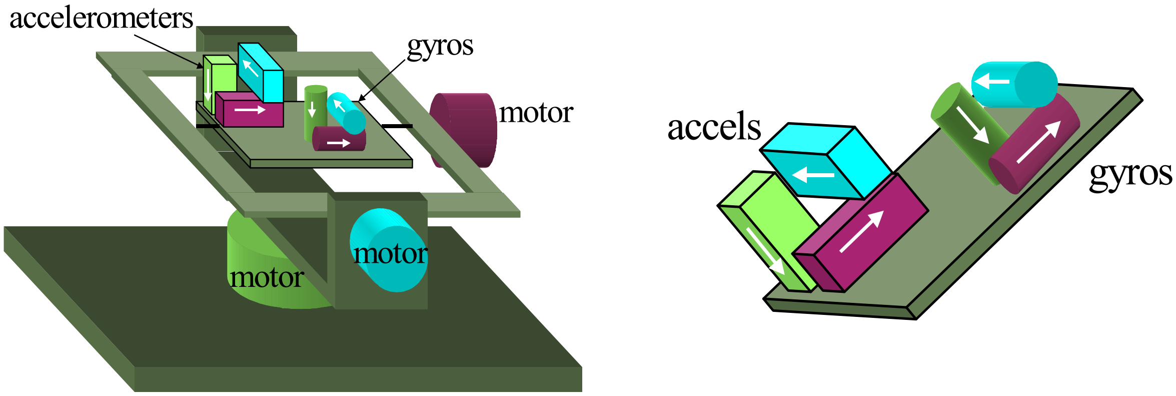

Figure 2a : Stable-platform INS **Figure 2b: Strapdown INS**

The operating principles for measuring orientation and position of a moving body using only gyroscopes and accelerometers have been well established in the field of inertial navigation systems (INS). The original navigation systems were built with a gimbaled platform (Figure 2a) that was stabilized to a particular navigation reference frame by using gyros on the platform to drive the gimbal motors in a feedback loop. The platform-mounted accelerometers could then be individually double integrated to obtain position updating in each direction. Most recent systems are of a different type, called strapdown INS (Figure 2b), which eliminates the mechanical gimbals, and measures the orientation of a craft by integrating three orthogonal angular rate gyros strapped down to the frame of the craft. To get position, 3 linear accelerometers, also affixed to orthogonal axes of the moving body, measure the non-gravitational specific force vector, **f**, of the body relative to inertial space. This specific force vector measured in body coordinates is resolved into geodetic navigation coordinates using the known instantaneous orientation of the body determined by the gyros. Position is then obtained by calculating the local gravity, adding it to **f** to get the total kinematic acceleration of the body **a** = **f** + **g**, and then performing double integration starting from a known initial position. Figure 3 illustrates this flow of information.

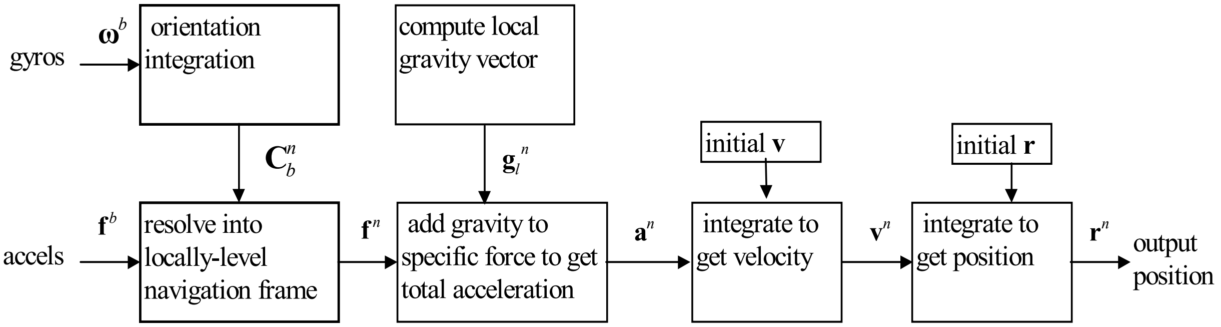

Figure 3: Basic Strapdown Inertial Navigation Algorithm

Drift in the linear position determined by an INS arises from several sources. First, there are accelerometer instrument errors corresponding to each of the 4 gyro errors listed above. Since position is obtained by double integrating acceleration, a fixed accelerometer bias error results in a position drift error that grows quadratically in time. It is therefore critical to accurately estimate and eliminate any bias errors. A much more critical cause of error in position measurement is error in the orientation determined by the gyros. An error of $\delta\theta$ in tilt angle will result in an error of $1g\cdot\sin(\delta\theta)$ in the horizontal components of the acceleration calculated by the navigation computer. Thus, to take proper advantage of μg-accurate accelerometers, the pitch and roll accuracy must be better than 1 μrad = 0.000057° for the duration of the flight, which puts a far more difficult task on the gyros than the accelerometers. In practice, it is the gyroscopes, not the accelerometers which limit the positional navigation accuracy of most INS systems, since the effects of gyroscopic tilt error will soon overtake any small accelerometer biases.

The scale of the human motion-tracking problem is vastly different from that of global navigation. Tracking is only required over a small area, but requires precision on the order of a centimeter or less, while with navigation a kilometer is often sufficient. The size and cost of the sensors must also be scaled down tremendously for human body-mounted use. Thus inertial human motion tracking would need to achieve far higher accuracy using tiny sensors than navigation systems are able to achieve using instruments far larger and more costly. We may reasonably ask the question whether purely inertial 6-DOF motion tracking will ever be viable.

Figure 4 shows the results of a simulation developed to answer this question. It shows the positional drift rates of navigation systems using sensors of various grades. As with the orientation drift plotted in Figure 1, the simulation only considers the effects of random error sources, predominantly those of the gyros, on drift accumulation for a stationary object over a 20-minute interval. If the object is moving, there will be additional drift due to scale factor calibration errors. The gyro and accel bias stability numbers listed in the figure represent typical values for the four grades of inertial measurement unit generally recognized in the inertial navigation market. In addition, a drift curve is plotted for a theoretical perfect inertial measurement unit to show the physical limits of inertial sensing for small scale position measurement. As discussed in section 7.3.2.1, there are gravity anomalies which create vertical deflections of up to 30 arcseconds or more, which corresponds to horizontal acceleration errors on the order of 150 μg. These have been mapped out in great detail by gravity surveys, and sophisticated inertial navigation systems are able to compute a local gravity vector with residual errors below 1 μg. It is impossible to drive these residual errors all the way to zero, though, because the local gravity conditions are constantly changing due to earth tides, seismic activity, and cultural features such as buildings and trucks. All these effects produce gravitational variations on the order of 0.1 μg – approximately equivalent to the pull of a 1-meter-radius stone ball close to its surface. Since it would not be practical to map out all these extremely local spatio-temporal variations in gravity, the simulation assumes the geophysical limit on the accuracy of inertial navigation to be 0.1 µg. Remarkably, strategic-grade systems come very close to reaching this limit for short-term navigation. However, MEMS gyros are currently pushing towards the tactical performance benchmark, and may even make it to navigation-grade performance levels over the next several decades, but are unlikely to go beyond. Therefore, human motion tracking systems that can maintain position to a few centimeters for more than a minute without external correction are not on the horizon. Nonetheless, 6-DOF inertial sensors are of prime importance in human motion tracking as part of hybrid tracking systems in which they impart robustness, smoothness, and low latency and reduce the requirements and therefore cost of the other technologies with which they are combined (e.g. Foxlin et al,1998). Techniques have been developed which allow inertial trackers to operate on motion-base simulators or moving vehicles without being disrupted by the platform motions (Foxlin, 2000).

	Commercial-Grade	Tactical-grade	Navigation-grade	Strategic-grade	Geophysical limit
Gyro bias stability	1500°/hr/√hr	15°/hr/√hr	0.015°/hr/√hr	0.000015°/hr/√hr	0°/ hr/√hr
Gyro bias initial uncertainty	150°/hr	1.5°/hr	0.0015°/hr	0.0000015°/hr	0°/hr
Accel bias stability	1 mg/√hr	100 µg/√hr	10 µg/√hr	0.5 µg/√hr	0 µg/√hr
Accel bias initial uncertainty	0.25 mg	10 µg	1 µg	0.1 µg	0.1 µg
Initial orientation alignment	1 arcsecond	1 arcsecond	1 arcsecond	0.1 µrad	0.01 µrad

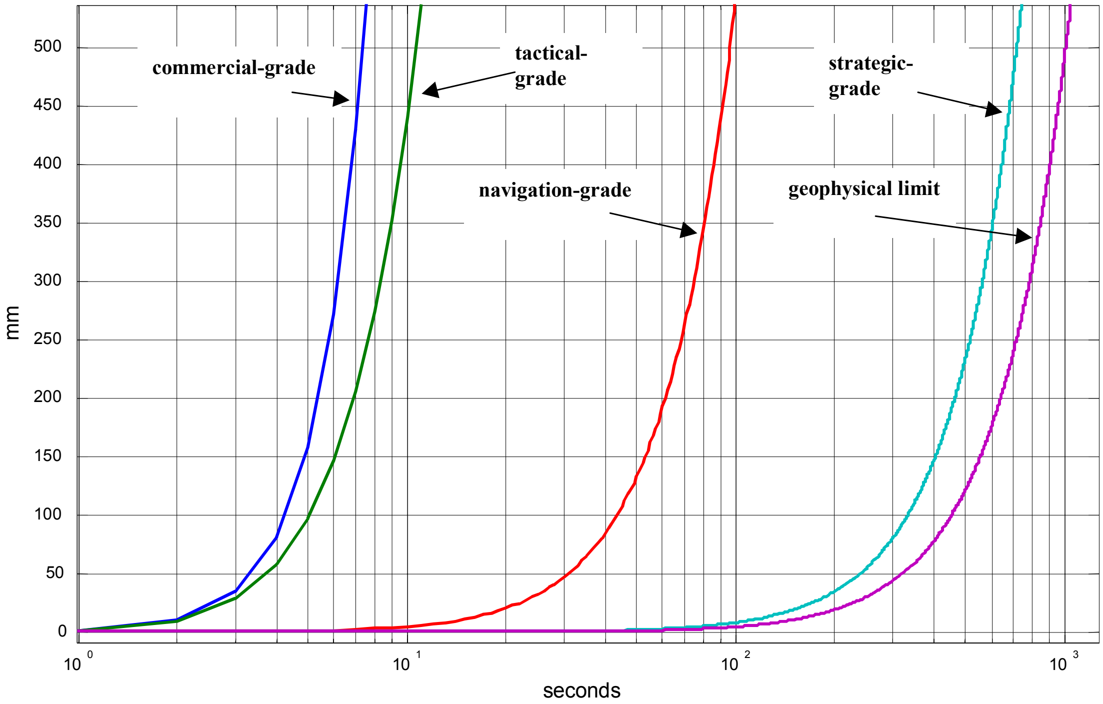

Figure 4: Comparison of 1-σ random position drift performance of commercial, tactical, navigation, strategic-grade, and "perfect" inertial navigation systems over a 20 minute covariance simulation.

7.3.2.5 body-dynamics-model-based tracking

The preceding discussion indicates that pure, unaided inertial navigation in its general form will never be an option for prolonged motion tracking on the human scale. The general form described above can track the motion of an arbitrary object moving along an arbitrary trajectory. All 6 degrees of freedom may evolve independently according to any continuous functions of time, within the limits on angular velocities and linear accelerations imposed by the maximum ranges of the sensors. In other words, the only kinematic constraint assumed in the development of the strapdown INS algorithms is that the body being tracked is rigid. Nothing needs to be known about its mass distribution, the forces acting on it, or kinematic constraints of connection or contact with other rigid bodies. If significant kinematic constraints or a priori dynamics models can be exploited, there is a possibility of tracking for longer time periods without the need for external position information. To illustrate this idea, we outline two hypothetical inertial tracking concepts below – zero-velocity-updating based on foot contact constraints, and body-dynamics-model-based tracking.

Researchers on the BodyLAN project at BBN have proposed a boot-mounted personal inertial navigation system (PINS) to help a soldier keep track of and report his own position during GPS outages (personal communications, 1997). It would operate just like an ordinary INS, except every time it determined that the foot was in contact with the ground it would reset the velocity to zero. Each mini-trip would be just as accurate as the ones before it, and total positional error would only accumulate linearly with the number of steps taken. This may be as accurate or more so than the bio-kinematic reckoning described in Section 7.3.1.2, but requires only one sensor mounted on the shoe.

If full-body motion capture is the goal, a dynamical model for the body may go a long way toward reducing the amount of sensing hardware needed. Suppose the body is modeled (crudely) using 17 rigid segments and two 3-DOF "goosenecks" to represent the lower and upper spine. If each segment were independently tracked with a 6-DOF sensor there would be 102 degrees of freedom. However, the kinematic constraints reduce this to about 39 independently controllable joint angles. Although there are a large number of muscles in the body, they work together in groupings to produce a net effect that can be approximately described by 39 torque values as a function of time. The complete state of the body can be described by 6 degrees-of-freedom for the root node plus 39 joint angles. These 45 generalized coordinates evolve according to a dynamical equation driven by the 39 muscle torques and constraint forces on any parts of the body that come in contact with the floor or other fixed objects. If the masses and moments of inertia of all the segments were approximately known (say by body scanning or just guesswork), it would be possible to produce lifelike animation by controlling the joint torques in a coordinated time sequence and enforcing the constraints with techniques from physically-based modeling. Conversely, estimating the motion of an actual human subject could potentially be reduced to estimating these 39 torques. By stacking the generalized coordinates, their derivatives, and all the causative forces and torques in the state vector of a dynamic system, it might be possible to develop a large centralized Kalman filter which can estimate the evolution of the joint angles from a set of indirect measurements such as angular velocities and/or linear accelerations. By modeling the unknown muscle torques with an appropriate stochastic process, a fairly small number of sensors may suffice to achieve observability of the motion over time. Using exclusively inertial and gravimetric sensors would make this solution sourceless, so the range of motion would be unlimited. Unlike the bio-kinematic reckoning approach, ballistic motions of the body are captured in the dynamics model, so the subject can run and jump without losing tracking. The potential for optimal and self-consistent motion estimation using any desired, even seemingly incomplete, combination of sensors warrants further investigation.

7.3.3 Acoustic Waves

One of the earliest position tracker technologies, used by Ivan Sutherland in his pioneering HMD work (Sutherland, 1968), and widely available today in many commercial products, is ultrasonic time-of-flight (TOF) ranging in air. Acoustic trackers can be very inexpensive (witness the Mattel "PowerGlove" that was sold in toy stores in the early 1990's and included a 6-DOF ultrasonic tracker as a subsystem). Alternatively, they can have fairly large tracking range or high accuracy. Typical drawbacks are latency, update rate and sensitivity to ultrasonic noises in the environment, but these can be mitigated in certain configurations.

All the known commercial acoustic ranging systems operate by timing the flight duration of a brief ultrasonic pulse. In contrast, the system used by Sutherland employed a continuous-wave source, and determined range by measuring the phase shift between the transmitted signal and the signal detected at the microphone. Meyer et al (1992) point out that this enables continuous measurement without latency, but can only measure relative distance changes within a cycle. To measure absolute distance, you need to know your starting distance and then keep track of the number of accumulated cycles. Another problem they did not mention, which may be the reason no successful implementation of the "phase coherent" approach has been developed, is the effect of **multipath** reflections. Multipath, a term also associated with radio transmission, refers to the fact that the signal received is often the sum of the direct path signal and one or more reflected signals of longer path lengths. Since walls and objects in a room are extremely reflective of acoustic signals, the amplitude and phase of the signal received from a continuous wave acoustic emitter in a room will vary drastically and unpredictably with changes in position of the receiver.

An outstanding feature of pulsed TOF acoustic systems is that it is possible to overcome most of the multipath reflection problems by simply timing until detecting the first pulse that arrives, which is guaranteed to have arrived via the direct path unless it is blocked. The reason this simple method works for acoustic systems but not for RF and optical systems is the relatively slow speed of sound, allowing a significant time difference between the arrival of the direct path pulse and the first reflection. We will analyze this in greater detail after briefly commenting on the nature of the ultrasonic transducers and signals that are commonly used in motion tracking systems.

Point-to-point ranging for unconstrained 3-dimensional tracking applications requires the use of transducers with radiation and sensitivity patterns that are as omnidirectional as possible, so that the signal can be detected no matter how the emitter is positioned or oriented in the tracking volume. The beamwidth of the sound emitted from a circular piston transducer is inversely proportional to D/λ, where λ is the wavelength and D is the diameter of the piston. To achieve an approximately hemispherical omnidirectional pattern (+/- 60° 3dB points) requires $D/\lambda = 0.6$ (Baranek, 1954). For a typical ultrasonic tracker frequency of 40 kHz, λ = 9mm, and one is required to use very tiny speakers and microphones with active surfaces of about 5.4 mm diameter. This is convenient for integration into human motion tracking devices, and helps reduce off-axis ranging errors, but the efficiency of an acoustic transducer is proportional to the active surface area, so these small devices cannot offer as much range as larger ones. To improve the range, most systems use highly resonant transducers and drive them with a train of about 6 or more electrical cycles right at the resonant frequency to achieve high amplitude. This results in a received waveform that "rings up" gradually for about 10 cycles to a peak amplitude then gradually rings down. For a typical pulse detection circuit which stops the timer at the peak of the envelope, this means the point of detection is delayed about 10 cycles, or about 90 mm, from the beginning of the waveform.

Returning to our discussion of multipath rejection, it should now be clear that as long as the reflected path is longer than the direct path by more than 90 mm, the detection circuit will have already registered the peak of the first pulse and stopped the timer before the reflected signal begins to arrive. Formalizing this analysis, let us call the time delay from the beginning of the pulse waveform to the peak or whatever feature is detected by the receiver circuitry t_d. Thus the displacement of this detection point from the beginning of the waveform is $R_d=c_st_d$, where c_s is the speed of sound in air. Let R_1 be the distance from the emitter to the reflecting object, and R_2 from the reflecting object to the receiver. We have said that any reflection path length R_1+R_2 that is longer than the direct path length R_0 by more than R_d will not corrupt the detection. Thought about slightly differently, objects outside the ellipsoid defined by $R_1+ R_2 = (R_0 + R_d)$ with the emitter and receiver transducers as its foci cannot possibly cause multipath ranging error because signals reflected off of them will arrive after the detection is complete. Conversely, objects inside this "ellipsoid of interference" risk causing reflections that interfere with the direct path signal to slightly distort the range measurement. Recalling high school geometry, the major axis of this ellipse is of length $2a = (R_0 + R_d)$, and the distance between the foci is $2c = R_0$, so the minor axis is of length

$$2b = 2\sqrt{a^2 - c^2} = \sqrt{(R_0 + R_d)^2 - R_0{}^2} = \sqrt{2R_0R_d + R_d{}^2}\ . \qquad (1.1)$$

For a transmitter-receiver separation R_0 of 2m, and the typical R_d of 90 mm mentioned above, the width of the ellipsoid in the middle would therefore be 0.6m. This allows significant opportunity for extraneous objects such as hands to come near enough to the line of sight between transmitter and receiver to produce reflections that corrupt the range measurement[2]. Much of our own development work has therefore focused on the design of circuitry to reliably detect the sound burst much earlier than the envelope peak. By detecting on the 2nd or 3rd cycle instead of the 10th, the volume of the "ellipsoid of interference" can be reduced by a factor of 4 or more, resulting in far fewer multipath ranging errors in typical real-world situations. This phenomenon is little discussed in the literature, but in our experience it is one of the most important issues for accurate ultrasonic tracking outside of controlled laboratory settings.

There are of course many other design trade-offs and considerations dictated by the physics of ultrasonic waves in air and transducer design. Ultrasonic noise sources such as banging metal fall off rapidly with increasing frequency, so operating at a higher frequency is very beneficial for avoiding interference, and also offers higher resolution due

[2] It may also seem that since the foci are inside of an ellipse, there is a danger of interference from objects behind the transducers, but the sensitivity patterns roll off dramatically by about 80° off axis, so objects beside or behind the transducer don't cause problematic reflections in practice.

to the shorter wavelengths. However, selecting a higher frequency also means less range due to the aforementioned problem with transducer size, and the frequency-dependent attenuation of sound in air. Attenuation of sound in air due to molecular absorption is basically negligible at 1 kHz, starts to play a significant role (compared to spherical spreading losses) by 40 kHz, and becomes the dominant factor in limiting range by 80 kHz. It also depends very significantly on relative humidity, with the humidity level that causes greatest attenuation shifting as a function of frequency (Baranek, 1954).

Since increasing the frequency much beyond 50 kHz is usually not an option due to range problems, other techniques may be considered to improve resolution and immunity to ambient noise sources. For resolution enhancement, a common trick is phase-locking. Using some envelope-based technique to determine the rough TOF, the final TOF is determined by finding the zero-cross of the carrier wave nearest to the rough TOF detection point. Since the slope of the carrier wave is very steep as it crosses zero, the location of the zero-cross point is not much affected by additive noise, and resolutions of a small fraction of a millimeter are easily obtained. However, no phase locking technique has been devised yet which can consistently pick out the same individual wave of the carrier every time, no matter how the transducers move, so the technique occasionally produces 9mm jumps in the output. To improve rejection of ambient noise, any communications engineer would probably suggest driving the emitter with a complex unique waveform, and detecting it with a matched filter at the receiver. However, this is not easy to accomplish because 1) the entire signature waveform has to be very short, in order to avoid multipath by getting an early detection, and 2) the piezoelectric transducers have narrow bandwidth and cannot be made to transmit very non-sinusoidal waveforms.

The main factors limiting accuracy of ultrasonic ranging systems are wind (in outdoor environments) and uncertainty in the speed of sound. In the vicinity of 25°C, the speed of sound varies with temperature as

$$c_s \approx 346.4\,\tfrac{\mathrm{m}}{\mathrm{s}} + (0.5813\,\tfrac{\mathrm{m}}{\mathrm{s}})(T - 25^o C) \qquad (1.2)$$

yielding 1.6 mm/m ranging error for every 1°C uncompensated temperature shift.

There are three techniques used to obtain the value of c_s used to convert TOF measurements into range values:

1) Mount a temperature sensor in the equipment (usually near one or more of the acoustic transducers), and use the reading in formula (1.2) to calculate c_s.
2) Mount a calibration transducer at a fixed known distance from one of the stationary reference transducers to directly calibrate the speed of sound traversing a known distance.
3) Use 4 reference transducers instead of 3, and calculate x, y, z, and c_s from the 4 range measurements.

If the temperature throughout the tracking volume is perfectly uniform, all 3 methods work well. If there is a gradient, the third method probably yields the most accurate results because it intrinsically calculates the average speed of sound over the actual paths of the ranging measurements.

The update rate of acoustic systems is limited by reverberation. Depending on the room acoustics, it may be necessary for the system to wait anywhere from 5 to 100 ms to allow the echoes from the previous measurement to die out before initiating a new one, resulting in update rates as slow as 10 Hz. The latency to complete a given acoustic position measurement is the time for the sound to travel from the emitter to the receivers, or about 1 ms per foot of range. This is unaffected by room reverberation and is usually well under 15 ms worst case. However, in a pure acoustic system with a slow update rate, the system latency is also affected by the need to wait for the next measurement. When used in a hybrid system to correct inertial sensors which are updating at a much higher rate, the acoustic sensors can be used in an inside-out configuration with no measurement latency (Foxlin et al, 1998).

7.3.4 Electric Field Sensing

Before the remarkable successes of nineteenth century physics, electricity, magnetism, radio, and light were all considered separate phenomena. In the wake of Einstein's Special Theory of Relativity, we can see that there is only one fundamental force required to explain all these effects, which is called the electromagnetic force[3]. Nonetheless, these four distinct manifestations of the electromagnetic force behave quite differently, and each yields a different set of motion tracking possibilities, so we cover them each separately in this and the following three sections.

[3] In fact, the electromagnetic force has already been unified with the weak force in the recent "electroweak" theory, and physicists continue to search for a Grand Unified Theory to unify all four fundamental forces.

The electric field is the only one of the four that has not been routinely used for motion tracking. However, if you have witnessed a demonstration of the experiment using a Van de Graaf generator to produce an electrostatic field that causes two sheets of gold leaf to splay apart, you will realize that electric fields are theoretically detectable. Reportedly, sharks and catfish do indeed sense weak electric fields to accurately determine object shape and distance and to communicate. The use of static electric fields from charged objects is not likely to produce any practical motion tracking systems, due to the difficulty of keeping the charge on the object from leaking off into the air, and the possibility that other objects may unintentionally acquire static charge. However, by using oscillating electric fields, even at very low frequencies, these problems are overcome, and the distance between conductive electrodes can be sensed with simple electronic circuits that measure capacitance.

Zimmerman et al (1995) have implemented a system for tracking hand motion or body location using capacitive sensing of electric fields. A radiating electrode and a ground electrode are set up in a workspace, and any conductive object that comes between or near them affects the capacitance between them. A human hand inserted between the plates acts as a conductive object connected to a large conductive mass that acts as a charge reservoir ground if it is outside of the capacitor gap. As the hand is inserted further into the capacitor gap, it shunts away more of the electric field lines that otherwise would have reached the ground electrode, and therefore reduces the capacitance. In another arrangement, the emitter electrode is placed close to or in contact with the person's body so that the excitation field is capacitively coupled into the person and their entire body becomes an electric field radiator. Then, the closer the hand approaches the ground electrode, the greater the capacitance.

They observe that the system is capable of low latency, high resolution, and can be built with lightweight low-power electronics that could be integrated into a palmtop or even wristwatch computer. The sensing is unaffected by non-conductive objects and requires no contact with the user's body. However, the electric field geometry in the dipole near-field regime is too complex for accurate analytical modeling and some form of training or calibration procedure is necessary to convert the capacitance measurements from multiple electrodes into a position estimate. The potential for precision tracking with electric fields is not good. The system will track the whole human body as a "blob", or that part of the body that is inserted in the electric field region. It can therefore be used with large-scale electrodes to tell where people are in a room or with smaller scale electrode arrangements to track where a hand is when it is inserted into the region between the electrodes. In this latter example, it may be very useful for qualitative gesture tracking, but not for quantitative precision pointing, since the indicated position will be affected by the shape of the hand, the arm, and the stance of the body if it is too close to the field region.

7.3.5 Magnetic Field Sensing

Unlike electric fields, magnetic fields are unaffected by the presence or absence of human bodies and other non-metallic objects in the environment. This offers a tremendous opportunity, because it enables magnetic trackers to overcome the line-of-sight requirement that plagues acoustic, optical, and externally-connected mechanical trackers. Magnetic tracking technologies have a long history, and to date have been more widely deployed in human-machine interface tracking applications than any other technology.

7.3.5.1 Geomagnetic Sensing

Loadstone (magnetite) was known to the ancients, who eventually discovered that if suspended properly it tends to align itself towards the North, and promptly invented the magnetic compass. Thus, the world's first motion tracking system was a yaw direction indicator based on the earth's magnetic field. Modern navigators learned how to build electronic compasses with digital readout, and these became the head-tracker of choice for consumer HMD's of the early 90's, because their ultra-low cost offset the poor performance and lack of position tracking capability. The earth's magnetic field from the surface outwards approximates a magnetic dipole field (ie. the magnetic field pattern produced by a small circulating current loop). A popular theory holds that this field is produced by a circulating current in the earth's molten iron outer core, propelled by the rotation of the earth and interacting with a permanent magnetic field whose origin is not yet known. In any case, the radial and tangential components of a dipole magnetic field are given by the equations:

$$B_r = \frac{2m\cos\theta}{r^3}$$
$$B_t = \frac{m\sin\theta}{r^3} \qquad (1.3)$$

where m is the magnetic dipole moment, θ is the angle away from the north pole, and r is the radius from the center of the dipole source (derived from Purcell, 1965, p.365).

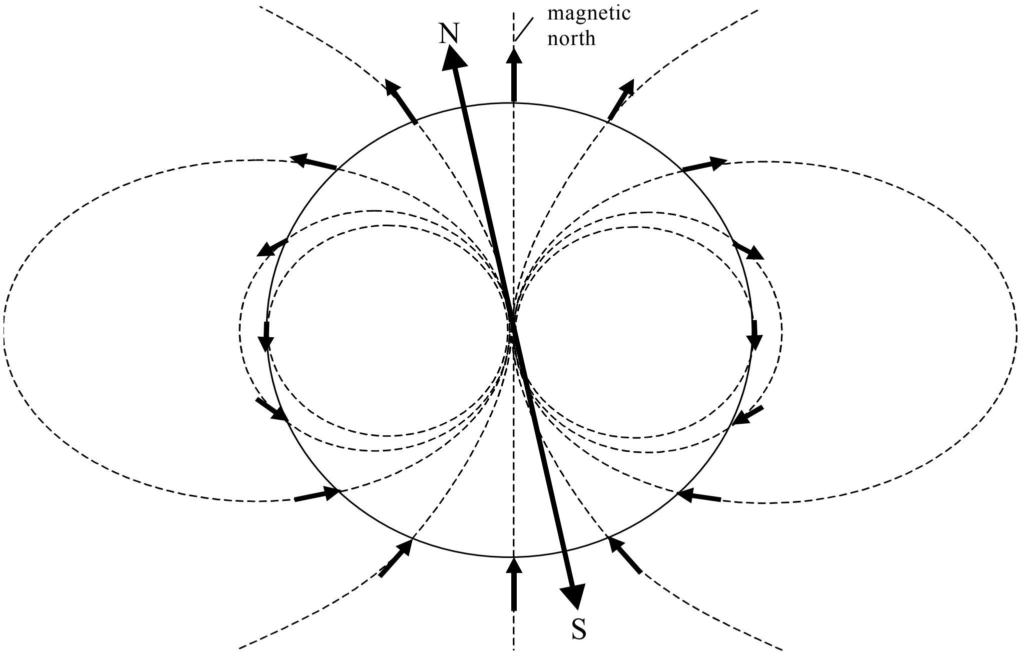

Figure 5: Earth's magnetic field at the surface. The field is horizontal at the magnetic equator and vertical with twice the magnitude at the magnetic poles, which are inclined with respect to the spin axis.

The strength of the earth's dipole source is such that the field is about 0.6 gauss vertically at the magnetic poles and about 0.3 gauss horizontally at the magnetic equator. This dipole is currently slanted about 11° with respect to the earth's spin axis, placing the magnetic north pole about 700 miles away from the geographic north pole at the top of North America, and moving about 15 miles west per year. Furthermore, local iron ore deposits significantly distort the ideal dipole field near the surface of the earth. The upshot of all this is that magnetic north deviates from true geographic north by an amount called the magnetic declination, D, which varies about +/- 20° across the United States for example (with anomalies as large as 60° in certain regions). To use a compass effectively for navigation, it is essential to use known latitude and longitude to look up the local value of D and correct the compass reading with it. Many GPS receivers have this capability built in. Even with access to a good map of local declinations, the accuracy of a compass can be limited by changing magnetic disturbances caused by solar winds, which can cause hourly deviations of +/- 0.3° on magnetically turbulent days.

An electronic compass must find the direction of the horizontal field, while disregarding the effect of the vertical field. Older style compasses do this by suspending a two-axis magnetometer from a mechanical gimbal so it hangs level. Then the magnetometers automatically sense only the horizontal components B_x and B_y, and the direction of the north vector relative to the instrument case is calculated as $\tan^{-1}\left(B_y/B_x\right)$. This is clumsy, and precludes calibration to compensate for metals in the instrument, since the sensors are not fixed with respect to the instrument chassis. Therefore, modern compasses, such as the ones in HMDs, sense the full magnetic field vector, then use pitch and roll information obtained by gravimetric and/or gyroscopic means to calculate the components of this vector in the horizontal and vertical directions. They then calculate heading using only the horizontal components in the above formula. Referring to Figure 5 and equations (1.3), it can be seen that the dip-angle or magnetic inclination, I, varies from 0° at the equator to 90° at the poles. The error in the just described compassing calculation due to a tilt measurement error ε is

$$\text{compass error} = \tan^{-1}\left(\sin(\varepsilon)\tan(\text{I})\right)$$

so in Burlington, Massachusetts, where I=69.27°, every 1° of tilt error will cause a compass yaw error of 2.64°. In Northern Europe or Canada, the inclination angle approaches 80°, with tilt error amplification factors of 4 to 5. This means that even a state-of-the-art magnetometer that has been calibrated to 0.1° may experience yaw errors of several degrees when used with a good gyro-stabilized inclinometer of 0.5° peak tilt error. When used with a plain inclinometer that experiences many degrees of slosh error whenever the person moves, the yaw reading will twist several times as much during the sloshing of the pitch and roll. This is the effect that made the simple inclinometer/compass orientation trackers in early consumer HMDs virtually unusable.

Clearly, geomagnetic sensing cannot be relied on as a primary yaw determination means in applications such as AR which demand absolute accuracy better than a couple of degrees. Even in less demanding applications, and if we assume the use of perfect magnetometers compensated by perfect pitch and roll data, there are some limitations that need to be considered in typical environments. I have observed in several buildings that if you walk down the hall in a straight line with a compass, the needle swings sometimes +/-20° or more as you pass by doorways and other architectural features. Likewise, seated in the office chairs in a variety of cubicles and offices with a compass on my head I found that the needle often rotates 3-8° while sliding the chair back a half meter or so, presumably due to the distortions caused by steel furniture or computer monitors. Outdoors, the readings were much more stable far away from buildings, but began to deflect a few degrees about 7 meters away from the building and over 5° within 3 meters of the building. Cars or signposts a couple meters away also disturbed readings by a few degrees. These are only anecdotal observations, but they do suggest a need for caution and perhaps sophisticated hybrid tracking algorithms when designing systems that make use of geomagnetic field sensing.

7.3.5.2 AC and DC Active Source Systems

A few years after Sutherland's early HMD-tracking experiments, Jack Kuipers of Polhemus Navigation Sciences invented a technique for tracking the position and orientation of a fighter pilot's helmet, using AC magnetic field coupling from a 3-axis source to a 3-axis sensor (Kuipers, 1975; Raab et al, 1979). More recently, similar systems using quasi-DC fields have been developed (Blood, 1989), and both technologies are now widely used in a broad range of human-machine interface applications.

In both systems, the magnetic fields are generated by a source consisting of three orthogonal coils of wire wound around a common core. The coils are activated in sequence by the electronics control unit to generate three orthogonal magnetic dipole fields similar in shape to the earth's dipole field illustrated in Figure 5 but on a much smaller scale. For the AC systems, the source is activated with oscillating currents of 7-14 kHz, and the sensor consists of a similar tri-axial coil assembly that is able to measure the components of these oscillating magnetic fields by inductive pickup.

For the DC systems, the triaxial sensor assembly uses devices sensitive to small DC magnetic fields, namely any of the sensor types that are used in electronic compasses. Traditionally the most popular such sensor was the fluxgate magnetometer, but recently solid-state technologies such as magneto-resistive, magneto-inductive, micro-mechanical and hall effect sensors have been replacing fluxgates. After each source coil is activated with a DC current, the system waits for eddy currents induced in any nearby metal objects to die out before measuring the resulting field. This way, the DC tracker is unaffected by any non-magnetic conductors which cause trouble for AC trackers due to the eddy currents.

As can be seen from Equation (1.3), the magnetic field magnitude $b \equiv \sqrt{B_r^{\ 2} + B_t^{\ 2}}$ falls off with the inverse cube of distance r from a dipole source. Naively, we might assume that this means that magnetic tracker resolution and accuracy will degrade according to the cube of the transmitter-receiver separation d_{tr}. Unfortunately, Nixon et al (1998) have shown analytically that at least position resolution must degrade as the fourth power of separation distance, and have confirmed experimentally that both position and orientation follow this trend for AC and DC magnetic trackers. The reason for the fourth power is that position resolution is not dependent on the magnitude of the magnetic field, but rather on the gradient of this magnitude with distance:

$$\Delta r = \frac{dr}{db}\Delta b \propto r^4 \Delta b \qquad (1.4)$$

This means a small disturbing field Δb will produce a position error component along the radial axis proportional to $d_{tr}{}^4$. It is not obvious from this analysis why the orientation errors should also grow proportional to $d_{tr}{}^4$, but the experimental data show that they do.

In addition to the dramatic effect of range on performance, which affects both AC and DC trackers the same way, there is a significant difference between the two in terms of sensitivity to external interference sources. The most common sources of interference are mains power wiring and appliances generating interference at 50 or 60 Hz, and computer monitors. Because the sensors in AC trackers only detect signals in a frequency band centered around typically 8,10,12 or 14 kHz, they are virtually immune to low-frequency mains interference. The tested AC tracker was able to operate at 0.2-0.25 mm resolution at a range of 600 mm in an ordinary room environment without special synchronization and filtering. The DC sensors are sensitive at low frequencies, and the tested system produced over 30 mm of position noise at the same range. To attenuate this down to a much more tolerable level, it was suggested that a DC tracker's sampling frequency should always be synchronized to twice the mains frequency, and then a filter is employed to average two adjacent samples, thus canceling the interference.

Nixon et al (1998) also analyzed the effects of metals on AC and DC trackers. Knowing that the eddy currents induced in any metal object (or magnetization induced in a ferromagnetic metal) are proportional to the applied field strength, they infer that the metal object will produce an unwanted source of strength proportional to $\frac{1}{d_{tm}{}^3}$ which will result in an interfering field at the receiver proportional to $\frac{1}{d_{tm}{}^3 d_{mr}{}^3}$, where d_{tm} and d_{mr} are the distance from transmitter to metal and metal to receiver respectively. Plugging this interfering field Δb into Equation (1.4) yields an error

$$\Delta r \propto \frac{d_{tr}{}^4}{d_{tm}{}^3 d_{mr}{}^3}. \qquad (1.5)$$

Although this model is simplistic, a variety of experiments in which the numerator was controlled independently of the denominator or vice versa produced data that fit the hypothetical model quite well. This equation leads to the obvious conclusion that the best countermeasure is to keep the transmitter close to the receiver and metals far away from both of them. Different types of metals were tried, and it was found that the DC tracker was completely unaffected by brass, aluminum and stainless steel, but committed larger errors than the AC tracker in the presence of copper, ferrite and mild steel. Presumably the AC tracker performed better with ferromagnetic steel because the magnetic permeability decreases with frequency. Additional data on the effects of metal size are included in the article.

The latency of magnetic tracking is limited only by the rate the system can cycle through three excitation states (plus a zero-excitation state in DC trackers to remove the effect of earth's magnetic field), and the need for noise reduction filtering. Adelstein et al (1996) measured the latencies of modern DC and AC magnetic trackers using a carefully designed mechanical testbed and data analysis procedure to isolate the internal latencies of the trackers. With all filtering disabled and tracking a single receiver, they report latencies of 7.5 and 8.5 ms respectively for position, and less for orientation. In an environment with 60 Hz power line noise, the DC tracker will normally need to be used with a two-tap averaging filter at 120 Hz, adding about 4 ms of additional latency. The main trade-off to consider is range v. resolution v. latency. If the resolution is just acceptable at a range r, then at 2r with no filtering there will be 16 times as much noise. To filter that noise back to the original level using a simple rectangular moving average filter would require 256 taps, thus adding a latency of 128 X the sampling period. More sophisticated filters can of course be designed to accomplish the same noise reduction with far fewer taps, but the example illustrates the general idea of the trade-offs involved.

7.3.6 Radio Waves, Microwaves, and Millimeter Waves

In the preceding two sections we have considered applications of the electric field and magnetic field as they were understood in pre-modern physics. During the nineteenth century, experiments by Oersted, Faraday and others uncovered the relationships between electricity and magnetism, which were unified by James Clerk Maxwell into a beautifully succinct framework of four equations which still express the essence of electromagnetic theory. From his equations, Maxwell predicted the existence of electromagnetic waves, later demonstrated using radio waves by Heinrich Hertz in 1888, and even developed a successful theory of light as electromagnetic waves. Indeed, the whole spectrum of electromagnetic radiation, from radio waves through microwaves, millimeter waves, infrared, visible light, ultraviolet, X-rays and Gamma-rays, all consist of nothing more than mutually induced electric and magnetic field fluctuations of various frequencies propagating through space according to Maxwell's Equations. The wavelengths vary over twenty orders of magnitude, and the differing wavelengths lead to different ways of interacting with matter. Thus, although all electromagnetic waves propagate through empty space in the same way, there are enormous differences in the equipment needed to create them, detect them, and in terms of how they penetrate or reflect off of various materials. For this reason, we divide the electromagnetic spectrum coarsely into a lower half, discussed in this section, and an upper half, discussed in the following section, which exhibit qualitative differences.

Radio and microwaves have so far not been much exploited in tracking human motion, but they are widely used in navigation systems including GPS, Glonass, LORAN, TACAN, Omega, and Transit (Getting, 1993), and various airport landing aids (e.g. VOR, DME, ILS) and radar systems. They have begun to find application in local positioning systems that find RF asset tags in warehouses or hospitals (Lanzl & Werb, 1998), and are likely to be used for human motion tracking systems in the future as the precision improves and the technology becomes smaller and cheaper. Electromagnetic wave based tracking techniques are capable of providing vastly greater range than quasi-static electromagnetic fields because radiated energy dissipates as $1/r^2$, while the dipole field strength gradient drops off proportional to $1/r^4$, as discussed in the previous section. Furthermore, radio waves suffer negligible absorption losses in air, and are virtually unaffected by wind and air temperature, so they are uncompromised outdoors or in large open spaces where acoustic systems are affected by attenuation, air movement and temperature gradients.

Table 3: Lower Electromagnetic Spectrum

Radio Waves (f < 1 GHz, λ>30 cm)						Microwaves (f>1 GHz)		
SLF/ ELF	Low Freq. (LF)	Med. Freq. (MF)	High Freq. (HF)	Very H. F. (VHF)	Ultra H. F. (UHF)	Super HF (SHF)	Millimeter Waves / Extra HF (EHF)	Submillimeter Waves (extreme-IR)
10 km 30 kHz	1 km 300 kHz	100 m 3 MHz	10 m 30 MHz	1 m 300 MHz	10 cm 3 GHz	1 cm 30 GHz	1 mm 300 GHz	

Table 3 above shows the names, wavelengths and frequencies associated with 8 decades of the electromagnetic spectrum. All electromagnetic waves originate from accelerating electric charges, and the macroscopic wavelengths in this portion of the spectrum can most efficiently be emitted or absorbed by free electrons oscillating in a metal antenna whose dimensions are a significant fraction of the wavelength, driven by electronic oscillator circuits. The extreme low frequencies, down to 100 Hz or less, are useful for communicating with submarines because they penetrate seawater significantly (skin depth of 10 m for 100 Hz ELF waves). They are not used for anything else because they require enormous antennas to achieve useful gain and have extremely narrow bandwidth. By 10 MHz the skin depth (depth at which the attenuation is 1/e) is down to only a few centimeters, and even pure water remains opaque to all higher electromagnetic frequencies with the exception of a narrow dip in the absorption curve at the frequency of visible light.[4] This implies that human bodies, made up largely of water, will block the propagation of higher frequencies that are more desirable for precision motion tracking. On the other hand, most non-conductive materials are transparent from DC all the way into the submillimeter band, conferring significant

[4] Perhaps this explains why vision evolved in the frequency band it did, since this is the only illumination available in the ocean.

line-of-sight advantages over optical trackers.

With the exception of Omega, which determines relative distance from base stations based on the relative phase of received low frequency radio signals, and direction-finding techniques like VOR, most radionavigation systems operate on the principle of time-of-flight (TOF) rangefinding, much as described for acoustic ranging in section 7.3.3. In the following subsections we will briefly describe TOF ranging, using GPS to illustrate, then discuss unique possibilities afforded by the millimeter waves at the very high end of the microwave spectrum, and Ultra-Wideband (UWB) technology which may someday offer higher resolution.

7.3.6.1 Time-of flight ranging

RF position tracking systems in general operate in much the same way as acoustic trackers, relying on the time delay for the propagation of a train of waves in order to measure the distance between a transmit and receive antenna. The waves travel about a million times faster (roughly 1 foot/nanosecond as opposed to 1 foot/millisecond for sound), making the task of measuring the time-of-flight with sufficient precision much more difficult. For example, ranging with 1 mm resolution in a single operation would require a timer/counter circuit that can count at 300 GHz, which would require expensive and power-consumptive electronics based on Gallium Arsenide or Indium Phosphide semiconductors.

One technique to eliminate the requirement for such high-speed timers is interferometry using a continuous wave source. The transmitted and received waves are combined and the phase difference between them can be measured to a small fraction of a wavelength. As with phase-based acoustic ranging, this approach suffers from an integer ambiguity problem: the distance between the transmitter and receiver can change by any integer multiple of the wavelength, and the phase difference will be the same. It is therefore necessary to know the initial position, and keep track of the number of phase roll-overs during the tracking, then add the integer number of wavelengths to the fractional wavelength determined by phase interferometry to compute the total distance. If the object moves too fast, gets temporarily blocked, or receives interference, there is a possibility of cycle slips leading to gross ranging errors.

Another approach to avoid the need for extremely high-speed digital counters is to use a combination analog/digital timer. A digital counter is run at a reasonable rate, say 1 GHz, and on each cycle it initiates an analog ramp signal. Reception of the pulse stops the ramp generator and the digital timer, and the stored voltage on the ramping capacitor is sampled and used to interpolate between the value where the counter stopped and the next count.

Another approach is to take advantage of the high speed of light to make ranging measurements at a very high repetition rate, and then average thousands of separate ranging measurements taken over a brief interval to produce a range measurement of higher resolution. Using conventional narrowband signals, only a limited number of separate bursts can be transmitted per second because each necessarily requires many cycles of the carrier to ring up and down again, but section 7.3.6.3 below describes technology for producing temporally shorter duration pulses.

A more sophisticated approach is the delay-locked loop (DLL) employed in GPS receivers to estimate the delay in the transmission of a complex spread-spectrum signal without having to measure the time to a specific single arrival event. A digital pseudo-random noise (PRN) code is modulated on the microwave carrier before transmission from the satellite. A replica of the code is played back in the receiver with a time delay τ which can be adjusted using a numerically-controlled oscillator, and multiplied by the incoming PRN code from the received signal. The resulting product is averaged over a period of time, yielding a value representative of the correlation between the incoming PRN code and the local copy. This correlation is maximized when the time shift τ causes the two signals to exactly line up in time. Once the correlation peak has been found and locked onto, the DLL uses a feedback loop to keep adjusting the delay τ to make sure they remain aligned. If the clock in the receiver were exactly synchronized with the clock in the satellite, the resulting time delay τ measured by the DLL would provide a direct measure of the time-of-flight from the satellite to the receiver. Since the receiver does not have an atomic clock, it is likely to have some clock bias Δt_c, and this must be estimated and added to τ to get the true range. In GPS, this is accomplished by measuring the "pseudoranges" from four satellites to solve for the four variables x, y, z and clock bias Δt_c. The GPS receiver exploits a very stable, but low frequency, numerically-controlled oscillator with a frequency of only 10.23 MHz, to lock onto the received waveform and determine its propagation delay to a small fraction of the period of the oscillator, corresponding to less than a meter. For greater precision, the receiver can also lock onto the phase of the carrier waveform to obtain a resolution of about 1% of the 20 cm carrier wavelength, or 2 mm.

However, there are multiple full-cycle phase shifts of the carrier waveform still consistent with the delay found by the code-tracking loop, so the integer ambiguity problem characteristic of continuous-wave interferometry exists here too, but only over a certain range of integers. An integer ambiguity search algorithm can be used to pick the carrier wave most consistent with the code-tracking loop, resulting in sub-centimeter GPS accuracy in surveying applications, but this technique cannot yet robustly track a dynamic object in real-time because cycle slips are likely to occur.

Multipath issues were discussed in Section 7.3.3 for an acoustic ranging system which stops a counter as soon as a received burst is detected. By detecting the received burst early, multipath energy arriving after the detection point is automatically rejected. Spread-spectrum radio receivers cannot use this simple strategy, since they must receive a substantial length of PRN code to form a strong correlation peak. Each chip of the P-code is a string of 154 cycles of the 1.575 GHz L_1 carrier frequency, so a 1023 chip sequence lasts 100 μs and spreads out over 30 km. It is therefore inevitable that many multipath reflections will begin to arrive long before a complete copy of one PRN code sequence can be processed by the DLL. Fortunately, the autocorrelation of the PRN code is essentially a unit impulse, so that copies of the code delayed by more than one chip have nearly zero correlation with the undelayed signal, and therefore don't disrupt the DLL's tracking of the delay in the direct received signal. However, reflection signals delayed by less than one chip do add to the direct path signal and shift the correlation peak, and therefore the measured signal propagation delay. The chip length of the P-code signal which is used for precise tracking after initial acquisition and lock are achieved is 154 cycles (one period of the 10.23 MHz master oscillator), or about 29 meters long. Therefore, any multipath signals that travel paths longer than the direct path by more than 29 meters can usually be rejected. This provides a reasonable measure of multipath rejection for an airplane in flight, since signals from overhead satellites that bounce off the ground will be delayed by well over 29 meters before they get to the plane. However, for short-range ranging between a "pseudolite" and receiver that are both in a building or near the ground, this provides almost no protection since thousands of the strongest reflection paths will be longer than the direct path by only a few meters or less. Abandoning the use of off-the-shelf GPS receiver electronics and developing a custom spread-spectrum ranging system for virtual environment applications would be unlikely to help very much. Using higher frequency microwaves and a shorter chip length might ultimately reduce the chip length to a few meters, but the analysis in Section 7.3.3 for acoustic systems suggests that in human-scale tracking applications we need to reject all signals that are delayed more than a couple centimeters.

7.3.6.2 Millimeter Waves

At the high frequency extreme of the microwave spectrum, with wavelengths from 1 cm down to 1 mm, lie the millimeter waves, which exhibit behavior halfway between radio waves and light. Although they were discovered in 1896, they have just recently become an active area of research, and practical applications in law enforcement are beginning to emerge (Williams, 1999). Most of the recent interest stems from the light-like capabilities of the waves. Because of the very short wavelengths it is actually possible to build an imaging sensor in a man-portable size by packing an array of tiny antennas into the focal plane behind a plastic lens. For a given desired resolution the minimum camera size is limited either by the size of the individual antenna elements or the diffraction limitation due to the lens diameter. Using typical antenna elements of 3 mm X 3 mm each, a 300-mm camera could capture 100 X 100 pixel images at video rates, or it can achieve higher resolution at slower rates by mechanically dithering the focal plane array. However, the angular resolution would be diffraction limited by the 300-mm lens aperture to about λ/D = 3 mm / 300 mm = 10 mrad, so a larger lens would be required to achieve greater than 100 X 100 resolution in a 60° field of view.

Clearly a much larger camera is required to achieve resolution comparable to an ordinary visible or IR video camera. Why then should anyone be interested in a millimeter-wave (MMW) imaging sensor? One reason is that humans glow quite brightly in MMW thermal images because of our higher body temperatures, while porous materials like clothing and wall-boards are quite transparent and dense solids like metals and ceramics are opaque. Thus passive MMW imaging can be used to detect concealed weapons, even non-metallic ones, beneath a person's clothing from a distance, or visualize the locations and activities of people in a room through the wall. Another advantage of the MMW sensor is the possibility of coherent detection and demodulation of the signals received at each antenna. This is being used to develop active MMW radar 3-D imagers that detect the distance, z, of objects in a scene as well as their horizontal and vertical locations. A linearly frequency swept chirp waveform is broadcast into the scene, and the reflected signals coming back from more distant objects will correspond to earlier parts of the chirp, or lower frequencies. By heterodyning the transmitted signal with the returned signal at each receiver antenna, a difference frequency representing range can be measured for each pixel, and other non-coherent background thermal radiation can be ignored. This type of radar offers much higher resolution than standard microwave radars, yet unlike infrared imagers it can penetrate even dense fog, smoke and dust, or even walls. Active MMW radar may therefore play an important role in firefighting and search-and-rescue.

Neither the passive nor active MMW imaging sensors are likely to offer enough cross-range resolution for precision HMD or tool tracking in virtual environments, but the millimeter waves could also potentially be used to advantage for simple point-to-point TOF ranging between omni-directional transmit and receive antennas. Compared to lower microwave frequencies used in GPS and other RF tracking systems, millimeter wave electronics can operate with much wider bandwidth, which could potentially be used to achieve higher resolution and tighter rejection of multipath interference. In fact, since the wavelengths are of the same size as those used in acoustic ranging, the simple early-detection-of-first-arriving-wavefront strategy could possibly be implemented with minor modifications in a MMW system. The electronics are currently much more expensive and complex than audio frequency electronics, and the FCC has not yet allocated any spectrum above 60 GHz, so the commercial deployment of such a solution is a long-term prospect.

7.3.6.3 Ultra-Wideband (UWB) Ranging

UWB ranging makes use of non-sinusoidal electromagnetic signals such as impulses. Since there is no carrier frequency these are sometimes called time-domain, carrierless or baseband signals. These signals have been studied since the 1970's and applied to radar (Taylor, 1995) as well as communications (Win & Scholtz, 1998). Interest has increased lately due to the development of simple low-power electronic circuits for generating and timing short impulses, including the famous Micro-Impulse Radar (MIR) from Lawrence Livermore Laboratory (McEwan, 1993) used in commercial applications such as studfinders and automobile warning radars.

Most UWB schemes use short pulses approximating impulse functions or doublets, such as a half-sine pulse or a Gaussian monocycle signal (derivative of a Gaussian function having a doublet-like shape with a positive excursion immediately followed by a negative one). By transmitting a sequence of such impulses with a random non-periodic distribution in time, the frequency spectrum of the signal is kept flat like white noise, and no appreciable interference is caused in narrowband radio receivers. Likewise, the UWB receiver tunes in a specific UWB transmission by knowing in advance the PRN code for the expected distribution of pulses in time, and is therefore relatively immune to interference from narrowband transmitters because it is only receptive during occasional narrow time windows. UWB cannot be allocated specific regions of the spectrum as with conventional RF systems because it necessarily emits energy across the whole spectrum from DC to several GHz. However it's emissions look like very low-level background noise and are therefore potentially inter-operable with conventional systems. The FCC is currently considering whether to allow commercial UWB deployment and how to regulate it. Narrowband systems accommodate multiple users in a given area by assigning each transmitter a different frequency band (frequency-division multiple access or FDMA), and spread-spectrum systems further allow multiple transmissions on the same frequency band by using different spreading codes (code-division multiple access or CDMA). UWB instead accommodates different transmitters because each is transmitting pulses following different pseudo-random time-hopping patterns, with a low probability of two pulses colliding because they are so short. By spreading each bit of information or ranging operation over many pulses, even occasional collisions are tolerable.

The outstanding advantage of the UWB paradigm is the improved ability to reject multipath signals. With pulses as short as 200 ps, all reflection paths delayed by 6 cm or more can be easily disregarded. For this simple reason, it is this author's opinion that if precise and robust electromagnetic ranging in indoor environments ever happens, it will be based on UWB impulses. Logically, this ought to be doable with much simpler electronics than that required to demodulate a complicated spread-spectrum signal. If this turns out to be true in practice, and if the FCC develops a policy that allows UWB transmissions without too many restrictions, then this may eventually become a preferable method of ranging in VE motion tracking systems.

7.3.7 Optical Tracking

Table 4: Upper Electromagnetic Spectrum

Infrared (IR)				Visible	Ultraviolet (UV)				X-rays			Gamma-rays		
Far-IR/ sub-mm waves		Mid-IR	Near IR		Near UV (UVA)	UVB	UVC	Extreme UV	x-rays		gamma-rays			
	50 µm	2.5 µm	780 nm	400 nm	315 nm	280 nm	100 nm	30 nm	4 nm	300 pm			0.3 pm	

Table 4 above shows the names and wavelengths associated with an additional 10 decades of the electromagnetic spectrum from about 10^{12} Hz to more than 10^{22} Hz . The lower 5 decades encompassing IR, Visible and UV are generally referred to as "light", while the upper 5 decades are made up of X-rays and Gamma-rays and generally referred to as "radiation". The light half is somewhat stretched out to show a variety of sub-bands having different properties. The mid-IR wavelengths are emitted and absorbed by matter through transitions in molecular vibration states, and thus can be perceived as a sensation of heat by the skin. In the near-IR it switches over to outer electron transitions, which are also responsible for the absorption and emission of visible and ultraviolet light. Silicon photodetectors actually have their peak sensitivity in the near IR, gradually tapering off in the blue end of the visible spectrum. Thus, common and inexpensive cameras based on CCD or CMOS sensing arrays are equally usable in the visible or near-IR. So-called thermal imaging cameras exist in the mid-IR, but they become progressively more expensive with increasing wavelengths, requiring thermal cooling and larger lens apertures to achieve good resolution. Electronic sensors exist in the UV range as well, but ordinary glass and plastic lenses block UV, so UV cameras require expensive quartz optics. For these reasons, all optical trackers to date have operated in the visible or near-IR, and there is no apparent reason to try more exotic techniques in the UV or long-wavelength IR. Active source optical trackers often use IR because it is less distracting and the cameras can filter out all the visible light interference created by the room lights. On the other hand passive vision-based trackers depend on the available scene illumination and usually operate in the visible range.

Beyond the UV lurk the X-rays, which originate from inner electron transitions, and the Gamma-rays, associated with nuclear transitions. (The distinctions between UV, X-rays and Gamma-rays are based on the type of transition that creates or absorbs them, thus the overlapping wavelength bands in Table 4). X-rays would seem to offer at least one potential advantage over the lower optical frequencies in that they can penetrate human flesh and many other common obstacles. There are now reasonably compact video cameras for X-rays using ordinary silicon CCD technology, but these are sensitive to lower-frequency X-rays which don't have much penetrating capability. The higher-frequency X-rays used in medicine are detected by phosphor cameras which use a sheet of phosphor to convert the image into visible light which can be digitized with a CCD array. There are no lenses for these types of X-rays, so the image must be formed as a shadow-gram on a sheet of phosphor larger than the object being imaged. These difficulties and the obvious health concerns probably rule out the use of X-rays for motion tracking purposes. Gamma rays are even more hazardous and their detectors more exotic, so the remainder of this discussion of optical tracking will assume the use of visible or near-IR light.

There are a particularly large number of different designs which use visible or infrared light in some way to track motion. Most of these use some form of bearing sensors (e.g. cameras, lateral-effect photodiodes or quad cells) to track point-like targets or beacons, as discussed in Section 7.3.7.1 below. Other optical techniques are discussed in Sections 7.3.7.2 and 7.3.7.3. Computer vision is an area of increasing interest as the cost of image processing comes down, and it might have been broken out as a separate section. However, most computer vision strategies involve identifying the 2-D coordinates of certain fiducials or landmark points in the scene, and are therefore included in the beacon-tracking discussion to facilitate comparison with the active beacon-tracking techniques. Additional vision techniques which don't rely on fiducial points are discussed in Chapter ? on gesture recognition.

7.3.7.1 <u>beacon-tracking</u>

Beacon-trackers can be classified into "outside-in" and "inside-out" systems. Outside-in beacon-tracking is the simplest and most common arrangement. Two or more cameras are mounted on the walls or ceiling looking in on the workspace. The sensors detect the direction to the targets or beacons attached to the object being tracked, and a computer then triangulates the 3-D positions of the beacons using the bearing angles from the two nearest cameras. The biggest problem with outside-in systems is a trade-off between resolution and working volume. If the sensors employ narrow field-of-view (FOV) lenses, the resolution is good, but the volume of intersection of the fields-of-view is small. With wide-angle lenses you can increase the working volume at the expense of resolution. For example, to cover a 16 X 16 X 8 ft. working volume inside a 20 X 20 X 10 ft room using four cameras mounted in the corners of the ceiling would require cameras with 78° horizontal and 76° vertical FOV. Assuming 1000 X 1000 pixel cameras and 0.1-pixel resolution for locating beacons, this would yield a resolution of about 0.7 mm. This is quite adequate positional resolution for many applications, but the orientation must be computed from the positions of three beacons mounted on a rigid triangle. A triangle of 15-cm per side would provide orientation resolution of about 0.4° which is too much jitter for some applications.

An alternative arrangement called inside-out optical tracking places the bearing sensors on the object being tracked, and the beacons at fixed locations on the ceiling or walls (Wang et al, 1990). This approach yields orientation resolution equivalent to the angular resolution of the bearing sensors, which is easily better than the requirements even using modest-resolution sensors. However, to achieve position resolution comparable to an outside-in system requires multiple sensors looking out in different directions, which can be too heavy for some applications. Conceivably, one could use outside-in tracking to achieve good position resolution combined with a single outward looking camera to provide good orientation resolution.

Beacon trackers may be further classified according to whether they use imaging or non-imaging sensors for detecting the bearing angles to targets. Imaging sensors such as CCD or CMOS cameras require some digital computation to find the locations of the targets in the image. They have the advantages that they can find the locations of multiple targets in a single image, and that the locations can be accurate even if there is background clutter, as long as the image processing is smart enough to distinguish the actual targets from the clutter. Non-imaging sensors such as quad cells (e.g. Kim et al, 1997) and lateral effect photo-diodes (LEPDs, e.g. Wang et al, 1990) are pure analog sensors that determine the centroid of all the light in the FOV. They require no digital image processing, but care must be taken to insure that the only light seen by the sensor at any given time is a single bright target. These sensors are therefore always used with active light source targets that can be switched on one-at-a-time. In most cases, the targets are infrared LEDs, and the sensor is equipped with an IR filter to block all the visible light clutter. A background subtraction between the result with the LED on and the result with it off can be used to further reduce the error caused by any IR sources other than the intended target. There is one type of error that even background subtraction cannot help: reflected light from the LED when it is turned on. For example, if the FOV of the sensor includes both an LED target on the ceiling and a portion of the wall, then some of the light from the LED will diffusely illuminate the wall and shift the centroid towards the reflection patch on the wall. To minimize this, the UNC inside-out optical tracker uses a cluster of outward looking LEPDs with only 6° FOV each (Welch et al, 99), so that when a particular sensor sights a target on the ceiling it is unlikely to also pick up a reflection on a wall. However, the system must use a very dense array of LEDs so that there will always be beacons available within the FOV of several of the sensors. Because outside-in systems require wide-FOV sensors to achieve good overlap volume, non-imaging 2-D sensors are not often used for these systems.

Another type of non-imaging sensor is the 1-D CCD array. These are often used with a cylindrical lens to measure a 1-D bearing angle with extremely high resolution. Arrays with 5000 pixels or more are readily available, and extraction of the center of the target distribution in one dimension requires only minimal digital signal processing. Because there are discrete pixels and no analog centroid processing, they are not subject to the same problems with clutter and reflections just described for quad cells and LEPDs. There are many commercial systems of quite similar form which have three 1-D bearing sensors mounted in a pre-assembled rigid bar of about a meter or more in length. In a typical case, the bar is mounted horizontally on the wall, and the LED targets are flashed one at a time. The two sensors at the ends of the bar are arranged horizontally and used for a simple 2-D triangulation to determine horizontal coordinates of the LED in the plane of the bar. The sensor in the center of the bar is arranged vertically and measures the elevation angle from the plane of the bar to the target, which is translated into the height of the target using the known horizontal location. Such systems provide high resolution and accuracy within a certain wedge-shaped volume in front of the tracking bar.

In contradistinction, imaging sensors can be used with active, retroreflective, or even passive targets. Many commercial videometric motion capture systems use cameras with a ring of LEDs around the lens to track balls coated with retroreflective film containing thousands of tiny corner-cube reflectors which return light back along the direction it came. Because the light source is so close to the camera lens, the balls reflect all the light toward the camera and appear very bright in the video image relative to normal objects which return only a small percentage of the illumination back towards the camera. The illumination scheme makes the targets so much brighter than the background that the only image processing required is to threshold the image and then find the centers of all the white circles. This technique can be used in both outside-in and inside-out tracking systems, but only works well indoors where the ambient illumination is not too high. Passive targets such as printed fiducial marks or natural scene features require considerably more image processing computation to track. Since they are no brighter or darker than other white or black objects in the scene, they must be identified on the basis of size, shape, and/or location using computer vision algorithms. The relentless pace of microprocessor development is making this method increasingly viable for cost-effective real-time tracking. Soon there will be CMOS cameras with enough onboard image-processing functionality to perform certain target extraction algorithms. The potential advantages of a vision-based tracking method using passive landmarks are compelling, especially for inside-out systems. Advantages over an electro-optical system using active targets include:

- no need to wire up the ceiling with an array of active LEDs
- range can therefore be expanded at much lower cost
- large numbers of users can share the same set of landmarks with no scheduling conflicts
- wearable system is untethered, without requiring radio telemetry
- can use wide-FOV cameras without errors due to reflection, and therefore far fewer landmarks

Advantages over a videometric system using retroreflective targets are:

- no need to carry an illumination source (and power for it) on the person being tracked
- targets are flat instead of spherical
- targets can be uniquely coded, and the image processing can identify the location AND identity of each
- can work indoors or outdoors
- with increasing computer vision sophistication, there is the potential to track natural scene features as targets, and thus enable tracking in an arbitrary unprepared environment of unlimited range

In light of these advantages, most recent research on tracking for AR has focused on vision-based tracking (e.g. Hoff et al, 1996, Koller et al, 1997, Mellor, 1995, Neumann & Cho, 1996). Since AR requires a self-contained wearable tracker that can operate over large areas with minimal preparation, inside-out vision-based tracking is a natural fit. So far, outward-looking vision alone has not yielded sufficient robustness, but hybrid techniques which combine inertial or magnetic tracking with vision are likely to succeed. For many VE applications the region of tracking is fairly defined, and the data is needed off-body to drive a graphics workstation. In these conditions, an outside-in approach may be more natural, if optical tracking is needed at all.

7.3.7.2 optical TOF ranging

There are also a variety of optical tracking techniques that don't entail finding the bearing angles from a sensor to certain target points. One such class of techniques involves optical ranging, in which the time of propagation of a light beam is used to measure the distance from a source to a detector much like the previously described acoustic and RF ranging methods. Both phase interferometry (of the carrier or of a slower modulation signal) and straightforward TOF counting for pulses exist. The most widely used optical ranging technique is lidar, in which distance along a laser beam to a reflecting target is measured based on round-trip TOF. The laser beam must be specifically pointed at the target. This is quite convenient for manual surveying applications with stationary targets. Automatic tracking systems for moving targets have been built by mounting the lidar on a servo-controlled pan-tilt mechanism programmed to continuously follow a given target once it has been locked on. The 3-DOF position of the target can then be directly read out in spherical polar coordinates using the current azimuth and elevation angles of the pan-tilt servo, and the radius measured by the lidar. Such a system is accurate but very expensive and can only track one target at a time. If the line of sight gets temporarily blocked, the system will lose lock, and reacquiring the target may require a time-consuming scan.

Ducharme et al (1998) have prototyped an omnidirectional point-to-point optical ranging system analogous to the acoustic and RF approaches discussed above. The light from a laser diode is fanned-out by a special lens to approximate a hemispherical point-source radiator. The laser diode is amplitude modulated by a 1-GHz sine wave, and a photodiode receiver within the illumination cone of the emitter produces a copy of this sine wave phase shifted by an amount proportional to the distance from the source. A digital phase processor circuit measures the phase difference between the transmitted and received signals, and keeps track of phase wrap-arounds which occur if the distance changes by more than 30 cm. Because of the omnidirectional emitter and receiver, the system will probably suffer from the same multipath issues discussed for similar acoustic and RF techniques, and the continuous-wave narrowband modulation scheme prevents the use of the multipath mitigation strategies described for those systems. However, under controlled laboratory conditions, the prototype exhibited peak ranging errors of +/- 0.2 mm over a range from 1 to 1.5 m separation. There are other point-to-point ranging techniques based on focus or intensity, but these are obscure and not very appropriate for VE motion tracking.

7.3.7.3 structured light

The techniques discussed so far used either no light sources or approximations of point-sources, possibly time-modulated. There are many technologies which generate spatially-modulated light fields such as lines, grids, or even more complex patterns, often scanned or otherwise time-varying. Most of these aim to recover the 3-D geometry of the scene, so this section will focus only on a few relatively simple examples which are concerned primarily with motion tracking.

The most common such technology is the laser scanner (Sorensen et al, 1989) which is now commercially available in a variety of different configurations. In all of them, a laser beam is fanned-out into a plane and then swept through the workspace by a spinning mirror. The time difference between the moment when the light plane crosses a reference detector mounted in the scanning mechanism and the moment it crosses a tracking detector in the workspace provides a measure of the bearing angle from the scanner head to the tracking detector. In a simplified configuration for illustrative purposes, there would be two scanners placed, say, in the front left and front right corners of a room. The two scanners would be synchronized so that a sequence of three non-overlapping scans would occur for each revolution of the motors: a horizontal sweep from the left scanner followed by a horizontal sweep from the right scanner followed by a vertical sweep from the right scanner. (Two sweeps in different directions from a single scanning motor can be accomplished with a multi-faceted mirror and/or multiple lasers.) The two horizontal bearing angles and the known baseline separation between the scanners are used to triangulate the horizontal coordinates of the detector, then the vertical bearing angle is used to calculate the height, just as with the linear CCD –based trackers. In fact, the system can be construed as a form of beacon-tracker in that it measures the bearing angle from one device, the scanner, to a point target, the detector. State-of-the-art laser scanners can achieve bearing angle measurement resolution on the order of 0.1 milliradians – quite comparable to state of the art camera/beacon systems. The major difference is that the sensor is located at the target rather than at the origin of the bearing angle. Mathematically, the configuration illustrated above is an outside-in system, but physically it is inside-out, with the data being measured and made available at a detector on the person. If the position data is needed on-board the moving object, as in robotic navigation, it can be computed autonomously on the robot using just the timing of the detection pulses without the need for any RF telemetry. Unlimited numbers of users can share the structured light fields without mutual interference. On the other hand, if the goal is to remotely track a moving user in a workspace, it may be more natural to use a camera-based outside-in tracker to avoid the need for data telemetry and computer circuitry on the person being tracked. For smaller volume applications such as cockpit head-tracking, there are also configurations which use a single scanner head sweeping in two directions together with a rigid assembly of 3 or 4 detectors to calculate 6-DOF pose.

A variation on the scanner theme has been recently proposed (Palovuori et al, 2000) which does not use any lasers, but rather an ordinary light bulb inside a rotating cylindrical shadow mask to produce the structured light field. The clear cylindrical drum is printed with a series of vertical black stripes whose varying widths and spacing form a pseudo-random bar code. A receiver contains matched-filter correlators for each of the shadow masks that are simultaneously rotating in the workspace, and thereby measures the code delay (proportional to bearing angle) from all the sources simultaneously and continuously through code-division multiple-access. The potential advantages are the continuous measurement capability and the use of arbitrarily bright light-sources without triggering concerns over eye-safety. It remains to be seen if the technique can produce resolution competitive with laser scanners.

Another class of structured-light devices uses projectors to paint patterns on the scene, and a camera viewing the patterns solves for its own pose if it knows the geometry of the surfaces on which the light is projected or is smart enough to recover the geometry from the distortions of the pattern. Livingston (1998) has proposed a very clever twist on this idea in which the scene geometry does not need to be known or recovered in order to track the camera. The concept is based on algorithms from computer vision (Longuet-Higgins, 1981) which can solve for the pose of a second camera relative to a first camera given a number of corresponding points in the two images, even though these points are at arbitrary and unknown locations in 3-D. He replaces the first camera with a projector which alternates complimentary patterns at a high speed so that the dynamic structured light is imperceptible to humans. The pattern contains points which are found by the camera and easily corresponded to points in the projected image because only one point is flashed at a time. Since the projector uses the same projective geometry as a camera, the computer vision algorithms may be directly applied to compute the pose of the camera relative to the projector. The primary application advanced for this technique is video-see-through AR, because the tracking can be done using the forward-looking cameras that are already part of a video-see-through HMD, without having to mount 3-D fiducials inside of the object being looked at. If that object is a human body, mounting fixed fiducials inside is impossible.

We know of no example in the literature, but a conceivable technique which could be classified as structured light is the use of polarized light. For a simple example, consider a person wearing a light bulb atop his helmet with a sheet of polarizing material over it. Above him on the ceiling is a photodetector with a sheet of polarizer spinning in front of it. The received intensity will oscillate with two peaks and two troughs per revolution of the motor, and the phase shift of this signal is a measure of the yaw direction of the person's head. A more elaborate scheme could be developed to measure multiple degrees of freedom.

7.4. The Mathematics of Motion Tracking

7.4.1 Observables and Pose Recovery Algorithms

None of the individual sensing technologies described in the previous section directly measures position and orientation of a moving body. Instead, each sensor measures certain **observables**, for example the range between two points, which are functions of the desired position and/or orientation. Acoustic, RF, and certain optical methods measure **range** from one transducer to another. Most optical technologies, and some radar and sonar methods, measure **bearing angles** from a sensor to a target, while goniometers measure **joint angles** between connected rigid bodies. Gravimetric and geomagnetic sensors measure **homogeneous field components**, and active magnetic sensors measure **dipole field components**. These are most of the absolute measurement types in common use. There are also a variety of relative measurements used for dead reckoning: **angular rates** from gyros or optical flow sensors, **linear acceleration** from accelerometers, **range rate** from doppler sensors, or **speed** and **direction** from optical flow (or wind or water sensors). There are other observation models possible, such as the **pseudorange** used in GPS, **range difference** measured by RF time-difference-of-arrival, **bearing difference** and the **accumulated delta-range** in GPS and the phase-coherent acoustic approach, but most of them are essentially variations on observable types already listed.

Since a single measurement of any of these types does not in itself reveal the position and orientation (collectively called pose) of the moving object, calculations, generally called pose recovery algorithms, are used to solve for the pose from several measurements. Some classic pose recovery algorithms are **trilateration** to solve for the position of a point when range measurements are available from that point to 3 known points, and **triangulation**, in which bearing angles from two cameras of known pose are used to solve for the position of an unknown target. Although trilateration is just an application of the familiar Pythagorean theorem, it is surprisingly difficult to come up with an efficient, exact and general algebraic solution in three dimensions, and this remains an active area of research (e.g. Manolakis, 1996). Classical triangulation, where the bearing angles are measured from multiple known observation points to an unknown target, is a simple matter of intersecting rays or planes. However, there are some pose estimation problems in computer vision which are closely related, yet far more difficult. The most important of these is perhaps the perspective-n-point-problem (PnP) in which a camera observes n 3-D points and finds their corresponding 2-D projections on the image plane (basically bearing angles from the camera to the points). If the 3-D points are known and the goal is to determine camera pose, the problem is historically called the "exterior orientation problem" or "space resection problem", but more simply should be called n-point camera pose estimation. The problem was first solved with 3 points in 1851 and has been solved many different ways since, which are reviewed and compared in (Harelick et al, 1994). While direct 3-point algorithms exist, they produce 4 solutions, and a unique solution requires a minimum of 4 points. Until recently such multi-point overdetermined solutions were iterative or nonlinear (eg. Longuet-Higgins, 1981). However, Quan and Lan (1999) just introduced a direct linear solution using only 5 points, and a two-step linear solution for 4 points, which also have fewer restrictions on the geometry of the points (e.g. the points may all be co-planar).

Clearly closed-form algebraic pose recovery algorithms are very complex, and the state of the art is still advancing. The alternative is to make an initial guess of the pose, linearize the nonlinear measurement equations about this guess, and then solve a least squares problem to determine corrections to the presumed pose that would minimize the weighted square errors of all the available measurements. This procedure is then iterated until it converges to a stable pose estimate. Foy (1975) argues that it almost always converges, and has a many attractive features:

- Multiple independent measurements are averaged naturally.
- Multiple measurements of different types are combined properly, i.e. with the correct geometric factors, and can be weighted according to their a priori accuracies.
- The statistical spread of the solution can be found easily and naturally.
- It usually converges even if the initial guess is quite far off, and failure to converge is easy to detect.

For unusual combinations of observables for which closed-form pose recovery algorithms are not already worked out in the literature, by far the easiest approach is to develop an iterative linearization-based algorithm.

7.4.2 Recursive Estimation for Tracking Moving Objects (Kalman Filtering)

All of the pose recovery algorithms discussed in the previous section are designed to solve for the pose of an object at time t given a set of measurements that are functions of that pose at time t. In other words, they assume either that the object is not moving, or that a complete set of measurements sufficient to determine the pose can be made simultaneously at time t. Even in the latter case, the traditional pose recovery algorithms are not optimal, because they re-estimate the entire pose from scratch at every frame, throwing away any information in its past history.

In a seminal paper, Rudolph Kalman (1960) combined a recursive least squares formulation with a state-space system dynamics model to develop a practical algorithm for computers to estimate the state of a dynamical system (e.g. pose of a moving object) by optimally combining past history, new measurements, and *a priori* models and information. Assuming that the system is linear, and the sensor and process noise are white and Gaussian, he proved that it is the unique best estimator by any reasonable criterion of optimality. Even without the Gaussian noise assumption, the Kalman filter is the best (in a least squares sense) linear estimator. Furthermore, in actual usage it has turned out to be robust in spite of modeling errors and violated assumptions, and can even be applied to systems with nonlinear dynamics and nonlinear measurement models through a linearization method called the Extended Kalman Filter (EKF). For all these reasons, Kalman filtering has become the foundation of modern multi-sensor data fusion and estimation.

We will not derive the Kalman filter here – the reader is referred to the many excellent textbooks on the subject (e.g. Gelb, 1974; Brown & Huang, 1992; Bar-Shalom & Li, 1993) – but merely introduce its form and how it is applied to motion tracking. The discrete Kalman filter assumes that the vector of states being estimated, $\mathbf{x}_k$, evolves according to a state propagation equation or **dynamics model**:

$$\mathbf{x}_{k+1} = \mathbf{\Phi}_k \mathbf{x}_k + \mathbf{w}_k \qquad (1.6)$$

where $\mathbf{\Phi}_\kappa$ is the state transition matrix from t_k to t_{k+1}, and that measurements $\mathbf{z}_k$ are related to the states by a linear **measurement model**:

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \qquad (1.7)$$

where $\mathbf{w}_k$ and $\mathbf{v}_k$ represent process and measurement noise vectors with covariance matrices $E\left[\mathbf{w}_k \mathbf{w}_k^T\right] = \mathbf{Q}_k$ and $E\left[\mathbf{v}_k \mathbf{v}_k^T\right] = \mathbf{R}_k$ respectively. Boldface lowercase and uppercase letters denote vectors and matrices respectively. The states in a motion tracking application actually evolve continuously, but we're only interested in them at certain times t_k, so all the evolution between such times is pre-integrated and rolled up in the state transition matrix $\mathbf{\Phi}_\kappa$.

The Kalman filter's main job is to make an estimate of the state, denoted $\hat{\mathbf{x}}_k$, which tracks the true state as closely as possible, i.e. with minimum estimation error $\tilde{\mathbf{x}}_k \equiv \hat{\mathbf{x}}_k - \mathbf{x}_k$. In order to do so, it keeps track not only of its current estimate $\hat{\mathbf{x}}_k$, but also the statistical uncertainty of that estimate as captured in the covariance of the estimation error $\mathbf{P}_k \equiv E\left[\tilde{\mathbf{x}}_k \tilde{\mathbf{x}}_k^{\,T}\right]$. Between measurements, the Kalman filter updates its estimate of the state using the dynamic model in (1.6):

$$\hat{\mathbf{x}}_{k+1} = \mathbf{\Phi}_k \hat{\mathbf{x}}_k \tag{1.8}$$

and every time it does so, it also updates the corresponding uncertainty in this estimate using:

$$\mathbf{P}_{k+1} = \mathbf{\Phi}_k \mathbf{P}_k \mathbf{\Phi}_k^{\,T} + \mathbf{Q}_k \tag{1.9}$$

Together, these constitute the **prediction** or **time update** step of the filter. Equation (1.8) says that without any outside information, your best guess of the next state is simply to propagate the current estimate forward in time using the known system dynamics. Equation (1.9) shows that when you make this prediction, the uncertainty of your estimate grows a little due to the process noise. It is derived simply by taking the covariance of both sides of

$$\tilde{\mathbf{x}}_{k+1} = \mathbf{\Phi}_k \tilde{\mathbf{x}}_k - \mathbf{w}_k ,$$

which follows from (1.8) and (1.6), and using the fact that $\mathbf{w}_k$ is uncorrelated with $\tilde{\mathbf{x}}_k$ because it is white noise.

When a measurement occurs, a correction is made to the state estimate:

$$\hat{\mathbf{x}}_k(+) = \hat{\mathbf{x}}_k(-) + \mathbf{K}_k\left(\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k(-)\right) \tag{1.10}$$

and the estimation covariance is reduced because the new information has reduced the uncertainty:

$$\mathbf{P}_k(+) = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k(-) \tag{1.11}$$

where the – and + in parentheses indicate the situation before and after this **correction** or **measurement update** step of the Kalman filter algorithm. The form of (1.10) is intuitively reasonable; the size of the correction to the state is proportional to the discrepancy between the actual measurement, $\mathbf{z}_k$ and the predicted measurement, $\mathbf{H}_k \hat{\mathbf{x}}_k$, based on (1.7). The constant of proportionality is the Kalman gain matrix $\mathbf{K}_k$, which is calculated by the formula:

$$\mathbf{K}_k = \mathbf{P}_k(-)\mathbf{H}_k^{\,T}(\mathbf{H}_k \mathbf{P}_k(-)\mathbf{H}_k^{\,T} + \mathbf{R}_k)^{-1} \tag{1.12}$$

We have skipped the derivation, but magically this formula provides the unique gain matrix that will cause the filter to track the state with minimum error. Other matrices, called suboptimal gain matrices, may still produce unbiased, non-divergent estimates of the state, but they will have greater mean-squared error.

There are a number of remarkable advantages you will notice just in the form of the filter. First off, the Kalman gain is computed fresh each time you wish to incorporate a new measurement. This makes it very easy to track systems with time-varying dynamics or measurement processes. For adaptive filtering, one can adjust the measurement noise covariance, $\mathbf{R}_k$, for each measurement to weight the measurement more or less heavily depending on the distance to the target, signal strength, or any other indicator of the probable quality of that measurement. This ability to handle time varying models is also the key to using the Kalman filter with nonlinear systems or nonlinear measurement models (which includes all of the observable types listed in Section 7.4.1). Linearizing about the current state estimate produces linear equations in the form of (1.6) and (1.7) for the residual errors, with $\mathbf{\Phi}_k$ and $\mathbf{H}_k$ matrices which are funtions of the current states. One then simply runs a time-varying standard Kalman filter on the error model, recomputing $\mathbf{\Phi}_k$ and $\mathbf{H}_k$ at each step based on the current pose. This is the basis of the Extended Kalman Filter (EKF). In applications with extremely nonlinear models, the EKF is sometimes unable to converge, and this has led to much research on advanced nonlinear filtering techniques. Some of these are extensions of the EKF paradigm such as iterating the measurement update, or using higher than second order statistics (the EKF uses the mean and covariance – just the first and second moments of the probability distribution – which is a complete characterization of a Gaussian distribution but higher moments are required if the distribution is non-Gaussian). More radical approaches dispense with the EKF completely. However, the types of observables listed above can be linearized well over a reasonable range, and normally one or two iterations of a standard EKF update produces unbiased estimation unless the measurements are extremely noisy or sparse, in which case high-resolution tracking will not be available by any means.

Another advantage is that the filter is very efficient with computer memory requirements. Everything it needs to know about the initial conditions and all the past measurements and motion is contained in the covariance matrix $\mathbf{P}_k$. The fact that the filter keeps track of the covariance as well as the estimate itself is also very useful in some applications. For example, in a GPS-enabled cell-phone, this could help E911 rescue workers decide how large an area to search for the caller. In an AR application it could be used to provide visual feedback to the user of potential position errors of a superimposed graphic in the scene, so that if the graphic may be pointing to the wrong part he'll know to check before removing that part.

An outstanding benefit of the Kalman filter is that it is very flexible about timing and the order in which it receives and processes measurements. There is no requirement for periodic time updates or measurement updates. In a typical run-time implementation, the KF program will have a main loop that processes time updates at a high update rate, and slightly interrupts this cycle to squeeze in a measurement update whenever one becomes available from a sensor. This enables flexible integration of data from disparate sensors that are not even synchronized. For example, an aided inertial navigation system may have signals from various navigation aids such as GPS, LORAN, ILS, altimeter, radar, star-tracker, compass, sonar, etc., arriving at different times. Each fix may have a different measurement model (1-D range measurement, 2-D bearing angle, 3-D GPS position fix, etc.) and is processed when and if it is available, calculating the appropriate $\mathbf{H}_k$ and $\mathbf{R}_k$ on the fly. A partial correction to the state is made immediately, conferring the benefit of all new information in that measurement. This approach fits particularly naturally with inertial navigation systems, which by their very nature consist of a high rate inertial integration process which drifts and gets updated (in the old days by hand) whenever a star fix or landmark sighting is made. This may be why inertial navigation engineers where among the first to adopt Kalman filtering in the mid-1960's. However, the asynchronous updating capability is valuable in other applications too. For example in ground-based multisensor tracking, bearings from two separated scanning sensors (e.g. radar or infrared) with unsynchronized scan rates are combined (Bar-Shalom & Li, 1995). Relying even more heavily on the incremental partial updating ability of the Kalman filter is a problem called bearings-only tracking, in which infrared or sonar bearing-angle measurements taken at different points along the trajectory of a moving vehicle are fused over time to yield the location of the target (e.g. Nardone, 1980). In active vision systems, a robot-mounted camera looks around the room, and whenever it sees a feature it recognizes, it updates its estimate of its own location and that of the feature (e.g. Harris, 1992; Chenavier, 1992).

Welch and Bishop (1997) coined the term SCAAT (Single-Constraint-At-A-Time tracking) to refer to this type of updating, and used it to dramatically improve the update rate and accuracy of the UNC optical ceiling tracker, which had previously used a batch pose recovery algorithm. They argue insightfully that this feature is particularly valuable in virtual environment systems, where high update rates and low latency are essential, and could beneficially be applied to magnetic and acoustic trackers as well. In fact, the InterSense IS-900 (Foxlin et al, 1998) takes advantage of this capability to eliminate the propagation latency of acoustic tracking. In most acoustic trackers, an emitter on the tracked object is fired, and the position is calculated once the spherical wavefront has reached all 3 receivers. If the object moved during the flight time, the range measurements don't reflect that. In the IS-900, the emitters are fixed in the environment, so when the wavefront arrives at the microphone on the moving object it registers a range measurement that is accurate at the instant of arrival, even if the receiver is moving. The EKF processes this individual range measurement, then continues to track the motion using inertial sensors until the next beacon's pulse is received.

7.4.3 Sensor Fusion Hybrid Tracking Design and Covariance Analysis

In designing or selecting a motion tracking system for an application, a central issue is often predicting the resolution and accuracy that will be achieved. Except in the case of "sourceless" trackers based entirely on gravimetric, geomagnetic and/or inertial sensing principles, the performance will depend on the tracked object's geometric relationship to the source(s) or landmarks. This dependence on location is often expressed for pure ranging-based systems using the concept of Geometric Dilution of Precision (GDOP). The GDOP is a unitless ratio which gives the position error per unit ranging error at any particular x, y, z location:

$$GDOP(x,y,z) = \frac{\sqrt{{\sigma_x}^2 + {\sigma_y}^2 + {\sigma_z}^2}}{\sigma_r} \qquad (1.13)$$

where σ_x, σ_y and σ_z represent the r.m.s. position uncertainties that result from equal ranging uncertainties σ_r to all the reference stations. For trilateration systems the GDOP will be 1 at the point where the three range measurement lines are orthogonal, and will increase rapidly as the object moves away from this point. Thus the GDOP provides a convenient map of the static resolution (accuracy) of a position tracker as a function of the basic range-finding resolution (accuracy). This can be used to decide how to space the landmarks and what quality of range measurements are needed. For systems which track 6-DOF, or use multiple types of measurements, one cannot express the sensitivity as a single number, but must specify the position and orientation performance as a function of each measurement error source, at each location. This is called error budgeting or sensitivity analysis. For dynamic motion, there will be additional errors that are not captured in such an analysis. These errors will depend on the nature of the motion, so to study dynamic accuracy, one must specify a trajectory and then use a Kalman filter simulation to study the covariance of position and orientation errors over time. This is what was done, for example, in the inertial system error growth analysis in Figure 4.

This covariance analysis capability is one of the major uses for Kalman filtering theory. By running a KF simulation once, one obtains the statistical distribution of errors that would occur if the same experiment were repeated numerous times. This is much faster than running a large set of Monte Carlo simulation runs and then computing the standard deviation of the errors at every timestep. Covariance analysis is very useful for designing hybrid tracking systems which fuse together measurements from different types of sensors. Many of the different types of sensing principles discussed in Section 7.3 have very different and sometimes complementary properties. Covariance analysis can help determine which technologies have the greatest synergy in an application, and can help optimize the trade-offs involved in the specification of hardware subsystems' performance and number and layout of sensors or targets.

The Kalman filter prescribes the optimal algorithm to fuse a given sequence of measurements, but in of itself it doesn't solve the problem of optimally scheduling the measurements or selecting which ones to use. In real world sensor systems, it is often necessary to assign sensors to certain targets or regions of sensing, or to schedule a measurement by initiating a scan or pulsing an emmitter, and there may be limited hardware resources or feasible repetition rates so that choices have to be made. In the motion tracker examples cited in the previous section, there would be the question of which LED to flash or which acoustic beacon to trigger next, to get the most information out of the Kalman measurement update. These two systems have used fairly simple ad hoc algorithms, but there exists an extensive literature on the subject of sensor management and scheduling (e.g. Nash, 1977; Manyika & Durrant-Whyte, 1992; McIntyre & Hintz, 1998) which contains some concepts of value to VE hybrid motion tracking systems. Many of these algorithms are computationally expensive, and the covariance analysis technique can be used to evaluate the benefit relative to simpler scheduling algorithms.

7.4.4 Auto-Mapping

In trackers built for short-range tracking, all the necessary sensors are usually built into one pre-assembled reference unit, which serves as the coordinate frame reference for the tracking data. In a magnetic tracker this is the source coil assembly, in an acoustic tracker it is often a plastic triangle with three microphones built in at the corners, and in many optical trackers it consists of a bar with linear array cameras mounted at the center and each end. The relative poses of the sensors in the housing are calibrated at the factory, and all tracking data is reported relative to the housing reference frame.

For larger area tracking, a single sensor reference unit cannot provide enough coverage area due to range limitations and high GDOP away from its "sweet spot". Therefore, a cellular strategy is often adopted using multiple sensor units. In optical motion capture, for example, a ring of cameras may be mounted around the perimeter of the room on high tripods, looking in towards the center of the room. After setting up the tripods, the user must perform a procedure to map out the location of all the cameras in a common world coordinate frame. This is usually done by placing a calibration structure in the center of the room where all the cameras can see it simultaneously.

What if the tracking area is so large that there is not one spot which can be seen by all the sensors, or the tracker uses an inside-out arrangement so that the head-mounted sensor(s) can only see a subset of the landmarks at once? This is almost always the case with AR applications, which use a wearable computer and allow the user to walk around freely interacting with the real environment. In this case a method is needed to auto-map out the entire array of landmarks or sensors by moving a calibration unit through the environment in advance. If the intended tracking environment is essentially unbounded, as in an outdoor AR application, even this is impossible and it is necessary to initialize and auto-map the landmarks on-the-fly during tracking. This is what we shall mean by auto-mapping from here on.

This problem has received a great deal of attention in the computer vision and mobile robotics navigation communities. In computer vision, the problem is referred to as Structure from Motion (SfM), and the primary goal is to reconstruct the geometry of a scene from moving camera imagery. The camera motion trajectory must be determined in the process, but that is not the main goal. There are several approaches to the problem, some involving deducing the shapes of surfaces from optical flow, and others involving auto-mapping the 3-D positions of a set of consistently observable feature points, and then connecting the dots. The modern EKF approach was introduced to the latter school by Broida et al (1990) who used 2-D bearing measurements over a sequence of frames to recursively estimate the camera pose and the 3-D locations of the observed feature points. Azarbayejani and Pentland (1995) refined the approach by simultaneously estimating focal length and changing the representation of the point positions into a 1-D format (depth along the ray of initial sighting) which allows for stable performance even if the initial positions of the features are not known.

The mobile robotic navigation community calls the problem Simultaneous Localization and Mapping (SLAM) and considers a wider variety of sensors including sonar, radar, and lidar as well as vision. Unlike SfM, the primary goal of SLAM is often to know the location of the robot, and developing a map of the environment is a necessary means to that end. The robotics community has been working on the problem even longer, and an EKF approach was introduced by Smith et al (1987) with similar structure. Like the early SfM papers, the early SLAM implementations employed a full-covariance Kalman filter. The position states of all N features being mapped were simply appended to the n_v states for the vehicle to produce a giant augmented state vector of length (n_v+ 3N). The covariance matrix of this entire augmented state vector is maintained, including all the cross-correlations between the vehicle states and each of the landmark positions, as well as between each landmark and every other. For optimal estimation according to Kalman filtering theory, one must maintain this full covariance matrix including the correlations between all the landmarks, and propagate it by the standard EKF formulae. Unfortunately, this rapidly becomes impractical for large numbers of landmarks, since the computational complexity (and numerical sensitivity) increase with N^2. A naïve solution is to drop the cross-correlations between the landmark position estimates, so that for each landmark one need only maintain the covariance of that landmark's own position estimate and the cross-covariance with the vehicle pose estimate. This reduces the problem to order N, but in practice the estimates will eventually diverge. The reason is that when each new measurement is made to a new landmark, the filter will think it contains new information (i.e. uncorrelated with the measurement to other nearby landmarks) and will make a correction to the vehicle pose accompanied by a dramatic drop in the vehicle pose covariance. Soon the vehicle will think it has been given a lot of independent information, and knows precisely where it is. The reality is that the position of landmark n was determined by a measurement made by the vehicle, whose position was determined in part by a measurement previously made to beacon n-1. If there was an error in the position of beacon n-1, some of that error was propagated to the vehicle pose estimate, which in turn propagated some of it to beacon n, so the positions of both beacons and the vehicle are all correlated and likely off in the same direction. The relative positions of the three may be known with little error, but the whole group has a large correlated error. If the correlations are dropped from the covariance matrix, repeated measurements to different beacons will be treated as independent evidence, and the filter will soon believe it knows exactly where everything is. The filter becomes "conceited" and then diverges. Much current research focuses on trying to find a work-around to this problem. Some approaches use local submaps, each of which maintains cross-correlations, and then attempt to stitch them together in a globally optimum way (Leonard, 2000) or build a relative map storing only inter-landmark distances so that repeated measurements are uncorrelated (Csorba et al, 1997). Unfortunately, the latter approach does not even estimate the pose of the vehicle. Another very interesting approach of very general applicability is a new method of data fusion called Covariance Intersection (Julier & Uhlmann, 1997). Covariance Intersection (CI) replaces the entire Kalman measurement update process in Equations (1.10) and (1.11) with something different in form. Recall from Kalman's proof that this means it must be suboptimal compared to the Kalman filter, but it is guaranteed to provide consistent results (covariance that is not unrealistically optimistic) even if there exist large and unknown correlations between the current measurement and previous ones. Using CI measurement updating, it is perfectly safe to ignore all the correlations between landmark position estimates, and the size of the problem is reduced to order N. This is one of the few techniques available so far for handling very large scale auto-mapping problems in a very general mathematical framework.

7.4.5 Prediction

A good motion tracker should have the ability not just to accurately follow the pose of the user at the present time, but also to predict motion enough to compensate for the end-to-end delay from user motion to visual feedback. This delay is caused by tracker latency, communications, rendering and image scan-out on the display, and depending on the number of pipelined rendering stages and frame rate it typically ranges from 25 ms to 150 ms or more. Obviously, the further you try to predict, the less accurate it is, so prediction is not a panacea for slow virtual environment generators. However, when the total latency is less than about 60-80 ms, it can help dramatically.

All prediction is based on the Taylor Series expansion:

$$x(t+T_p) = x(t) + \dot{x}(t)\cdot T_p + \frac{1}{2}\ddot{x}(t)\cdot {T_p}^2 + \frac{1}{6}\dddot{x}(t)\cdot {T_p}^3 + \ldots$$

The more derivatives at time t are known accurately, the further into the future this prediction holds. In model-free prediction (most of the work to date) this formula is applied separately to each of the 6 degrees of freedom, as if they each could evolve independently. This is the most general approach but doesn't take advantage of any kinematic constraints that might be known to exist, such as attachment of a head to a neck.

Normally one only has samples of the position and orientation variables at discrete points in time and must obtain the derivatives by numerically differentiating them, so the higher derivatives become too noisy to use. Early efforts to use prediction with magnetic trackers used a Kalman filter to estimate the velocity (Liang et al, 1991) or velocity and acceleration (Friedman et al, 1992). This is much less noisy then simple numerical differentiation because it performs optimal smoothing based on an *a priori* model of head motion "jumpiness" specified in the **Q** matrix of the dynamics model (1.6). Nonetheless, it is much more accurate to actually measure the velocities or accelerations using inertial sensors. Azuma (1994) showed 2-5 times higher accuracy with inertial sensors measuring the linear accelerations and angular velocities compared to a prediction method where these quantities had to be estimated from the optical position tracker data using a Kalman filter.

Whether using estimation or inertial sensing to obtain the derivatives, a model-based approach would allow longer-term prediction with the same accuracy from the same data. To see this, consider a hand which is constrained to remain attached to the end of a forearm, which is rotating about the elbow joint. To first order, a model-free prediction would predict the future location along a straight line projection of the current velocity of the hand, while a model-based method would predict the future location along the circular arc originating from the elbow, with radius based on the measured forearm length. If predicting far enough into the future that the elbow would significantly rotate, the latter would be far more accurate. Akatsuka and Bekey (1998) proposed an extremely simplified model of head motion (a lollipop on a stick of length L, rotating about a fixed point at the base of the neck), and showed improved prediction compared to a model-free approach, although the paper doesn't explain how the model is used in the predictor. For human motion prediction using a fairly simple model of 17 rigid objects, a model-free prediction would have 17 X 6 = 102 separate variables, whereas a model based approach would reduce this to predicting approximately 46 joint angles. By respecting these kinematic constraints, much accuracy would be gained, but even more might be attainable by considering a dynamics model of the body which includes mass distributions and muscle forces. Hans Weber at UNC has been working primarily on the former, and the Advanced Displays and Spatial Perception Laboratory at NASA Ames has an interest in the latter, but neither has yet published results (personal communications, June 5, 2000).

In addition to improving the kinematics models underlying prediction, there may be some benefits achievable with adaptive multiple-model filtering as well. In a Kalman filter, the state of the system is always modeled as a linear dynamic process driven by white noise, as expressed in Equation (1.6). The white noise is used to model the unknown inputs, namely muscle forces (when no inertial sensors are present) and derivatives of these forces when inertial sensors have directly measured the effects of the forces. This isn't a very good model of human behavior, as it implies Brownian motion going on all the time with the same driving intensity. Real human motion is much more episodic, having periods of stillness interspersed with bursts of motion activity. The onset of the motion is unpredictable, but once the motion starts, it is likely to follow a certain course.

Filters for tracking aircraft from radar observations make use of multiple-model techniques in which they use a second order kinematics model driven by white noise acceleration during straight and level flight:

$$\begin{bmatrix} r \\ v \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} r \\ v \end{bmatrix}_{k} + \begin{bmatrix} 0 \\ w \end{bmatrix}$$

and switch to a third order model driven by white noise jerk during maneuvers which involve acceleration:

$$\begin{bmatrix} r \\ v \\ a \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & \Delta t & 0.5\Delta t^2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r \\ v \\ a \end{bmatrix}_{k} + \begin{bmatrix} 0 \\ 0 \\ w \end{bmatrix}$$

There are a variety of statistically motivated techniques for detecting the onset of maneuvers and switching models or adjusting the blend of multiple concurrently running models (Bar-Shalom & Li, 1993). In addition, it would seem that model selection could be viewed as a classification problem, and traditional classification algorithms such as neural nets or fuzzy logic could be usefully combined with the KF estimation paradigm, although this has not been discussed in the literature. Like aircraft motion, human motion may also be parseable into several distinct modes, but it will require model-switching on much shorter time-frames, and the maneuvers are more complex and may require many more than two different modes. This is a fertile ground for research on prediction that has not yet been adequately explored. Short of this variable state dimension multiple model approach, there are adaptive techniques for tuning the process noise to increase it during maneuvers. The best prediction methods of the future will probably combine all 3 techniques: inertial sensing, specific kinematic modeling, and adaptive stochastic models that adjust to the presence or absence of “maneuvers” such as gestures or visual pursuit patterns. Better yet, virtual environment designers will select rendering platforms that can render 60 frames per second without pipelining, and even simple prediction algorithms will be sufficient for the 20-30 ms of prediction that are required in such a system.

7.5. The Engineering of Motion Tracking

To develop motion tracking systems into successful commercial products (or components of successful products) requires disciplines that are often quite foreign to the types of researchers who work with the physics and mathematics concepts outlined in the previous two sections. For such researchers, the end goal is often a proof of concept demonstration running on a UNIX or Windows computer, with sensor data read in through serial ports or data acquisition boards using vendor-supplied driver software over which the researcher does not have adequate control. Synchronization and timing may be dependent on the computer system clock, whose resolution is way too course, and the operating system randomly interrupts the tracking software to perform other system maintenance processes, causing annoying pauses in the tracking algorithm. Most obvious of all, the demonstration tracking system may require a lab cart to haul around and have so many boxes and cables that it is virtually irreproducible and treacherously unreliable, especially on days of major sponsor demos.

To go from this type of demo to a reliable, producible and cost-effective device requires methodical software and hardware engineering practices and a great attention to detail. The code must be rewritten by professional real-time embedded programmers using a real-time operating system, or perhaps no operating system but a lot of careful benchmarking, timing diagram design, and conformance testing. Bug and feature tracking, and release-level testing that were at best informal in the research laboratory take on highest priority and consume many times over the number of man-months that were required to build the original working prototype.

Besides robustness and performance, the obvious engineering goals are miniaturization, power-reduction, cost-reduction, and increased ease of use. All of these are achieved initially by stripping away support for features that aren’t yet implemented, but have been reserved for future use in case the customers demand them. This may be accomplished by moving from a rack mount unit full of separate sub-assemblies to a single-board computer with a custom interface daughter-board, and then eventually to a fully custom product design which integrates all the sensors and electronics with an embedded processor in a small plastic enclosure. Vendors often find it difficult to justify even this level of integration, because the product can no longer be modified easily to service a new group of customers with a slightly different application. If enough customers emerge with a common product need to justify the development expense, the next step is to reduce the size, power and cost further by developing an ASIC. So far, there have been few tracking applications large enough to justify such development, but that is likely to change soon due both to increasing demand in certain applications, and to sweeping changes in the semiconductor industry that are rapidly bringing down the cost of developing semi-custom ASICs. However, unless someone invents a universal tracker that does precise 6-DOF tracking in any environment, every highly miniaturized and cost-reduced tracker development is going to involve making a variety of usage-specific compromises, and probably won't result in consumer-priced tracking that is useful in other applications.

A particular need that has not yet been addressed satisfactorily for most trackers is the ability to track multiple users in a shared VE workspace without any cables. For HMD-based systems, the limiting factor has been the difficulty of making the HMDs themselves wireless, since video imagery requires much higher bandwidth than does tracking data. However, FSD-based virtual environments are increasing in popularity, and the stereoscopic viewing glasses are already wireless. There is therefore an acute need for wireless tracking solutions for these environments. Certain tracking technologies have no electronics on the object being tracked, and are therefore intrinsically wireless. These include computer vision techniques to directly track the image of the hands or their silhouettes (Leibe, 2000) or passive markers on handheld tools (Dorfmuller, 1999). For other tracking technologies which require electronic sensors on the moving object, power consumption must be reduced to allow battery operation, and some infrared or RF telemetry link provided to bring the data back to the host. Designing such telemetry links has previously required a great deal of RF engineering, but there is rapid progress recently in the development of embeddable RF modules or even single-chip radio solutions, driven by developments in the mobile computing and telecommunications sectors. Of particular interest is an emerging standard called, strangely, "Bluetooth", which is designed to facilitate initiation of spontaneous data exchange networks between cell phones, PDAs, notebooks, and other portable and office based electronics, without the user having to perform any configuration (www.bluetooth.com). The Bluetooth consortium members will soon begin to introduce a variety of low-cost single-chip radio solutions that support data rates up to 1Mbit/second, quite adequate for multiple trackers in a virtual environment.

7.6. The Systems Integration of Motion Tracking

After all of the best engineering practices have been followed, resulting in a wonderful motion tracker with low latency and state-of-the-art prediction algorithms, it gets shipped to a user who integrates it with a virtual environment simulation system and frequently gets miserable results. This happens because end-to-end system latency and image stepping are properties of the whole system and cannot be cured by an external black-box device without modifications to the rest of the system. The comments in the previous section about real-time operating systems and timing conformance testing apply as well to the development of VE system software, and this must be done in accordance with a synchronization policy that the motion tracking system supports.

Before discussing the solutions, let us consider some of the problems that result from poorly synchronized systems:

- Longer average latency
- Latency jitter
- Multiple images
- Image shearing

In a typical asynchronous virtual environment system, the tracker loop, graphics rendering process, and display screen refresh (scan out) all operate independently. The display will refresh itself at a constant rate, normally 60 Hz, following a raster scan pattern from top to bottom. There are exceptions, such as calligraphic CRT displays, retinal scanning displays, and frameless rendering (Bishop et al, 1994), but these are not in common use. The rendering process or pipeline will have a non-constant update rate that may vary from over 70 Hz down to less than 30 Hz, depending on the complexity of the part of the scene currently in view. The tracker runs internally at its own constant rate, say 130 Hz, and either spits data at the host computer continuously or provides the latest data record whenever it is polled. Figure 6 below illustrates a section of this asynchronous operation, with continuous mode tracker reporting, as the renderer frame rate drops from about 70 Hz to just below 30 Hz.

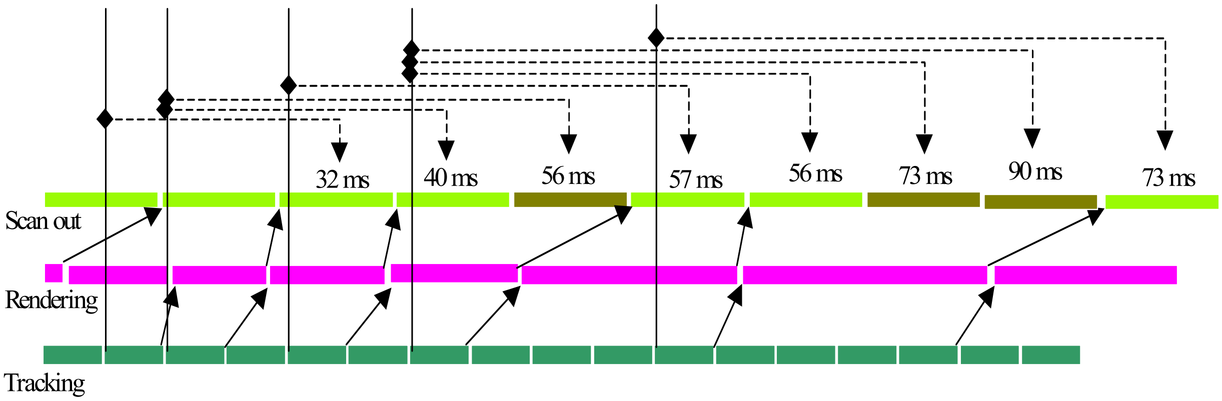

Figure 6: Illustration of typical non-synchronized VE system

The light colored bars in the top row indicate frames of the video that contain freshly rendered images; the dark bars on frames 5, 8 and 9 indicate "dropped frames" where there was no new rendering cycle computed in time, so the frame buffer scanned out the same image as the last frame. Dropped frames cause the perception of multiple images (Moore, 1996). To see this consider a 60 Hz display device driven by a graphics engine that renders at 30 Hz, so that every other frame is dropped. As the eye tracks a moving object in the series of new frames, and blends the sequence of discrete frames into an apparent continuous motion, the repeated frames do not fit into this interpolated motion trajectory, but rather create a ghost image of the object lagging behind it by a distance proportional to the speed of motion. Likewise, a 20 Hz graphics update rate would produce a triple image. A similar triple image can be seen in field sequential color displays in which the same image is redrawn three times, and the image therefore separates into non-aligned red green and blue images even at field rates of 180 Hz. From the figure, it is apparent that at rendering rates of 60 Hz or more there will be no dropped frames, from 30-60 Hz there will be single dropped frames, and as soon as the renderer dips below 30 Hz, there is the possibility of occasionally dropping two frames in a row, as shown. If frames are dropped only sporadically, they may appear as image twitching rather than a steady multiple image, but this can be equally annoying.

Even when no frames are being dropped and the image generator is rendering at a steady 60 Hz, there is a variable latency between the sampling of the head motion sensors and the display of the image. This latency ranges from 33.3 ms for best-case synchronization to 58.5 ms for worst case, for objects halfway down the raster display. When there is no synchronization policy, the latency will be varying across this range due to the beat frequencies between the display refresh, the rendering cycle, and the tracking loop. At a head rotation rate of 200°/s, this latency variation of +/- 12.5 ms will cause +/-2.5° peak spatial oscillation. Thus, in an unsychronized system, the effects of latency jitter are probably even more detrimental than the effects of the average latency itself, and the average latency is also worse in this system than it needs to be. Finally, the effect of image shearing is caused by the 16 ms difference in latency between the top of the image and the bottom. This causes objects at the bottom of the image to lag behind those at the top and vertical lines to take on a slant proportional to the speed of panning. The remainder of this section discusses approaches to solving these problems.

7.6.1 Hard Genlock

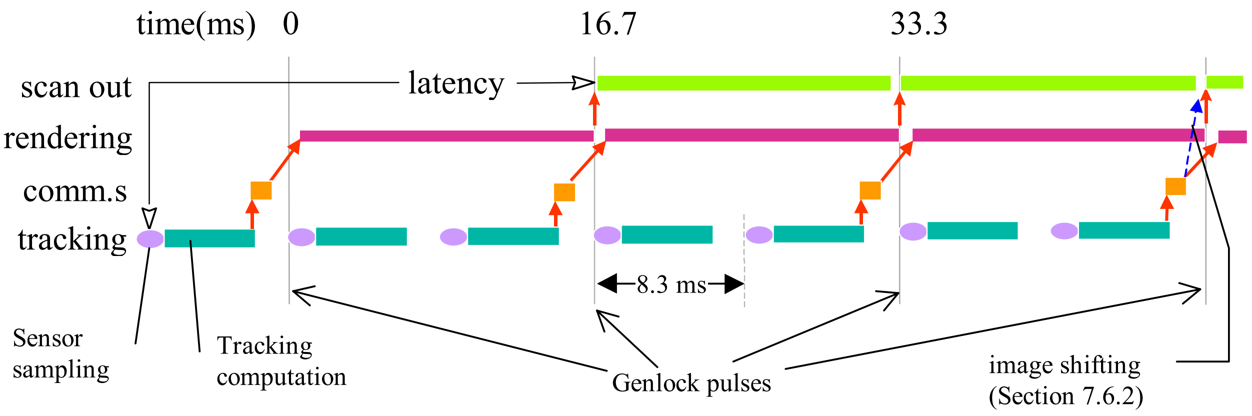

Figure 7: Baseline synchronization policy using master "genlock" signal

Figure 7 above illustrates the most traditional and in many cases still the most dependable synchronization policy that can be used to integrate a VE system with tracking. Used universally in professional video production and in many high-performance VE simulators, this system is based on a master "genlock" synchronization timing pulse at the video field rate, to which all other processes in the system are slaved. This imposes a requirement for the image generator to strictly maintain a 60 Hz frame rate. To accomplish this, the scene must be simplified until no portion of the scene causes "overloading", or else the rendering software must be provided with automatic level-of-detail control that checks every frame and merges polygons if necessary to keep the total load below the point of overloading. The case illustrated shows the lowest latency configuration, with no pipelining in the renderer. This is called guaranteed single frame latency, and is becoming the norm on modern PC graphics cards. The synchronization policy still works with pipelined graphics architectures: the latency will be larger but still constant. The tracker loop is shown running at twice the genlock rate. The first sensor sampling is initiated directly by the genlock pulse, and another one is internally initiated exactly half a frame period later. With many trackers it would be sufficient to sample at 60 Hz, but for inertial trackers a quicker update rate should be maintained for higher accuracy integration, and for DC magnetic trackers, the 120 Hz rate allows the use of a two tap filter for canceling mains interference, as discussed in Section 7.3.5. With the inertial tracker, the latency from sensor sampling to the beginning of scan out is 25 ms, as shown in the figure. For a magnetic tracker with two-tap averaging filter, the latency would be 29 ms. These values are so small that they can be compensated very effectively with prediction, especially using inertial sensors. In a pipelined rendering system, the overall latency may be increased by one or more frames, so that prediction algorithms may begin to demonstrate some noticeable overshoot. To overcome this, it is recommended that the motion tracker be sampled again right before the final stage of the rendering pipeline, with a prediction to the final frame now short enough to be very accurate, and the newest prediction be used to drive the final rendering operations. Since this view vector may be off by a degree or two from the earlier predictions, it would be necessary to perform the earlier culling with a slightly enlarged viewing frustum, then narrow it down to final size on the last stage. Note from the drawing that the tight loop timing shown is only possible if the communications of tracker data to the host can be completed before the beginning of the next frame. With an rs-232 serial port running at 115,200 bits/second, a tracker datum containing 6 values, each encoded as a 4-byte binary floating point number, can be transmitted in 2.3 ms. As long the data output can be initiated at least 2.3 ms before the end of the frame, this is workable. For longer data packets (e.g. containing multiple sensor data), less efficient encoding or slower bit rates, the serial port may be a limitation, and either a faster communications link must be found, or a longer latency endured.

7.6.2 Image Shifting

The effect of small rotation angles of the virtual camera in azimuth and elevation is nearly equivalent to a simple linear translation of the 2-D image. This fact makes it possible to perform a final adjustment of the scene using the very latest tracking data *after* the final stage of rendering, just before or during the scan out. The last cycle in Figure 7 has a dotted arrow showing how data from the tracker is used not only to start the next rendering cycle, but also to compute an image shift amount for the frame just rendered, right before scan out. The new tracker datum is based on sensor data sampled only 8.3 ms before scan out, and thus can be predicted forward with almost no error. The yaw and pitch values are compared to the predictions that had been used for the rendering process, and the differences are used to determine the necessary horizontal and vertical shift values.

Image deflection was conceived by Rediffusion in the 70's and implemented in the early 80's by NAWCTSD for a laser projector display (Breglia, 1981) and by the Institute of Sound and Vibration Research for a CRT in an HMD (Wells & Griffin, 1984; So & Griffin, 1992). In a CRT, it can be achieved with simple analog electronics which add an additional offset current to the horizontal and vertical deflection yokes of the CRT. For LCD displays, they can't be deflected this way, but the shifting can still be accomplished without bothering the main CPU by playing games with the frame buffer addressing hardware. In both cases, the image has to be slightly over-computed to allow shifting. The amount of over-computed area required can be greatly reduced by using the best possible prediction technique before rendering the image. This will also minimize perspective error distortion.

Even with a system that is tightly genlocked and designed to maintain a steady 60 Hz image rendering rate, there is always a concern that it may at some point become overloaded and drop some frames. Image shifting can also be used to help compensate for lost frames by predicting and shifting the previous frame until a newly rendered frame is available (Moore et al, 1998). In a system where 60 Hz rendering is not possible due to cost limitations, but the display must be run at 60 Hz to prevent flicker, image shifting can go a long way towards reducing the multiple images effect, although it does have some side effects for moving objects in the scene (Moore, 1996).

A more sophisticated technique which is basically an extrapolation of the image shifting concept is the "address recalculation pipeline" architecture proposed by Regan and Pose (1994). Instead of just overcomputing the frame a little on each side, they compute a complete surrounding world, rendered on six faces of a cube around the user's current head position. Once this is rendered the user with an HMD can look all around, and the orientation tracker is used to read out an appropriate portion of the 6-sided frame buffer, automatically undistorted by hardware in real time at 60 Hz. Latency with respect to head-orientation changes is therefore eliminated. However, whenever the user translates in position, all 6 faces of the cube have to be re-rendered. Since translational movement has relatively small effects on distant objects, closer objects get re-rendered first, thus minimizing the translational latency penalty. By keeping sets of objects at different distances in different frame buffers which are composited together during scan out, they can be re-rendered at different rates.

All the prediction, synchronization and shifting techniques discussed so far are designed to achieve consistent and minimal latency from the head motion to the beginning of the image scan out. However, the scan out itself takes 16 ms, and so objects towards the bottom of the display screen will suffer more latency than objects at the top. The most obvious manifestation of this is during horizontal scanning, when everything in the frame appears to slant. A solution called Just In Time Pixels (Olano et al, 1995) has been proposed, in which ideally each pixel is rendered separately with tracker data concurrent to its display time. They suggest an approximation by using a separate viewing transformation for each line, or even just two transformations for the first pixel and the last, with all the others calculated by linear interpolation. In practice, the effect will be very nearly linear over the 16 ms scan period, so all that is needed is to measure the head rotation rate about the vertical axis with a gyro, or estimate it with a Kalman filter, and then shift each scan line by an increasing amount, proportional to this rate. Reichlen (1993) implemented a frame buffer with this shifting feature built in. It is of course also possible to pre-distort each polygon in software if it is known how fast it will be moving across the screen.

7.7. Summary of Recent Progress & Future Potentials

Since Meyer et al's 1992 Survey of Position Trackers, the field has evolved and produced some new tracking options as well as some new demands. Drift-corrected gyroscopic orientation trackers have appeared, and where only orientation is needed, they now provide an affordable use-anywhere solution with sufficient resolution, robustness, responsiveness and sociability for any HMD fly-through application. Outside-in optical tracking has increasingly made inroads in motion capture, and is beginning to reach a level of real-time performance and price suitable for performance animation. Laser scanners have come out of the laboratory and are now commercially available from several sources. The UNC optical ceiling tracker has been significantly reduced in size, and at the same time the performance has been increased to extreme levels to prove the advantages of the inside-out approach and serve as a test platform for demanding research applications. The realization of the need for hybrid tracking has sprung up everywhere, with an acoustic-inertial hybrid on the market and research papers on magnetic-inertial, optical-inertial and optical-magnetic combinations signaling a diverse future. GPS has proliferated throughout the world, and driven by the huge volumes, vendors have succeeded in shrinking the complex receivers down into incredible tiny packages. The field of machine vision, bolstered by industrial parts inspection applications and the ascent of PC computing power, has made great strides in bringing formerly expensive algorithms down to the level of routine use, including a new breed of compact vision systems and smart cameras suitable for embedding. Ultra-wideband radio technology has appeared in the form of Micro-Impulse Radar products, and at least theoretically holds out the promise for an improved method of omni-directional point-to-point ranging.

Two paradigm shifts in the VE field are happening which are creating new types of requirements for motion tracking systems. The first is a trend away from head-mounted displays and towards fixed-screen projection displays such as the CAVE™, the PowerWall™, the virtual workbench, and personal mini-dome projectors. In simulation and training applications where a sense of presence is an important aspect of the application, HMDs are still the norm, but for interactive visualization and design, the easier group dynamics tend to favor the headgear-free FSD paradigm. This greatly reduces the need for high-resolution low-latency head-orientation tracking, but increases the emphasis on high-quality 6-DOF hand-tracking. Furthermore, the user is no longer forced to don a heavy headset with a cable, so the requirement for a lightweight wireless tracking device has become much more urgent. A second trend is the increasing interest in AR. This imposes tracking requirements far more challenging than any encountered in immersive VE applications. To achieve visual registration, the tracking needs to be far more accurate. At the same time, AR is normally done using a wearable computer for high mobility, so the tracking system must be able to operate over an extended range in a cluttered environment. The data needs to be available to a computer on the user, so the tracker processor unit should be wearable. So far, the consensus is that a hybrid of inside-out computer vision and inertial technology is likely to be the best fit for this problem, and that is unlikely to change since that is the solution that biological evolution has developed for the same problem. Initial offerings will require the use of specially prepared artificial fiducial marks to simplify the computer vision requirements, but the long-term goal of AR tracking developers is to make use of natural features found in both indoor and outdoor scenery to allow unrestricted tracking in arbitrary unprepared environments.

This survey has attempted to overview all the possible physical principles that can be exploited for VE-style motion-tracking applications. It is a testament to the diligence of the engineers in the field that almost all of them are already being used. However, a few ideas have been discussed that have not yet been developed as far as they could, or in some cases have not even been discussed in the literature. These represent the "low-hanging fruit" which may be able to yield new and useful tracking technologies for certain applications in the next few years. For example, the bio-kinematic reckoning approach of Section 7.3.1.2 or the bio-dynamic model-based tracking of Section 7.3.2.5 may yield new approaches to full-body avatar animation with significant mobility and cost advantages over the magnetic and optical systems that are prevalent today. The potentially reduced multipath incidence with ultra-short electromagnetic impulses mentioned in 7.3.6.3 suggests that this could eventually become a better alternative than acoustic ranging in aided inertial trackers, especially outdoors. The polarized light technique in Section 7.3.7.3 might make an acceptable alternative to the compass for correcting inertial orientation-tracker yaw drift in confined metallic environments such as inside a vehicle. The vision-aided inertial approach, with GPS priming in outdoor applications, will eventually find a wide audience as portable and wearable systems become common. Better solutions to the large-scale SLAM problem discussed in Section 7.4.4 are needed, MEMS technology must advance to the level of a single-chip IMU, and advanced computer vision capability based on massively parallel SIMD or cellular neural networks must be reduced to the level of low-power integrated vision chips. However, the research groundwork in these areas is being laid today, and it is possible to imagine in the foreseeable future ubiquitous computing devices offering new human-machine interface capabilities based on precision position and orientation tracking.

7.8. 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), pp. 302-318

Akatsuka, Y. & Bekey, G. (1998). Compensation for end-to-end delays in a VR system. Proceedings of VRAIS 98 Conference, Atlanta, GA, IEEE Computer Society Press, pp. 156-159.

Azarbayejani, A. & Pentland, A.P. (1995). Recursive estimation of motion, structure, and focal length. IEEE Transactions on Pattern Analysis and Machine Intelligence, 17(6), pp.562-575

Azuma, R. & Bishop, G. (1994). Improving static and dynamic registration in an optical see-through HMD. In SIGGRAPH 94 Conference Proceedings, ACM Annual Conference Series, Orlando, FL, August.

Badler, N., Hollick, M.J., & Graneri, J.P. (1993). Real-time control of a virtual human using minimal sensors. Presence: Teleoperators and Virtual Environments, 2(1), pp. 82-86

Bar-Shalom, Y. & Li, X.R. (1993). Estimation and Tracking Principles, Techniques, and Software, Boston: Artech House

Bar-Shalom, Y. & Li, X.R. (1995). Multitarget-Multisensor Tracking: Principles and Techniques. ISBN 0-9648312-0-1.

Baranek, Leo L. (1954). Acoustics. New York: McGraw-Hill.

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

Bishop, G. Fuchs, H., McMillan, L., & Scher-Zagier, E.J. (1994). Frameless rendering: double buffering considered harmful. In SIGGRAPH 94 Conference Proceedings, ACM Annual Conference Series, Orlando, FL, August.

Blood, E.B. (1989) Device for quantitatively measuring the relative position and orientation of two bodies in the presence of metals utilizing direct current magnetic fields. US Patent 4,849,692.

Breglia,D. (1981). Helmet mounted laser projector. Proceedings of the Third I/ITSEC Conference, pp. 8-18

Broida, T.J., Chandrashekhar, S., & Chellappa, R. (1990). Recursive estimation of 3D motion from a monocular image sequence. IEEE Transactions on Aerospace and Electronics Systems, 26(4), pp.639-656

Brown, R.G. & Hwang, P.Y.C. (1992). Introduction to Random Signals and Applied Kalman Filtering. New York: John Wiley & Sons, Inc.

Chenavier, F. & Crowley, J.L. (1992). Position estimation for a mobile robot using vision and odometry. IEEE International Conference on Robotics and Automation. Nice, France, pp. 2588—2593

Csorba, M., Uhlmann, J.K. & Durrant-Whyte, H.F. (1997). A sub optimal algorithm for automatic map building. In Proceedings of the American Control Conference, Albuquerque, NM, pp. 537-541

Dorfmuller, K. (1999). An optical tracking system for VR/AR-applications. In M.Gervautz, A. Hildebrand & D. Schmalstieg (Ed.s), Proceedings of the Eurographics Virtual Environments 99 Workshop, Vienna, May, 1999, Vienna: Springer-Verlag, pp.33-42

Drascic, D. & Milgram, P. (1996). Perceptual issues in augmented reality. In Proceedings of SPIE Vol. 2653: Stereoscopic Displays and Virtual Reality Systems III, pp. 123-134

Ducharme, A.D., Baum, P.N., Wyntjes, G., Shepard, O., & Markos, C.T. (1998). Phase-based optical metrology system for helmet tracking. Proceedings of SPIE Vol. 3362 Helmet and Head-Mounted Displays III, AeroSense 98, Orlando, FL.

Ellis, S.R., Young, M.J., & Adelstein, B.D. (1999). Discrimination of changes in latency during head movement.

Proceedings of HCI ‘ 99 International, Munich.

Ellis, S.R., Young, M.J., Adelstein, B.D., & Ehrlich, S.M. (1999). Discrimination of changes in latency during voluntary hand movement of virtual objects. Proceedings of the Human Factors and Ergonomics Society, Houston,TX.

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 VRAIS 99 Conference, Houston, TX, IEEE Computer Society Press, pp. 218-221.

Emura, S. & Tachi, S. (1994). Compensation of time lag between actual and virtual spaces by multi-sensor integration. In Proc. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 94), pp. 463--469.

Ferrin, F.J. (1991). Survey of helmet tracking technologies. In Proc. SPIE, vol. 1456, pp. 86-94.

Foxlin, E. (1993). Inertial head-tracking. M.S. Thesis, MIT Dept. of Electrical Engineering and Computer Science, Cambridge, MA.

Foxlin, E. (1996). A complementary separate-bias kalman filter for inertial head-tracking. In Proc. IEEE VRAIS 96. IEEE Computer Society Press, March-April 1996.

Foxlin, E. (1997). Inertial orientation tracker apparatus having automatic drift compensation for tracking human head and other similarly sized body. U.S. Patent 5,645,077. Filed June 16, 1994.

Foxlin, E., Harrington, M. & Pfeiffer, G. (1998). Constellation™: a wide-range wireless motion tracking system for augmented reality and virtual set applications. In SIGGRAPH 98 Conference Proceedings, ACM Annual Conference Series, Orlando, FL.

Foxlin, E. (2000). Head-tracking relative to a moving vehicle or simulator platform using differential inertial sensors. Proceedings of Helmet and Head-Mounted Displays V, SPIE vol. 4021, AeroSense Symposium, Orlando, FL.

Foy, W.H. (1976). Position-location solutions by taylor-series estimation. IEEE Transactions on Aerospace and Electronic Systems, 12(2), pp. 187-194.

Friedmann, M., Starner, T., & Pentland, A. (1992). Device synchronization using an optimal filter. Proceedings of the 1992 Symposium on Interactive 3D Graphics., Cambridge, MA.

Gelb, A., ed. (1974). Applied Optimal Estimation. Cambridge, Massachusetts: The MIT Press.

Getting, I. (1993). The global positioning system. IEEE Spectrum, December, 1993.

Harelick, R.M., Lee, C.N., Ottenberg, K., & Nolle, M. (1994). Review and analysis of solutions of the three point perspective pose estimation problem. International Journal of Computer Vision, 13(3), pp. 331-356

Harris, C. (1992). Geometry from visual motion. In A. Blake & A. Yuille, (Ed.s), Active Vision (pp. 263-284) Cambridge: The MIT Press.

Held, R., Efstathiou, A. & Greene, M. (1966). Adaptation to displaced and delayed visual feedback from the hand. Journal of Experimental Psychology, 72, pp. 887-891.

Hoff, W., Nguyen, K., & Lyon, T. (1996). Computer vision-based registration techniques for augmented reality. Proceedings of Intelligent Robots and Computer Vision XV, SPIE Vol. 2904, Boston, MA, pp. 538-548.

Holloway, R.L. (1997). Registration error analysis for augmented reality. Presence 6(4), pp.413 – 432.

Ishii, M. & Sato, M.(1994). A 3D space interface device using tensed strings. Presence, *3(1)*:, pp 81-86.

Julier, S.J. & Uhlmann, J.K. (1997). A non-divergent estimation algorithm in the presence of unknown correlations. In Proceedings of the 1997 American Control Conference.

Kalman, R.E. (1960). A new approach to linear filtering and prediction problems. ASME Transactions Journal of Basic Engineering, 82(1), pp. 35-45

Kim, D., Richards, S.W., & Caudell, T.P. (1997). An optical tracker for augmented reality and wearable computers. Proc. of IEEE Virtual Reality Annual International Symposium (VRAIS 97), pp. 146-150

Koller, D., Klinker, G., Rose, E., Breen, D., Whitaker, R. & Tuceryan, M. (1997). Real-time vision-based camera tracking for augmented reality applications. Proceedings of ACM VRST 97 Conference, Lausanne, Switzerland.

Kuipers, J. (1975). Object tracking and orientation determination means, system and process. U.S. Patent 3,868,565.

Lanzl, C. & Werb, J. (1998). Position location finds applications. Wireless Systems Design, June, 1998. Cleveland: Penton Media.

Leibe, B., Starner, T., Ribarsky, W., Wartell, Z., Krum, D., Singletary, B. & Hodges, L. (2000). The perceptive workbench: Toward spontaneous and natural interaction in semi-immersive virtual environments. In Proceedings of the Virtual Reality 2000 Conference, New Brunswick, NJ, IEEE Computer Society Press, pp. 13-20

Leonard, J.J., & Feder, H.J.S. (2000). A computationally efficient method for large-scale concurrent mapping and localization. To appear in Robotics Research: The Ninth International Symposium, J. Hollerbach & D. Koditschek (Ed.s) London: Springer-Verlag.

Liang, J.D., Shaw, C. & Green, M. (1991). On temporal-spatial realism in the virtual reality environment. Proceedings of the Fourth Annual ACM Symposium on User Interface Software and Technology, Hilton Head, SC, November.

Livingston, M.A. (1998). Vision-based tracking with dynamic structured light for video see-through augmented reality. Doctoral dissertation, Dept. of Computer Science, University of North Carolina, Chapel Hill.

Longuet-Higgins, H.C. (1981). A computer program for reconstructing a scene from two projections. Nature, 293, pp.133-135, Sept, 1981.

Lynch, D. (1998). Coriolis vibratory gyros. Included as Annex B in IEEE gyro and Accelerometer Panel Working Draft P1431/D16, Standard Specification Format Guide and Test Procedure for Coriolis Vibratory Gyros. IEEE Standards Department, June 1999.

Manolakis, D. (1996). Efficient Solution and Performance Analysis of 3-D Position Estimation by Trilateration. IEEE Trans. on Aerospace and Electronic Systems, 32(4)

Manyika, J.M. & Durrant-Whyte, H.F. (1992). An information-theoretic approach to management in decentralized data fusion. SPIE vol. 1828 Sensor Fusion V,pp. 202-213

McEwan, T. (1993). Ultra-short pulse generator. U.S. Patent 5,274,271. Filed July 12, 1991.

McIntyre, G. & Hintz, K.J. (1998). Sensor measurement scheduling: an enhanced dynamic, preemptive algorithm. Optical Engineering, 37(2), pp. 517-523

Mellor, J.P. (1995). Realtime camera calibration for enhanced reality visualization. Proceedings of Computer Vision, Virtual Reality and Robotics in Medicine (CVRMed 95), Nice, France, pp. 471-475, IEEE.

Meyer, K., Applewhite, H.L., & Biocca, F.A..(1992). A Survey of Position Trackers. Presence: Teleoperators and Virtual Environments, 1(2), pp. 173--200.

Mine, M., Brooks, F.P., & Sequin, C.H. (1997). Moving objects in space: exploiting proprioception in virtual-environment interaction. In SIGGRAPH 97 Conference Proceedings, ACM Annual Conference Series.

Molet, T., Boulic, R., & Thalmann, D. (1999). Human motion capture driven by orientation measurements. Presence: Teleoperators and Virtual Environments, 8(2), pp. 187-203.

Moore, R.G. (1996). Multiple image suppresion. Proceedings of the 18th I/ITSEC Conference, Orlando

Moore, R.G., Pope, C.N., Foxlin, E. (1998). Toward minimal latency simulation systems. Proceedings of American Institute of Aeronautics and Astronautics Conference, vol. 4176, Boston.

Nardone, S.C. & Aidala, V.J. (1980). Necessary and Sufficient Observability Conditions for Bearings-Only Target Motion Analysis, Technical Report, Naval Underwater Systems Center, Newport, RI

Nash, J.M. (1977). Optimal allocation of tracking resources. In IEEE International Conference on Decision and Control, pp 1177-1180

Neumann, U. & Cho, Y. (1996). A self-tracking augmented reality system. Proceedings of ACM VRST 96, pp. 109-115

Nixon, M.A., McCallum, B.C., Fright, W.R., & Price, N.B. (1998). The effects of metals and interfering fields on electromagnetic trackers. Presence: Teleoperators and Virtual Environments, 7(2), pp. 204-218

Palovuori, K.T., Vanhala, J.J. & Kivikoski, M.A. (2000). Shadowtrack: a novel tracking system based on spread-spectrum spatio-temporal illumination. Presence: Teleoperators and Virtual Environments, 9(6).

Purcell, E.M. (1965). Electricity and Magnetism. New York: McGraw-Hill.

Quan, L. & Lan, Z. (1999). Linear n-point camera pose determination. IEEE Transactions on Pattern Analysis and Machine Intelligence, 21(8), pp. 774-780

Raab, F.H., Blood, E.B., Steiner, T.O., Jones, H.R. (1979). Magnetic position and orientation tracking system. IEEE Transactions on Aerospace and Electronic Systems, 15(5), pp. 709-718.

Regan, M. & Pose, R. (1994). Priority rendering with a virtual reality address recalculation pipeline. In SIGGRAPH 94 Conference Proceedings, ACM Annual Conference Series, Orlando, FL, August.

Reichlen, B. (1993). Sparcchair: A 100 million pixel display. In Proceedings of VRAIS, Seattle, WA.

Smith, R., Self, M., & Cheeseman, P. (1987). A stochastic map for uncertain spatial relationships. Fourth International Symposium on Robotics Research, MIT Press

So, R.H.Y. & Griffin, M.J. (1992). Compensating lags in head-coupled displays using head position prediction and image deflection. Journal of Aircraft, 29(6), pp. 1064-1068

So, R.H.Y. & Griffin, M.J. (1995). Effects of lags on human operator transfer functions with head-coupled systems. Aviation, Space, and Environmental Medicine 66(6), pp. 550-556

Sorensen, B., Donath, M., Yang, G.B., & Starr, R. (1989). The Minnesota scanner: a prototype sensor for three-dimensional tracking of human body segments. IEEE Transactions on Robotics and Automation 5(4), pp. 499-509.

Suryanarayanan, S. & Reddy, N. (1997). EMG-based interface for position tracking and control in VR environments and teleoperation. Presence: Teleoperators and Virtual Environments, 6(3), pp. 282-291.

Sutherland, I.E. (1968). A head-mounted three-dimensional display. 1968 Fall Joint Computer Conference, AFIPS Conference Proceedings, 33, pp. 757-764

Taylor, J., Ed. (1995). Introduction to Ultra-Wideband Radar Systems. CRC Press. ISBN: 0-8493-4440-9.

Wang, J.F., Azuma R., Bishop, G., Chi, V., Eyles, J. & Fuchs, H. (1990). Tracking a head-mounted display in a room-sized environment with head-mounted cameras. SPIE Proceedings vol. 1290: Helmet-Mounted Displays II., Orlando, FL

Ware, C. & Balakrishnan, R. (1994). Reaching for objects in VR displays: lag and frame rate. ACM Transactions on Computer-Human Interaction, 1(4), pp. 331-356.

Watson, B., Spaulding, V., Walker, N., & Ribarsky, W. (1997). Evaluation of the effects of frame time variation on VR task performance. Proceedings of VRAIS 97 Conference, IEEE Computer Society Press.

Welch, G. & Bishop, G. (1997). Single-constraint-at-a-time tracking. In SIGGRAPH 97 Conference Proceedings, ACM Annual Conference Series. ACM Press

Welch, G., Bishop, G., Vicci, L., Brumback, S., Keller, K., & Colluci, D. (1999). The HiBall tracker: high-performance wide-area tracking for virtual and augmented environments. Proceedings of VRST 99, London.

Wells, M.J. & Griffin, M.J. (1984). Benefits of helmet mounted display image stabilization under whole body vibration. Aviation, Space, and Environmental Medicine, 55(1), pp. 13-18

Williams, T. (1999). Millimeter waves and the EHF bands. Proceedings of The 25th Eastern VHF/UHF Conference, August, 1999, Vernon, CT. Published by ARRL. ISBN: 0-0-87259-760-1

Win, M.Z. & Scholtz, R.A. (1998). Impulse radio: how it works. IEEE Communications Letters 2(2), pp. 36-38.

Zimmerman,T.G., Smith, J.R., Paradiso, J.A., Allport, D., Gershenfeld, N. (1995). Applying electric field sensing to human-computer interfaces. In Proceedings of the Computer-Human Interface Symposium '95. ACM Press.