Hybrid Self-Tracker: An Inertial/Optical Hybrid Three-Dimensional Tracking System

A Dissertation Proposal by
Gregory F. Welch

Department of Computer Science
University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-3175

Thesis

Inertial navigation and computer vision systems embody complementary frequency and velocity characteristics that motivate the combination of the two approaches in a completely self-contained system for tracking the three-dimensional position and orientation of a moving person or object in a dynamic environment.

Abstract

This proposal describes a new approach to tracking the three-dimensional position and orientation of a moving person or object in a dynamic environment. The approach involves using a technique similar to that historically employed in spacecraft navigation whereby sun and star trackers are used to optically aid the spacecraft's inertial navigation system (INS). Similarly the idea of the hybrid Self-Tracker is to track the (earthly) position and orientation of a person or object by employing an INS as the primary means of tracking, and then supplementing that with an outward-looking computer vision system (CVS) that when aided by the INS can lock on to still targets in a not necessarily static environment.

The novel aspect of this approach is that the complementary behavior of each system is leveraged to obtain more accurate and stable tracking information than either system alone. With an INS, bias and drift errors become noticeable during periods of slow movement. However, during such periods these errors could be controlled by a CVS which exhibits its best behavior under such conditions. Conversely a CVS typically performs worst during rapid movement, precisely the conditions where the INS signal-to-noise ratio is high. In addition, while a typical CVS is affected by unrelated motion in an environment, an INS is not and can provide assistance with static feature discrimination. Kalman filter theory can be employed in weighting each subsystem most heavily in the circumstances where it performs best, thus providing more accurate and stable estimates than either system alone.

Advantages of this new approach include: high-rate and low-latency position and orientation information across a wide range of motion; operation in a dynamic environment; self-contained passive operation.

1. Introduction

1.1 General Motivation

One of the important problems in Virtual Environment (VE) research today is that of providing a fast, accurate, and possibly even portable method for reliably tracking a computer user's real-world position and orientation. Such tracking is necessary in VE systems because a user must continually be provided with two-dimensional computer generated images that match the user's three-dimensional real-world position and orientation. In certain applications, if the user's position and orientation are not tracked accurately or fast enough, disturbing or even harmful effects can be observed. There are two general VE situations for which we desire fast and accurate position and orientation information.

The first situation has come to be known nominally as *Virtual Reality*, where a user becomes completely immersed in a computer generated "world" by donning some form of an opaque head-mounted display system [SUTH68]. In this case, the user's real-world view is completely replaced by an artificial view formed by a sequence of computer generated images. As such, if the changes that a user observes in the sequence of computer generated images does not match the changes that the user would expect to see based on their internal (biological) sense of changing orientation and position, a user can experience discomfort or even sickness. Typically however, tracking problems in Virtual Reality systems only pose an inconvenience.

The second situation has come to be known nominally as *Augmented Reality*, where a user's view of the real-world is not replaced but instead augmented by computer generated images that are superimposed on a user's otherwise unobstructed real-world view. In this case, the head-mounted display system is not opaque, but is instead transparent. Here the computer generated images would appear to float in front of the user, partially obscuring their otherwise natural view. For example, a surgeon might some day use an Augmented Reality system as a surgical aid. Here the system could be used to provide the surgeon with visual information, e.g. vital statistics or even graphical guidance that is superimposed directly on his view of the patient. Tracking problems in Augmented Reality systems pose more than an inconvenience—small errors in registration of the artificial and real images can severely impact the usefulness of such systems.

In appropriately constrained work spaces, mechanical tracking systems can provide a high degree of speed and absolute accuracy. At the cost of sensitivity to magnetic interference, greater sociability[1] can be afforded by using magnetic devices in place of mechanical linkages [Meyer92]. Acoustic tracking systems while insensitive to magnetic interference are sensitive to temperature, and provide only position information directly. On the other hand, optical tracking systems can in principle offer sufficiently accurate measurements of both position and orientation over relatively large working volumes, without many of the problems inherent in magnetic or acoustic tracking systems. Current optoelectronic tracking systems are typically limited by line-of-sight requirements, they are relatively costly to implement, and once implemented are somewhat inflexible in terms of the location and size of the working volume.

One might therefore suppose that an "ideal" tracking system might be built using either computer vision or inertia sensing techniques. Neither of these techniques suffers from the particular above-mentioned problems, although each has its own unique problems. For interactive work with computers, Bishop's Self-Tracker [Bishop84] is the classic attempt at the former, while Foxlin has attempted to solve the problem with the latter [Foxlin93]. These attempts and other related work are discussed further in "Related Work" on page 10. For the moment however, let me proceed to motivate my particular hybrid with some basic analysis of the two individual methods.

---

[1] Sociability here refers to the ability to simultaneously track multiple targets (e.g. multiple users) in a common environment.

1.2 Computer Vision System (CVS) Navigation

In order for a passive computer vision system (CVS) or pattern recognition based tracking system to operate effectively across a wide range of frequency[1] and velocity[2], one would want the employed image processing techniques to be extremely fast. Fast image processing is necessary (desired) for a variety of reasons. For example, if the 3-D motion is being estimated by observing the affine motion of objects on a monocular 2-D (or 1-D) image plane, fast image processing will result in small changes from frame to frame. Such small changes often provide opportunities for simplifying approximations. These simplifying approximations then generally result in faster processing, hence smaller image changes, faster processing, etc. Such a "spiral of goodness" is described by Bishop in [Bishop84]. Even in the case where such simplifying approximations are not used, e.g. when directly tracking the 3-D positions of scene points via a stereo camera setup, high speed is still required to capture the basic high frequency and high velocity physical motion of (for example) one's head or hand.

Despite our best hopes however, there is a finite limit to the speed with which features in a pair of temporally sequential two-dimensional image frames can be compared. In this section we will briefly explore this limitation and the resulting impact on the frequency and velocity response of the image processing system.

In any implementation of a computer vision based system that processes temporally or spatially separated discrete images there will exist a constant minimum amount of time t_{min} required to process each set of images to generate some form of a motion estimate. The inverse of this minimum time determines the Nyquist (sampling) rate of the system, limiting the ideal frequency response to half this rate as shown below in Figure 1 below.

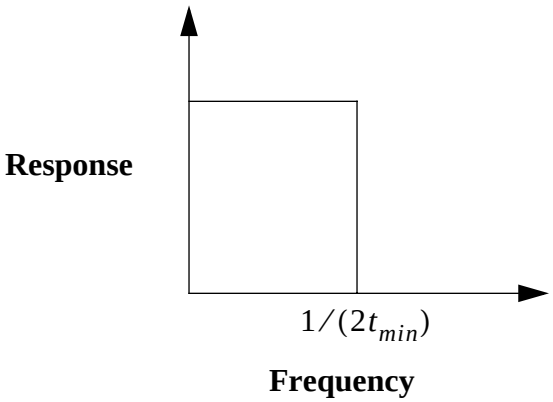

Figure 1. Ideal frequency response for minimum image processing time t_{min}

As an example, if our imaging device contains 256x256 pixels and we are able to examine pixels serially at a rate of 50MHz, the time required to examine an entire image (e.g. looking for trackable features) would be approximately 15 ms[3]. Further more, in section 3.3 on page 14 we will argue that a stereoscopic CVS is necessary to estimate position and orientation. This requirement implies processing of two images on each update, increasing our estimate of image processing time to 30 ms.This image processing time corresponds to a frame rate of approximately 33 Hz, or a

---

1. Consider physical sinusoidal oscillations in one of the six degrees of freedom in our three-dimensional position and orientation tracker.
2. Linear or angular velocity.
3. Assuming approximately 10 operations (cycles) per pixel. It might be possible to improve feature searches by using previous feature positions as starting points.

physical sinusoidal cutoff frequency of approximately 17 Hz.

As reasoned by Foxlin [Foxlin93] and measured by Azuma [Azuma94], head motion energy above 20 Hz is likely to be very small. It therefore seems that our rough 17 Hz cutoff might suffice. Indeed if we could process the images faster, we could improve (raise) this cutoff. However, one would also like to implement some filtering to improve the reliability of the CVS estimates[1], which will lower the cutoff frequency. In any case, it appears that a filtered CVS is suited to providing *low-frequency* estimates of position. In section 3.2 on page 13 we will see specifically why such a independent low-frequency estimator of position and orientation so nicely complements an inertial navigation system.

Having looked at the frequency characteristics of a stereoscopic computer vision system, we point out that the frequency analysis of such a CVS does not tell the entire story. In addition to the frequency of the expected motion, we must consider the *velocity* of the expected motion, i.e. the magnitude of the anticipated sinusoidal motion.

In Figure 2 below we see sinusoidal waveforms representing two possible physical motion trajectories of the same frequency but different magnitudes. Assuming that the frequency is well below the cutoff represented in Figure 1, what effects do the different velocities have on the ability of a computer vision based system to track the motion of an image feature between two temporally sequential two-dimensional image frames?

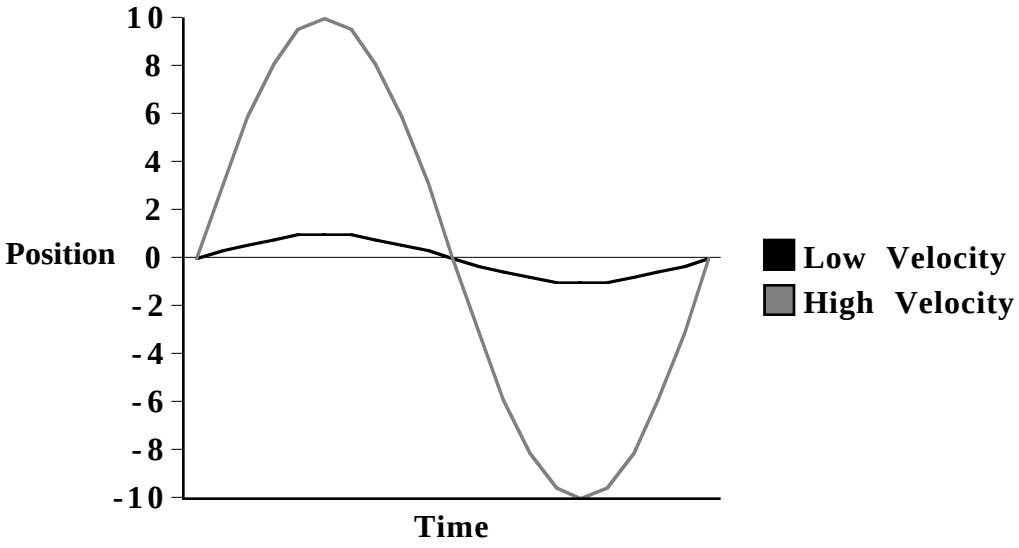

Figure 2. Two position waveforms, same frequency, different velocities

Consider the simplified two-dimensional (pinhole) camera and scene model shown below in Figure 3. As a scene point located distance *d* from the camera translates laterally by a distance *M*, the image of that scene point translates a distance *m* across the image plane of the camera.

---

1. It will be argued in section 3.3 that stereoscopic depth computations are necessary to provide independent CVS estimates of position and orientation. Such depth computations are inherently susceptible to noise and can be improved by appropriate error modeling and filtering [Matthies86].

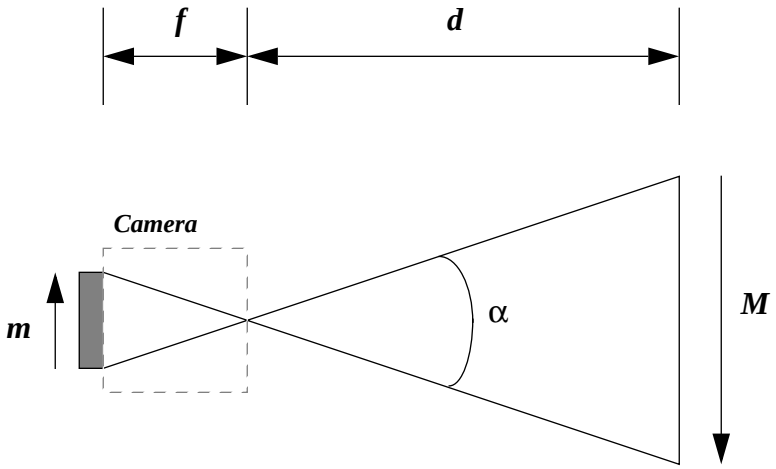

Figure 3. Simplified (pinhole) camera and scene model

Conversely, given the camera's focal length *f*, and the measured translation *m* of a feature movement across the image plane, the distance *M* through which the physical object corresponding to the tracked feature translates, is given by

$$M = \frac{dm}{f} \tag{1}$$

The physical motion *M* occurs at some average velocity *v* over a time period *t*, i.e. $M=v_M t$. Therefore given the constant minimum amount of time $t = t_{min}$ required to process a pair of temporally sequential 2-D image frames and the image sensor size μ, the maximum detectable velocity corresponding to translation *M* at distance *d* is found from

$$'_{max} = \frac{d\mu}{f t} \tag{2}$$

This maximum velocity could be associated with a still camera and an object translating through distance *M*, or as in the case of our outward-looking tracker, with a fixed scene object and a camera translating through distance *M*.

As an example, let's examine a computer vision system with an image sensor size of 1 cm, a focal length of 35 mm, and our previous estimate of 30 ms to process a pair of images. If our CVS is tracking scene objects that are on average 2 meters from the sensor, equation (2) tells us that the maximum detectable translational sensor velocity is approximately 19 meters per second.

Figure 4 below plots maximum translational velocity against scene object distance *d* for various image processing times *t*. To put things in perspective, I have performed some simple experiments and found that the fastest that I can translate my hand through a distance of 1/2 meter is under 3 meters per second.

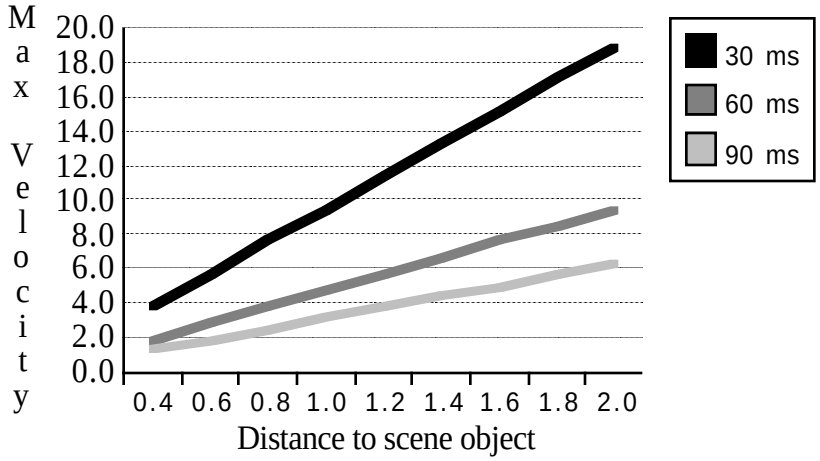

Figure 4. v_M vs. d for example values of t from equation (2)

However, let us also look at rotational velocity. If we define the origin of rotation to be at the principal point of the camera, i.e. the "pinhole" of our simplified camera in Figure 3, then the camera can be rotated a maximum of α degrees between processing of image pairs lest currently tracked features be lost. The angle α, the field of view, is defined below in equation (3) where μ is the physical size of the image sensor, and f is the focal length of the camera.

$$\alpha = \operatorname{asin}\left(\frac{\mu}{f}\right) \tag{3}$$

The maximum rotational velocity v_α of our CVS is then given by equation (4).

$$v_\alpha = \frac{\alpha}{t} \tag{4}$$

For the camera parameters above, equation (3) gives us a field of view α of approximately 17 degrees. From equation (4) we see that with our estimate of 30 ms for image processing time, this corresponds to a maximum rotational velocity v_α of approximately 550 degrees per second.

As the response time of the CVS increases (e.g. as a result of filtering), things become worse as shown below in Figure 5. To put things in perspective here, note that while typical rotational velocities are under 500 degrees per second, Foxlin argued for a rotational velocity specification of 1000 degrees per second [Foxlin93].

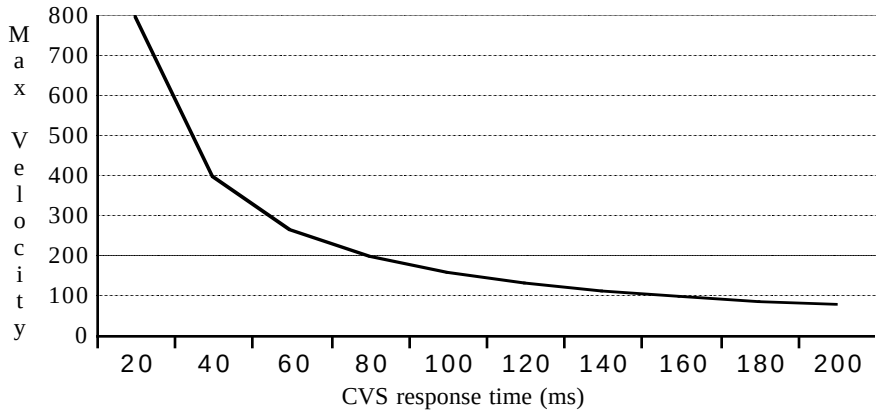

Figure 5. v_α vs. *t* from equation (4)

There is also the question of resolution to be considered. As can be seen from equation (2) and observed in Figure 4, the farther away a tracked object (i.e. as *d* increases) the higher its allowable relative translational velocity. However, from equation (1) we also see that the translation *m* of a feature on the image plane is given by equation (5) below as follows.

$$m = \frac{fvt}{d} \tag{5}$$

If the motion $M=vt$ remains constant, then as the distance *d* to an object increases, the observed feature translation on the image plane decreases. This reduced translation on the image plane can cause problems as it approaches the resolution of the imaging device. For example, for our 1 cm 256x256 pixel image sensor, the pixels are spaced approximately 39 microns apart. At a scene object distance of 1/2 meter, we can resolve approximately 1/2 mm of translation. At 2 meters, the translational resolution increases (worsens) to approximately 2 mm. The resolution is affected by the focal length also, so if we make the assumption that there will always be interesting objects within some specified distance, then fixing *f* appropriately can generally alleviate this problem. In addition, sub-pixel processing using grey-scale images can improve the resolution.

Notice also from equation (5) that as either the velocity *v* of motion *M* or the time *t* between image frames increases, the translation *m* of the induced image plane feature also increases. Given finite sensor dimensions, both *v* and *t* become limiting parameters.

In practice, when designing the computer vision system we would choose components that would define some physical limits of *f* and *m*, and hence *d* also (maximum *d*). From equations (2) and (4) we see that the final parameter needed to determine the maximum velocity[1] is t_{min}. Obviously based on equations (2) and (4) we want to keep t_{min} as small as possible in order to track image features while moving at higher velocities.

However as stated earlier there is always some finite t_{min} no matter how small. Assuming that the image processing time is fixed, it is this time that will limit the ideal velocity response[2] of the tracking system. This ideal limitation is shown below in Figure 6. Once again however equations (2) and (4) do not present the whole story. While the ideal response will resemble that of the solid line in Figure 6, a more realistic (albeit qualitative) response is shown by the dashed line. The degradation in practice is caused by several factors. Primarily as velocity increases the increasing dis-

---

1. Velocity used in a general sense here, reflecting both translational and rotational velocities

2. Response used as a qualitative indication of the system's ability to track objects.

parity between subsequent image frames generally causes increased difficulty in feature correspondence and/or optical flow determination, resulting in increasingly unreliable results.

All of these problems are normally further confounded by assumptions made about the environment in which the CVS is operating. If one is going to infer self-motion from changes in images, one generally needs to assume that all of the observed changes in the images are due to that self-motion and not to other objects moving in the environment. This static environment assumption is usually violated, even if there is only one person in the environment (consider a person's hand moving in front of their face—"Did I move with respect to the hand or did the hand move with respect to me?").

So as we work to make such a system faster and simultaneously less sensitive to noise, we are inclined to believe that the estimates offered by a computer vision based tracking system will be smoother and more reliable at lower frequencies and velocities, and in environments that are not changing independently (i.e. static environments).

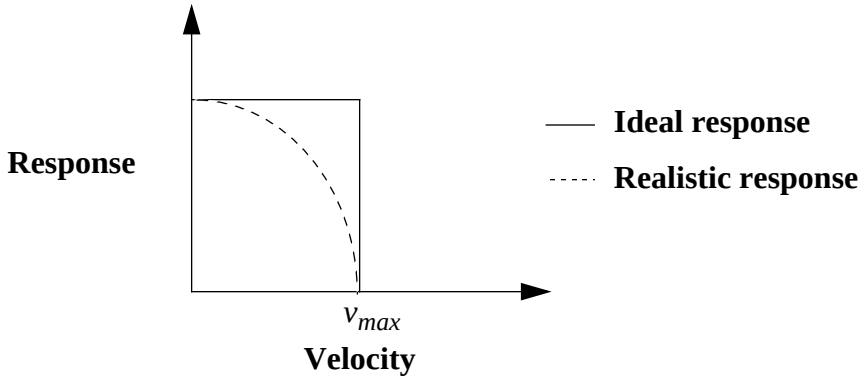

Figure 6. Response[1] vs. velocity based on equation (2)

1.3 Inertial Navigation System (INS) Navigation

When compared to computer vision based systems, inertial navigation systems (INS) exhibit completely complementary behavior in terms of both frequency and velocity response: low relative error at high frequencies and velocities, and high relative error at lower frequencies and velocities. At low velocities (very slow or no movement) one must in practice contend with pronounced bias and drift error (noise). As movement slows, such noise begins to grow with respect to the true signal. In this section we will briefly explore this behavior and the resulting impact on the response of a purely inertial navigation system[2].

---

1. Response used as a qualitative indication of the system's ability to track objects.
2. For the purpose of this discussion, consider an INS with one or more linear accelerometers and one or more angular rate (velocity) sensors.

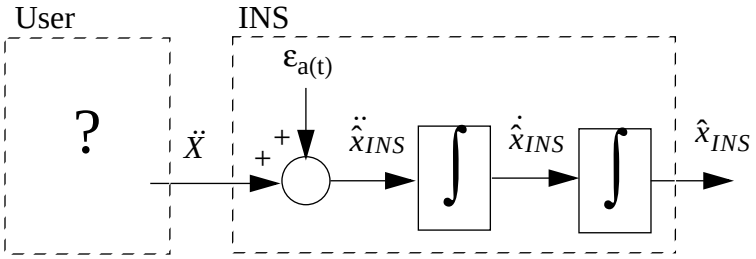

Figure 7. Transformation of user acceleration to INS-based position estimate

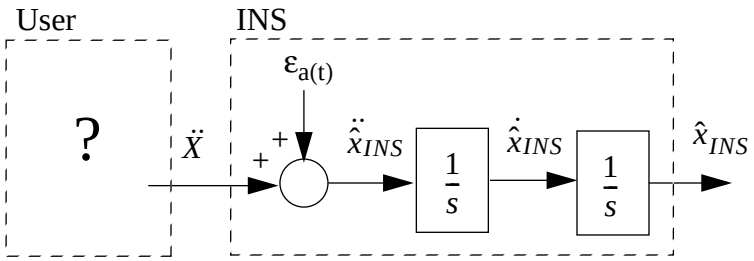

Figure 8. Continuous frequency coefficients for Figure 7

Models useful in explaining the frequency characteristics of an INS are shown above in Figure 7 and Figure 8. In each of the figures the dashed box marked “User” contains a question mark to indicate that the user’s acceleration (motion) is unknown. In Figure 7 the dashed box marked “INS” shows acceleration measurement noise $\varepsilon_{a(t)}$ being summed with the ideal user acceleration signal, and the sum then being integrated twice to obtain a position estimate. In Figure 8 the same box shows the corresponding transfer function coefficients.

The user’s true acceleration is their position twice differentiated, i.e. the user’s acceleration is weighted by the square of the frequency (*s*) of their motion. This acceleration signal is then weighted by the inverse square of the frequency (*s*) as it is integrated twice in the INS to obtain a position estimate. The end result is a unity frequency weighting of the position in the final estimate.

From Figure 8 we also see that any electrical noise $\varepsilon_{a(t)}$ incurred during the measurement of the accelerometer output is weighted solely by the inverse square of the frequency (*s*) as it is integrated twice in the INS. The end result is an inverse square frequency weighting of electrical measurement noise in the final position estimate.

In Figure 9 below the estimated position signal, noise, and signal-to-noise ratio for an INS (using accelerometers) are plotted together against frequency. This figure demonstrates why the practical application a solely INS-based tracking system is impractical. At low frequencies the position estimate noticeably diverges as measurement noise is erroneously interpreted as acceleration. The most common sources of low frequency noise are the unavoidable and often time-dependent random “DC” (or very low frequency) biases. Such bias errors can cause an INS based tracker to report that a subject is moving even when that subject is completely still, or conversely to report that a subject is still when in fact they are slowly moving.

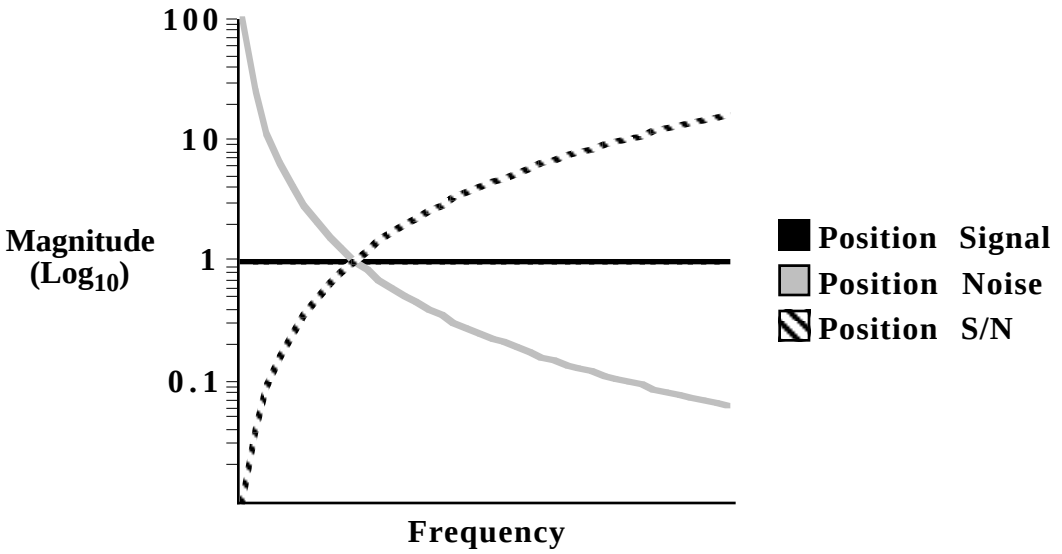

Figure 9. Logarithmic plots of typical INS position signal vs. noise (constant velocity)

With respect to velocity we see similar behavior. Velocity sensing devices (e.g. angular rate sensors) produce an output voltage that is proportional to the velocity. Therefore as angular velocity increases in magnitude, the voltage also increases in magnitude, improving the ratio of measured signal to measurement noise. Conversely as velocity decreases, the magnitude of the device output voltage also decreases, increasing the sensitivity to measurement noise in a linear fashion.

These signal-to-noise characteristics provide some insight into why inertia sensing techniques are most reliable during relatively high frequency and velocity movement, precisely the opposite of a properly filtered CVS. Also, unlike a CVS-based tracker, an INS-based tracker is completely insensitive to independent (unrelated) motion in the environment. A final "win" with an INS is that the operation is relatively simple and can therefore be very fast. The discrete implementations of the integrators shown in Figure 7 are simply adders. These additions can proceed generally as fast as measurements can be taken, providing estimates of position and orientation at a very low latency and high frequency (the inter-sample time and its inverse respectively).

One unrelated form of INS error is that caused by mechanical misalignment. Misalignment errors cause a portion of the motion along or about one axis to be incorrectly interpreted as motion along or about another axis. Such errors can be (if necessary) modeled in a Kalman filter at the cost of additional states.

It is the complementary frequency/velocity behavior and environmental sensitivity that leads me to believe that a hybrid approach combining a computer-vision based system and an INS based system holds great promise in solving the tracking problem in general. Furthermore it is the self-contained (autonomous) nature of the two individual systems that offers promise in solving the *self-tracking* problem in particular.

2. Related Work

The notion of a Self-Tracker was introduced by Bishop in his dissertation proposal [Bishop82] and later in his finished dissertation [Bishop84]. Like Bishop's pioneering optically based Self-Tracker my hybrid will offer unrestricted user motion, large working environments, the "sociability" of multiple nearby trackers as described by Meyer et al. [Meyer92], and immunity to most traditional interference. Unlike Bishop's Self-Tracker which relies on intensity changes in one-dimensional images as its primary method of tracking, my system would use natural landmarks found in two-dimensional images of the environment as secondary information. The observed motion of natural landmarks in the environment would be used to aid the INS at low frequencies, while the primary INS would provide continuous high frequency (low latency) information.

Previous work by Azuma and Bishop [Azuma94] at the University of North Carolina at Chapel Hill explored the use of inertia sensing devices to aid an optoelectronic ceiling tracker. Again in contrast, I am proposing reversed roles whereby an inertial navigation system (INS) provides the primary means of tracking, and a computer vision system provides assistance. Additionally while the optoelectronic ceiling tracker requires the placement of light emitting devices throughout the working environment, my proposed self-contained system would instead optically control INS error by observing (looking out at) at a completely unmodified environment in a passive manner.

Foxlin [Foxlin93] implemented an orientation-only system that employs inertia sensing devices aided by inclinometers and flux-gate compasses. While significant results were demonstrated, his system provides estimates of orientation only. In addition, his INS aid is available only during certain presumed randomly occurring pauses in user motion. In contrast, the computer vision portion of the proposed hybrid Self-Tracker would provide both orientation and position aid to the INS, at regularly timed intervals.

Gillis [Gillis91] proposed and simulated a system for real-time estimation of angular motion only. In his system, the outputs from nine linear accelerometers and three orthogonal gyroscopes are ensemble-averaged, with statistics being collected in the process. These statistics are then used by the estimator to compute an estimate for the 3-D angular velocity of a rigid body. This proposed system, using a linear Gaussian estimator (not unlike a Kalman filter), offered estimates of angular velocities only. In his system, low-frequency results were improved by ensemble averaging the measurements from multiple inertial devices.

In addition to providing six degrees of freedom, robustness, a wide range of motion, and a reduced latency not found in previous work, my proposed hybrid (the INS in particular) tracker offers an inherent basis for the motion predictions shown to be necessary by Azuma [Azuma95] to reduce the side-effects of overall (non-tracker) system latencies.

3. Proposed Implementation

3.1 System Design

I propose using a slightly modified Kalman filter to appropriately weight the redundant information from the INS and the CVS. A classical application of the Kalman filter is embodied in an aided inertial navigation system [Brown92]. In such a system the Kalman filter can be tuned to weight each subsystem most heavily in the region of the frequency spectrum where it provides the most reliable data, and conversely to suppress each subsystem where it is most prone to errors. In my modified Kalman filter the low frequency and velocity errors inherent in the INS can be controlled by the better performing CVS, while at higher frequencies and velocities the unreliable CVS data can be down-played or ignored in favor of the INS data. In addition, the INS data can be used by the CVS to single-out and subsequently ignore unrelated motion in the environment.

Specifically I propose using a Kalman filter in an *indirect feedback* configuration [Maybeck79] as shown below in Figure 10 to combine the navigation information provided by each subsystem in order to obtain results that are more accurate than the results obtained by either subsystem alone. In such a configuration, the Kalman filter estimates the difference between the current INS and CVS outputs, i.e. it continually estimates the *error* in the INS by using the CVS as a second (redundant) reference for position and orientation. This error estimate is then used to correct the INS. The tuning of the Kalman filter parameters (discussed briefly below) then adjusts the weight of the correction as a function of frequency. By slightly modifying the Kalman filter, adaptive velocity response can be incorporated also. This can be accomplished by adjusting (in real time) the expected CVS measurement error (the measurement error covariance) as a function of the magnitude of velocity. The dashed line in Figure 10 indicates the use of INS estimates by the CVS to prevent tracking of moving scene objects (i.e. unrelated motion in the environment).

In particular, the indirect or *error-state* implementation is motivated by four factors. First the complementary frequency and velocity behavior of the INS and CVS. As is the case in the classical INS aided system, we wish to weight the INS information heavily in during motion where it is most accurate, and to suppress it where it is prone to error. By estimating the INS error (as opposed to the total state) with a heavily filtered CVS, we can nicely implement such blending.

The second motivating factor is that the Kalman filter is based on the assumed validity of a linear system model. Because INS noise (bias and drift error) is typically low frequency, the use of a Kalman filter to model the error is more reasonable than modeling INS signals which are nonlinear and (potentially) of high frequency.

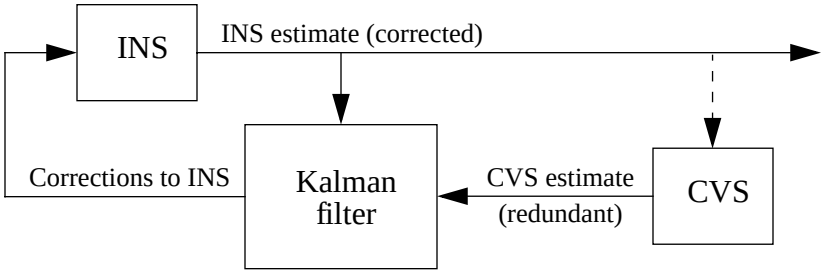

Figure 10. Indirect (error state space) feedback configuration

The third motivating factor is that by modeling only the error in the INS data, we avoid the necessity of a linear model for human motion. While such a model can (must) be used if for example one intends to predict user motion as shown to be necessary by Azuma [Azuma94], it is not an inherently necessary component of the tracking system.

Finally, an implementation such as that shown in Figure 10 provides a method of filtering the undesired noise without distorting the desired signal. In particular, note that the INS signal is not directly filtered in any way, thus it is not delayed or band-limited. In an implementation that attempts to directly filter the navigation signal, some signal distortion will always be incurred as a result of the delay caused by the filter.

The use of the INS data to assist the CVS in discerning unrelated motion in the environment is motivated by the fact that the INS is physically unaffected by such independent motion. Also, because such motion is usually *relatively* high frequency or velocity in nature, the INS can be relied upon to ignore such motion when seen by the CVS.

Following the derivations in Maybeck [Maybeck79] we arrive at the following continuous-time state equations for a one-dimensional position-only (linear accelerometer) tracker

$$\begin{bmatrix} \dot{\hat{X}}_{INS} \\ \ddot{\hat{X}}_{INS} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{X}_{INS} \\ \dot{\hat{X}}_{INS} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \left[a(t) + \varepsilon_{a(t)} \right] - \begin{bmatrix} K_1(t) \\ K_2(t) \end{bmatrix} \left[\hat{X}_{INS} - \hat{X}_{ONS} \right] \tag{6}$$

where the last term represents the Kalman filter's error estimates being used as feedback to correct the INS estimate of position.

As shown by Maybeck, if the true acceleration is corrupted by white noise ε_{ax} (see Figure 7 on page 8) and the CVS readings are corrupted by white noise ε_{ox}, the steady-state covariance matrix P and the Kalman gain matrix K become

$$P = \begin{bmatrix} \sqrt{2}A^{1/4}O^{3/4} & A^{1/2}O^{1/2} \\ A^{1/2}O^{1/2} & \sqrt{2}A^{3/4}O^{1/4} \end{bmatrix} \tag{7}$$

$$K = \begin{bmatrix} K_1 \\ K_2 \end{bmatrix} = \begin{bmatrix} P_{11}/O \\ P_{12}/O \end{bmatrix} = \begin{bmatrix} \sqrt{2}\omega_n \\ \omega_n^2 \end{bmatrix} \tag{8}$$

where

$$E[\varepsilon_{ax}(t)\varepsilon_{ax}(t+\tau)] = A\delta(\tau) \tag{9}$$

$$E[\varepsilon_{ox}(t)\varepsilon_{ox}(t+\tau)] = O\delta(\tau) \tag{10}$$

and

$$\omega_n = \left(\frac{A}{O}\right)^{1/4} \tag{11}$$

in meters per second. The motivation for (11) is to emphasize the fact that in the steady-state, this Kalman filter becomes a second order system with an undamped natural frequency of ω_n. In the steady state, the Kalman filter acts as a low-pass filter on the CVS, rolling-off the CVS feedback to the INS at frequencies above ω_n.

The effect in our simulated tracking system should be that the tracker's position estimates follow the heavily filtered CVS more closely at lower frequencies or slow movements, while following the accelerometer-driven INS estimates more closely, i.e. attenuating the CVS corrections, at higher frequencies or faster movements. The resulting system will have a frequency response from zero to $1/(2dt)$ where dt is the INS inter-sample time, and a latency of dt.

The complete derivation and study of the above equations and statements will not be repeated here. See Maybeck for a complete derivation of the above equations and the indirect-feedback filter configuration, as well as some detailed discussion of the frequency response of the Kalman filter.

A block diagram of the continuous-time version of the indirect-feedback system mathematically represented by (6) is shown below in Figure 11. In this continuous differential model, it can be seen that the change in position and velocity is being corrected by the Kalman filter's estimate of the INS error. In the actual implementation, the velocity error is integrated as a part of the normal (inherent) Kalman filter operation so that the actual correction is performed on the position and orientation estimates rather than their derivatives.

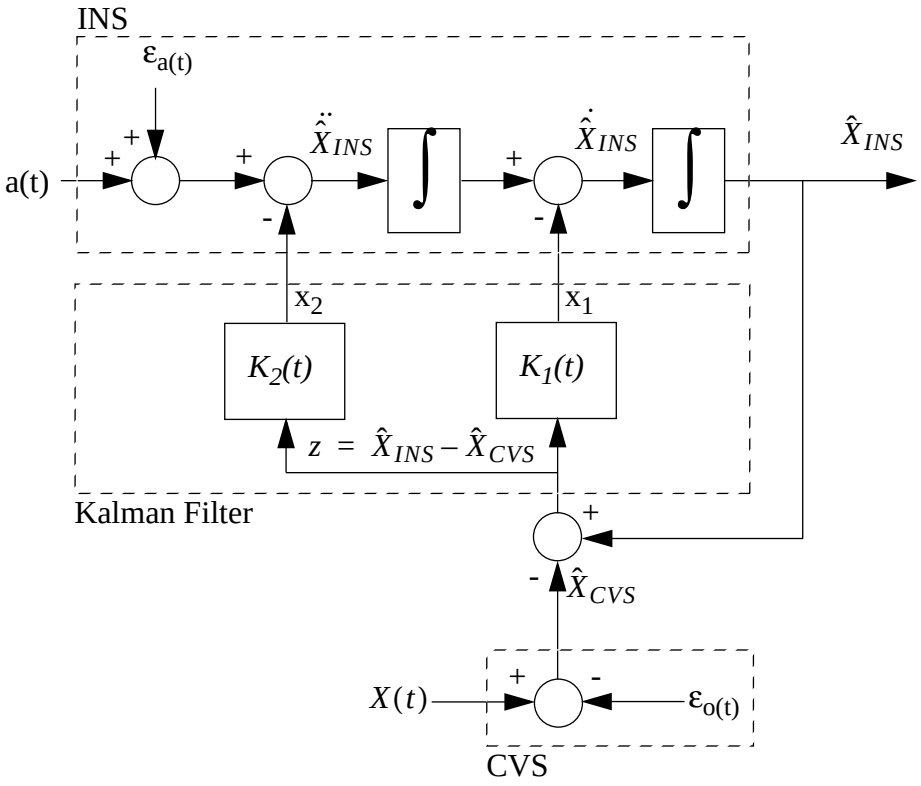

Figure 11. Indirect feedback filter configuration

If necessary, mechanical misalignment errors (of the inertia sensing devices) can be modeled in the main Kalman filter also. A complete accelerometer error model is given in [Maybeck79].

3.2 Inertial Navigation System (INS)

The INS subsystem is potentially the simplest portion of the overall system. The basic function of the INS would simply be to continually integrate (sum) inertial device measurements in order to obtain a continuous estimate of position and orientation. In the process the INS estimates would be corrected by subtracting the INS error estimates (estimated by a Kalman filter).

This process would essentially proceed as rapidly as measurements could be obtained. As indicated previously, the INS (hence the hybrid tracker) will have a frequency response from zero to $1/(2dt)$ where *dt* is the INS inter-sample time, and a latency of *dt*.

3.3 Computer Vision System (CVS)

The implementation of the CVS subsystem will be significantly more complicated in terms of both hardware and software, and is as of yet the CVS has not been completely specified.

In general, per Bishop's "spiral of goodness" argument [Bishop84] we want the CVS to operate simply and hence rapidly. However, in this hybrid system we want the CVS to provide a *completely independent* and *absolute* estimate of the 3-D current position and orientation of the tracker, not simply a relative estimate of translation. In order to more directly facilitate such independent absolute CVS estimates, I am proposing to use 2-D image sensors as opposed to 1-D sensors. Although 2-D processing will take longer (the algorithms will inevitably be more complex than their 1-D counterparts) I believe that the 2-D data is both necessary for the hybrid and implementable as described later below.

Papers describing the 3-D motion information contained in 2-D optic flow data include work by Prazdny [Prazdny83], and Longuet-Higgins Prazdny [Longuet-Higgins80]. A proposal for determining optic flow data was presented by Horn and Schunck [Horn81].

Recently several authors have proposed methods for estimating 3-D "motion" from time-sequential 2-D images [Fermüller93][Irani93][Lawn94]. These proposed methods are nice in that they are relatively robust and can be implemented with a single camera. In particular, the method proposed by Irani [Irani93] et al. might be considered attractive for the tracker's CVS as it temporally integrates spatially registered images. In addition to making the motion estimation more robust, this increases the accuracy by reducing sensitivity to sampling noise.

Each of these methods however can only provide information on five degrees of 3-D motion, three parameters describing rotation and two parameters describing the *direction* of translation—the focus of expansion or FOE on the image plane. Because the FOE estimate provides the only translation information in time-sequential images, and because the FOE only indicates the *direction* of translation (on the image plane), such methods will not be sufficient.

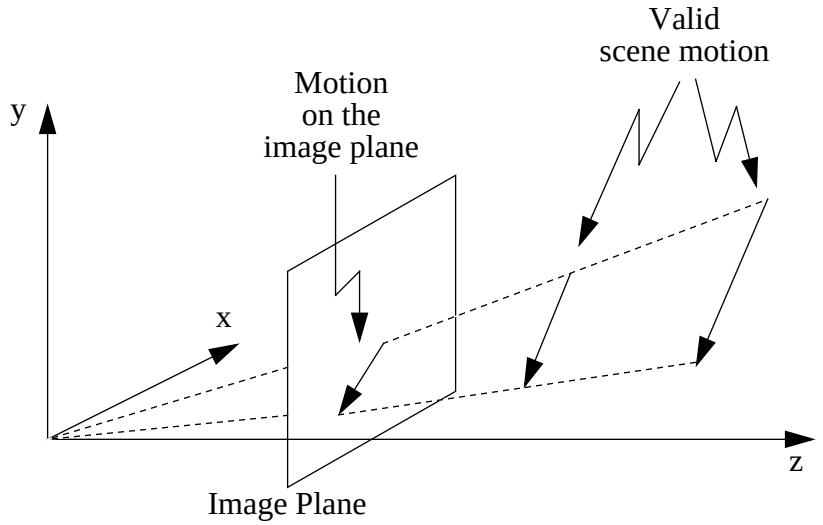

Figure 12. Translation ambiguity in single camera images [Nalwa93]

The problem is depicted above in Figure 12. For a given camera translation, the component of the motion field (on the image plane) that is due to that translation is invariant under equal scaling of the true translational-motion vector component of the scene motion. As a result, translational-motion vectors on the image plane can be explained by an infinite set of real-world 3-D scene mo-

tion vectors. Without knowing the distance to the actual scene points, the magnitude of the translation cannot be determined. This is not the case for rotation—measurements of rotation are not dependent on the spatial position of a scene points, only translation [Nalwa93].

As stated earlier, in this hybrid system we want the CVS to provide an *absolute* redundant estimate of the 3-D current position and orientation of the tracker. For this reason, information about the *magnitude* of the translation of *scene* points is essential. As a result of dependency I believe that it will be necessary to directly compute the scene depth of some selected scene features or *landmarks* in order to make direct measurements of spatial position translation. Such direct computation will require the implementation of one or more stereoscopic imaging systems. Having established stereo correspondence of a sparse number of feature points, triangulation techniques can be used to determine the complete relative 3D position of scene points. Such triangulation techniques (with error analysis) are presented by Roberts and Ganapathy in their 1986 AT&T Bell Labs paper [Roberts86]. In addition, Adelson and Wang present ideas for obtaining single lens stereo using a plenoptic camera. Such a camera is purported to have higher reliability than its two-camera alternative, and would seem to allow a more compact implementation [Adelson92].

While several locally cooperative and parallel mechanisms for establishing *overall* image correspondence have been proposed, for example [Marr76], it is important to note that such dense depth estimates should not be necessary. Instead, stereo image correspondence should be necessary only at sparsely selected image features or landmarks. Once the initial stereo correspondence for the selected features has been accomplished, the pairs of corresponding stereo image landmarks can be “locked on to” and tracked as they appear and disappear from the view of the imaging devices. A relatively simple serial method for finding correspondence between previously selected features in a pair of images is suggested in [Ballard82].

Related work has been undertaken at Carnegie Mellon University in cooperation with the Wright Research and Development Center (U.S.A.F. Wright-Patterson AFB, Dayton, OH). As a part of this work they have published a series of papers (technical reports) detailing algorithms used to recover camera motion (again only 5 DOF) in an image stream. In [Tomasi91] the authors describe and document a robust method for detecting and tracking point features in a scene. Intuitively this is very close to what we are looking for the CVS—a method to “lock on to” landmarks in the scene—although we would want to track 3-D landmark motion. A separate Kalman filter could be used to reduce sensitivity to noise in the CVS, and the CVS could use the current INS estimates to help avoid tracking moving scene objects (dashed line in Figure 10 on page 11).

When actually implementing the CVS, it might be possible to fabricate the CVS completely (or almost completely) in silicon. Using such techniques, an array of image sensors can be combined with general purpose circuitry in a standard CMOS process. For example, Anders Åström presents work with both linear and 2D *Smart Image Sensors* in his Ph.D. dissertation [Åström93], work that is now being pursued commercially by Metolius, Inc. Recently Dickinson et al. presented new techniques for fabricating active image sensors in a standard CMOS process [Dickenson95]. One benefit of their approach is that access to the image sensor array can be accomplished in a random-access fashion, i.e. it is not necessary to read the entire array. Such flexibility might very well have its place in tracking a limited number of features in the image plane, providing a potentially significant speedup.

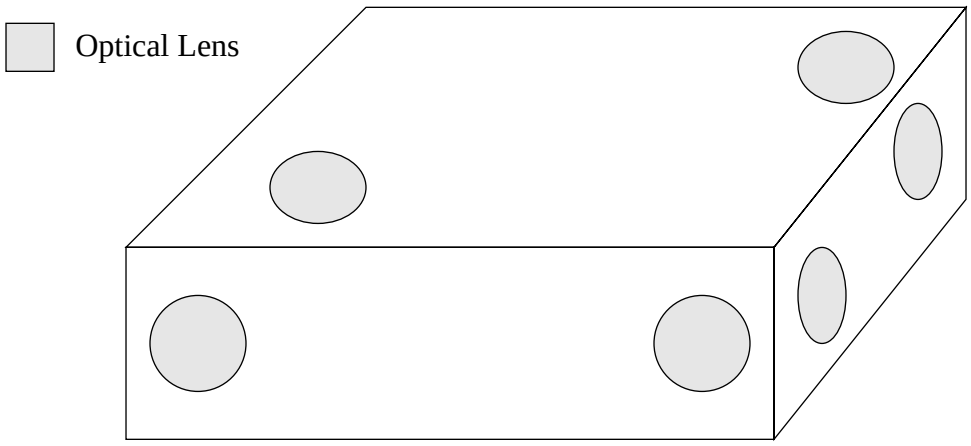

Figure 13. Mock-up of Tracker Cluster

Figure 13 depicts a rough mock-up of what a tracking cluster might look like. Imaging devices and lenses could be constructed as shown below in Figure 14, using a “periscope” lens system to extend the focal length as needed. Inertial devices could be buried inside the cluster.

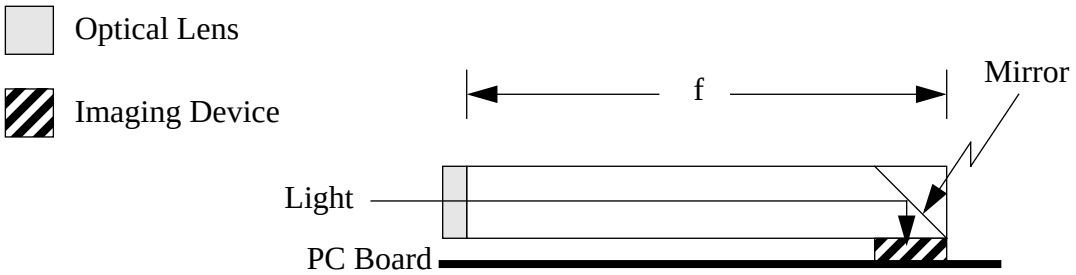

Figure 14. Periscope lens system to extend focal length

4. Research Plan and Schedule

4.1 Thesis Demonstration

My thesis would be confirmed by showing that the proposed hybrid offers better[1] position and orientation tracking than either a strictly INS or a strictly CVS self-tracker. The preferred option for demonstration is to *build* in hardware and software a complete working prototype of an INS/CVS hybrid self-tracker. Less glamorous but sufficient alternative options are also available in the event that unforseen complicating circumstances arise, e.g. the unforseen need for custom image processing hardware.

4.1.1 Preferred Demonstration Option

The preferred option for demonstrating my thesis will involve several major steps along the way—these steps are listed below. The steps are followed by some discussion of less glamorous but sufficient alternative options that would be considered in the event of unforseen complicating circumstances.

(1) Design and construction of a mechanical test rig, complete with two cameras (a stereo setup), angular rate gyros, and linear accelerometers. The rig should also offer a method for connection to a mechanical tracker such as the Faro arm (for "truth" measurements).

(2) Collect a series of camera, gyro, accelerometer, and truth samples while moving the test rig through some test motion sequences. Include some sequences with independent motion in the environment.

(3) Implement an INS model that uses the gyro and accelerometer data to estimate position and orientation. Simulate and evaluate the INS using the collected samples.

(4) Choose and implement a known stereo correspondence algorithm for sparse feature tracking by the CVS model.

(5) Implement a complete CVS model that uses the above correspondence algorithm (step 4) to estimate position and orientation. Simulate and evaluate the CVS using the collected samples.

(6) Based on the observed performace of the INS and CVS models, determine the initial parameters for the indirect feedback Kalman filter implementation, including the adaptive velocity response discussed in section 3.1 on page 11.

(7) Implement a complete software model (an off-line simulator) of the INS/CVS hybrid using the parameters discussed above in step 6. Simulate and evaluate the hybrid using the collected samples.

(8) Optimize the simulator and convert it to a real-time implementation of the hybrid tracker using the test jig assembled in step 1.

[1.] Better quantitatively in terms of mean-squared-error and peak-error during motion sequences that exhibit a broad range of frequency and velocity characteristics. Better qualitatively in terms of reliability or stability in the presence of independent motion in the tracking environment.

4.1.2 Alternative CVS Option

In the event that steps 4 and 5 under the preferred option become unreasonably problematic, e.g. they begin to resemble dissertation efforts in their own right, it would be sufficient to approximate the CVS by using the existing optoelectronic ceiling tracker at UNC. In this event, the ceiling data would be preturbed or distorted in a manner that results in data with characteristics resembling that of a passive CVS as described in section 3.3 on page 14. Steps 6, 7, and possibly 8 could still be taken in order to demonstrate the thesis, albeit in a less desirable fashion.

4.1.3 Off-Line Simulation Option

Regardless of the work performed for steps 4 and 5 of the preferred option, if step 8 appears to be unrealizable in a reasonable amount of time, the off-line simulation described in step 7 will suffice to demonstrate my thesis.

4.2 Completion Criteria

My dissertation will be considered complete after meeting each of two major milestones. First, one or more of the three thesis demonstration options (section 4.1) must be completed. In the event that the results of step 7 seem to contradict my thesis, and I am able to formulate a reasonable explanation for why this is so, these results will suffice to support a modified thesis which contradicts the original thesis. Second, all of the normal dissertation criteria must be met, e.g. my defense and complete documentation of my work.

4.3 Research Topic Areas

3D Passive Tracking

Stochastic Control Theory

Inertial Navigation

Computer Vision Based Navigation

4.4 Proposed Schedule

References

Adelson92	Adelson, E.H. and J.Y.A. Yang. Single Lens Stereo with a Plenoptic Camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 2* (February, 1992)
Åström93	Åström, A. Smart Image Sensors. Ph.D. Dissertation, Linköping University, Linköping, Sweden (1993).
Azuma94	Azuma, R. and G. Bishop. Improving Static and Dynamic Registration in an Optical See-through HMD (1994).
Azuma95	Azuma, R. Predictive Tracking for Augmented Reality. Ph.D. dissertation, The University of North Carolina at Chapel Hill, TR95-007 (1995).
Ballard82	Ballard, D.H. and Brown, C.M. Computer Vision. Prentice Hall, Inc. (1982), pp. 207-210.
Bishop82	Bishop, G. and H. Fuchs. Self-Tracker: A VLSI-based Three-dimensional Input System. Dissertation proposal for G. Bishop, The University of North Carolina at Chapel Hill (1982).
Bishop84	Bishop, G. Self-Tracker: A Smart Optical Sensor on Silicon. Ph.D. dissertation, The University of North Carolina at Chapel Hill (1984).
Brown92	Brown, R.G. and P.Y.C. Hwang. Introduction to Random Signals and Applied Kalman Filtering, Second Edition. John Wiley & Sons, Inc. (1992).
Dickenson95	Dickinson, A., B. Ackland, E. Eid, D. Inglis and E. Fossum. Standard CMOS Active Pixel Image Sensors for Multimedia Applications. *IEEE 16th Conference on Advanced Research in VLSI Proceedings* (1995), pp. 214-224.
Fermüller93	Fermüller, C. Global 3-D Motion Estimation. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Proceedings* (1993).
Foxlin93	Foxlin, E. Inertial Head-Tracking. M.S. Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology (September 1993).
Gillis91	Gillis, J.T. Estimation of 3-D Angular Motion Using Gyroscopes and Linear Accelerometers. *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 27, No. 6 (1991).
Horn81	Horn, B.K.P. and B.G. Schunck. Determining Optical Flow. *Artificial Intelligence*, Vol. 17 (1981).
Irani93	Irani, M., B. Rousso and S. Peleg. Robust Recovery of Ego-Motion. Computer Analysis of Images and Patterns, 5th International Conference, *CAIP '93 Proceedings* (1993).
Lawn94	Lawn, J. and R. Cipolla. Robust Egomotion Estimation from Affine Motion Parallax. Third European Conference on Computer Vision, *Computer Vision—ECCV '94* (1994).
Longuet-Higgins80	Longuet-Higgins, H.C. and K. Prazdny. The Interpretation of a Moving Retinal Image. *Proceedings of Royal Society of London*, Vol. 208. B (1980).

Matthies86	Matthies, L. and S.A. Shafer. Error Modeling in Stereo Navigation. ACM/ IEEE Fall Joint Computer Conference, Dallas, TX (November 5, 1986).
Marr76	Marr, D. and T. Poggio. Cooperative Computation of Stereo Disparity. *Science*, Vol. 194 (1976).
Maybeck79	Maybeck, P.S. Stochastic Models, Estimation, and Control, Volume 1. Academic Press, Inc. (1979).
Meyer92	Meyer, K., H. Applewhite and F. Biocca. A Survey of Position Trackers. *Presence*, a publication of the *Center for Research in Journalism and Mass Communication*, The University of North Carolina at Chapel Hill (1992).
Nalwa93	Nalwa, V.S. A Guided Tour of Computer Vision. Addison-Wesley Publishing Company (1993).
Prazdny83	Prazdny, K. On the Information in Optical Flows. *Computer Vision, Graphics, and Image Processing*, Vol. 22 (1983).
Roberts86	Roberts, K. and S. Ganapathy. Stereo Triangulation Techniques. AT&T Bell Laboratories Technical Memorandum, Charge Case 311306-0399, File Case 39394 (November 21, 1986)
Sutherland68	Sutherland, I.E. A Head-Mounted Three Dimensional Display, *Fall Joint Computer Conference, AFIPS Conference Proceedings 33* (1968), pp. 757-764.
Tomasi91	Tomasi, C. and T. Kanade. Detection and Tracking of Point Features. Carnegie Mellon University, Technical Report CMU-CS-91-132 (1991).