Predicting Accuracy in Pose Estimation for Marker-based Tracking

Larry Davis
University of Central Florida
Orlando, FL 32816-2700, USA
davis@odalab.ucf.edu

Eric Clarkson
University of Arizona
Tuscon, AZ, USA
clarkson@radiology.arizona.edu

Jannick P. Rolland
University of Central Florida
Orlando, FL 32816-2700, USA
jannick@odalab.ucf.edu

Abstract

Tracking is a necessity for interactive virtual environments. Marker-based tracking solutions involve the placement of fiducials in a rigid configuration on the object(s) to be tracked, called a tracking probe. The realization that tracking performance is linked to probe performance necessitates investigation into the design of tracking probes for proponents of marker-based tracking. A challenge involved with probe design is predicting the accuracy of a tracking probe. We present a method for predicting the accuracy of a tracking probe based upon a first-order propagation of the errors associated with the markers on the probe. Results for two sample tracking probes show excellent agreement between measured and predicted errors.

1 Introduction

A significant challenge in virtual environments is tracking objects accurately and precisely within the environment. The need for accurate and precise tracking is accentuated when considering augmented reality (AR) environments, which require registration of real and virtual objects. While there are many tracking techniques that are currently used within virtual environments, marker-based tracking solutions can be used to provide fast, accurate tracking. In this approach, features upon the tracked object(s) are identified by the tracker. These features, or markers, may be intrinsic to the object tracked or may be fiducials placed upon the object. The collection of markers is what we refer to as a tracking probe. When experiencing marker-based tracking, one observes that the accuracy of a pose estimate from a given tracking probe is dependent upon the probe topology as well as the performance of the overall tracking system. The tracking probe topology includes the number of markers utilized and the location of the markers with respect to the probe origin. In seeking to overcome environmental constraints (e.g. line of sight issues, probe size) tracking probes designs do not always meet desired performance levels.

It would be beneficial for practitioners to be able to predict the accuracy of a given probe topology beforehand, and to adjust the marker locations and number of markers utilized accordingly. The research presented provides a starting point for designing marker-based tracking probes by providing a method to predict the performance of pose estimation for a given tracking probe. This method is based upon applying an error term to each marker location and propagating this error through the resulting estimation of pose within a first-order approximation. All results will be expressed in terms of visual space (i.e., distances instead of pixels). The research presented does not include systems that provide only orientation data because such systems often do not require more than one marker. Moreover, we will currently examine discrete changes, or "snapshots", of a tracking probe given that accuracy in stationary pose estimation, through discrete changes in position and orientation, is a necessary condition for dynamic accuracy.

2 Previous Research in Pose Error Determination

There is a significant body of work that exists on quantifying the different types of errors present within tracking systems and their effects on pose determination. We limit the scope of our survey of this area to research whose aim is determining errors in pose determination related to the topology of marker distributions.

Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)
0-7695-2006-5/03 $17.00 © 2003 IEEE

Along this line, Woltring et al examined the effects of marker errors on pose estimation and provided a maximum error statistic to predict the pose error of a given tracking probe topology [12]. The error statistic derived used first-order errors within its error computation. This approach dealt strictly with the case when the markers were symmetrically distributed with respect to the probe origin. In addition, the probe origin was coincident with the marker centroid (e.g. a tetrahedron or a cube).

In an extension of the work in [12], Morris & Donath quantified the cumulative effects of multiple error sources, including the effects of algorithmic errors and dynamic target array deformation errors [6]. Based upon the work presented in [12], a modified maximum error statistic was presented for determining the pose error for a given tracking probe topology. The research was valid for marker distributions that lay on a sphere and had the center of the sphere as the probe origin (e.g. a square, pyramid, tetrahedron, etc.) .

In a context related to virtual environments, Vogt et al implemented a method for designing tracking probes using a Monte Carlo simulation technique [11]. The design methodology minimized the jitter error associated with the tracking probe, using the probe radius, marker heights, and number of markers as input variables. Because Tsai's calibration technique [10] was utilized to determine the pose of the tracking probe, the method required probe topologies with at least seven, simultaneously detected markers.

The approach presented in this work expands upon the previously mentioned efforts by presenting a theoretical framework for determining pose error that is valid for any probe topology with at least three, non-collinear markers. Moreover, the error sources for the location of each marker are considered as an ensemble, and the error on each marker considered is thus a combination of jitter, tracker bias, probe deformations, and the probe topology. Combined with previous work regarding methods for marker placement [2] [3], this research represents a starting point for a general framework for designing marker based tracking probes.

3 Modeling the Impact of Noise on Pose Estimation

The data obtained from a tracking system are noisy by nature. Furthermore, tracking probes are subject to slight errors in manufacturing and pose errors due to asymmetry. We use a first-order error propagation model to determine how the noise affects, or propagates to, the error in pose. This means taking each marker location, adding a differential error vector based upon the system properties, and determining the resulting mathematical relationships based upon first order approximations. A method based upon error propagation is utilized because it facilitates quantification of the effects of individual markers on the pose estimation process. Results will be presented in Section 4.1 that show that a first-order error propagation model is appropriate, provided that the errors in marker position are at least an order of magnitude smaller than the scale of the tracking probe.

3.1 Propagating Marker Error

Given a set of K markers on a tracking probe, we denote their positions within a local coordinate frame as x_k and their positions within a global coordinate frame (coincident with the tracker) as y_k. We can then express the relationship between x_k and y_k as

$$y_k = Rx_k + T \ , \tag{1}$$

where R is a 3x3 rotation matrix, T is a 3x1 translation vector, and y_k and x_k are 3x1 column vectors. R and T can be determined using the pose determination method proposed by Argotti et al [1], which is built upon methods presented in [8] and [4]. These pose estimation algorithms are based upon minimization of the least squares error and are reviewed briefly in Appendix A.1.

We first determine the local marker coordinate error, Δx_k, and global marker coordinate error, Δy_k and propagate these errors to the local and global marker coordinates as

$$\begin{aligned} x_k &\rightarrow x_k + \Delta x_k \\ y_k &\rightarrow y_k + \Delta y_k \ . \end{aligned} \tag{2}$$

We then define the errors applied to each marker with respect to the centroid of the errors in the local and global coordinates, $\Delta\widetilde{x_k}$ and $\Delta\widetilde{y_k}$, as

$$\begin{aligned} \Delta\widetilde{x_k} &= \Delta x_k - \frac{1}{K}\sum_{k=1}^{K}\Delta x_k \\ \Delta\widetilde{y_k} &= \Delta y_k - \frac{1}{K}\sum_{k=1}^{K}\Delta y_k \ . \end{aligned} \tag{3}$$

Provided that the "ideal" local and global marker locations with respect to the centroids, $\widetilde{x_k}$ and $\widetilde{y_k}$, are computed as

$$\begin{aligned} \widetilde{x_k} &= x_k - \frac{1}{K}\sum_{k=1}^{K}x_k \\ \widetilde{y_k} &= y_k - \frac{1}{K}\sum_{k=1}^{K}y_k \ , \end{aligned} \tag{4}$$

we then propagate the errors to the local and global coordinates with respect to the marker centroids, as

$$\begin{aligned} \widetilde{x_k} &\rightarrow \widetilde{x_k} + \Delta\widetilde{x_k} \\ \widetilde{y_k} &\rightarrow \widetilde{y_k} + \Delta\widetilde{y_k} \ . \end{aligned} \tag{5}$$

Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)
0-7695-2006-5/03 $17.00 © 2003 IEEE

Normally, errors are implicit within a pose estimation algorithm. That is, the x_k and y_k used as input to the error minimization (see Appendix A.1) are noisy. However, in propagating errors through the pose estimation, we define the error for each marker location explicitly, rather than implicitly, within the pose estimation procedure. Thus, the "ideal" x_k and y_k must be obtained *a priori* along with Δx_k and Δy_k. x_k and Δx_k can be obtained from the manufacturer's data (if using a commercially available tracking probe) or from the tracking probe design process. y_k can be obtained by applying a transformation to the ideal x_k which approximates the probe pose with respect to the tracker where the probe is most likely to be used. Δy_k can be obtained by first measuring the RMS error in position of a single stationary marker, $\sigma_{tracker}$ with respect to the tracker, then applying gaussian noise with $\sigma = \sigma_{tracker}$ to each y_k.

3.2 Expressing Pose Error

We now determine how the errors in marker location propagate to errors in pose estimation. An error in pose can be represented as a differential transform from the true pose. Given a true rotation and translation, R and T, and differential rotation and translation ΔR and ΔT, we can express the overall pose, with error included as

$$\begin{aligned} R_{err} &\approx \Delta R R \\ T_{err} &\approx T + \Delta T \ , \end{aligned} \tag{6}$$

where an equation of the form $a \approx b$ is taken to mean that a and b are approximately equal to a first order approximation.

From Appendix A.1, we know that $R = VDU^T$, with U and V resulting from the SVD of H. Therefore, to determine ΔR and ΔT, we must determine how the marker errors affect the matrix H, which in turn is affected by U and V. Therefore, we propagate the error to R through the matrices V and U by

$$\begin{aligned} V &\rightarrow V + \Delta V \\ U &\rightarrow U + \Delta U \ . \end{aligned} \tag{7}$$

The differential errors applied to U and V can be defined as a transformed version of the original U and V matrices. To represent the transformations, we define two matrices A and B such that

$$\begin{aligned} \Delta V &\approx AV \\ \Delta U &\approx BU \ . \end{aligned} \tag{8}$$

The procedure for determining A and B is given in Appendix A.2, where it can be noted that matrices A and B are anti-symmetric ($A = -A^T$). Given V, U, and D, which are all determined from the marker data and the errors associated with each marker, R_err can be expressed as

$$\begin{aligned} R_{err} &\approx \Delta R R \\ &\approx (V + \Delta V) D (U + \Delta U)^T \\ &\approx VDU^T + VDU^T B^T + AVDU^T + AVDU^T B^T \\ &\approx VDU^T - VDU^T B + AVDU^T - AVDU^T B \\ &\approx (I + A) VDU^T (I - B) \\ &\approx (I + A) R (I - B) \ . \end{aligned} \tag{9}$$

Taking into account the fact that a matrix exponential can be approximated by a Taylor series expansion as

$$e^A \cong 1 + A + \frac{1}{2}A^2 + \frac{1}{6}A^3 + \dots \ , \tag{10}$$

a first order approximation of R_{err} is ,

$$\begin{aligned} R_{err} &\approx e^A R e^{-B} \\ &\approx e^A R e^{-B} I \\ &\approx e^A R e^{-B} R^T R \\ &\approx \underbrace{e^A R e^{-B} R^T}_{\Delta R} R \ . \end{aligned} \tag{11}$$

Thus, we can compute the amount of rotational error, ΔR, introduced into a pose calculation by noise in the marker data as

$$\Delta R = e^A R e^{-B} R^T \ . \tag{12}$$

A matrix exponential is used in Eq. 12 because using the approximate expressions of $(I + A)$ or $(I - B)$ may yield an invalid rotation matrix, that is $det(\Delta R) \neq \pm 1$.

If we define the centroids of the "ideal" local and global coordinates, x and y, as

$$\begin{aligned} x &= \frac{1}{K}\sum_{k=1}^{K} x_k \\ y &= \frac{1}{K}\sum_{k=1}^{K} y_k \ , \end{aligned} \tag{13}$$

and the centroids of the local and global coordinates with error, x_c and y_c, as

$$\begin{aligned} x_c &= \frac{1}{K}\sum_{k=1}^{K} (x_k + \Delta x_k) \\ y_c &= \frac{1}{K}\sum_{k=1}^{K} (y_k + \Delta y_k) \ , \end{aligned} \tag{14}$$

we can compute the translational error, ΔT, introduced into a pose calculation as

$$\begin{aligned} \Delta T &= T_{err} - T \\ &= y_c - \Delta R R x_c - y + R x \end{aligned} \tag{15}$$

Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)
0-7695-2006-5/03 $17.00 © 2003 **IEEE**

4 Simulation Results and Experimental Verification

4.1 Simulations

To test the method, we began by examining the performance of a simulated tracking probe in MATLAB. Noise in the probe data was simulated by adding Gaussian distributed noise, with a σ which was chosen according the probe characteristics.

The metric used for determining the pose error of a tracking probe was the average error for each marker on the probe. Given R_{err} and T_{err} as the predicted rotation and translation with propagated error, y_k as the k^{th} "true" marker location in the tracker coordinate frame, and x_k as the k^{th} "true" marker location in the probe frame, the average error, E may be expressed as

$$E = \frac{1}{K}\sum_{k=1}^{K} \|y_k - R_{err}x_k - T_{err}\| \ . \qquad (16)$$

The error was averaged over 1000 pose determinations in each case.

We then investigated the relationship between the pose error predicted for a tracking probe and the value of σ used for our noise propagation. The probe topology remained fixed, while σ was reduced incrementally by factors of ten. The tracking probe simulated had six markers, whose positions were defined in millimeters, while σ was reduced from a starting value of 1 mm. The resulting plot of pose error vs σ is shown in Figure 1. As expected, the error in pose increases as σ increases. Moreover, we compared the pose error computed by first order approximation to the pose error computed using the Horn's method of determining orientation using unit quaternions [5] and found that both methods provide similar results. Details of this comparison will be published at a later date.

In addition, we verify the relationship between pose error and the size of the tracking probe. Using a tracking probe with five markers, we simulated markers whose positions were initially defined in centimeters. We then uniformly scaled the size of the probe incrementally by factors of ten, while keeping the errors applied to the markers constant at σ = 0.1mm. The plot of probe size vs. pose error in millimeters is shown in Figure 2. As the size of the tracking probe is increased, we see that the pose error decreases, as expected.

The simulation results demonstrate that the first-order approximation method for predicting pose error behaves in a manner consistent to what is expected. Next, the pose error prediction was applied to actual tracking probes.

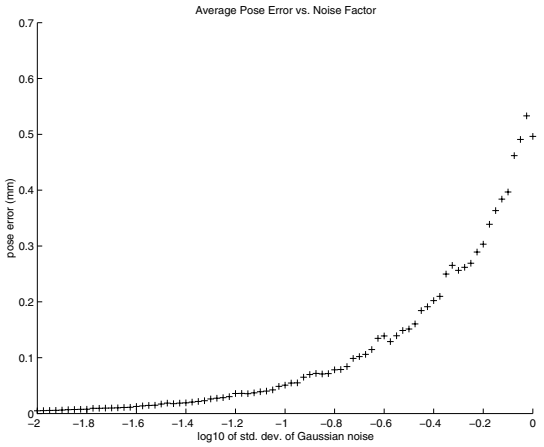

Figure 1. Pose Error vs. σ of Gaussian Noise

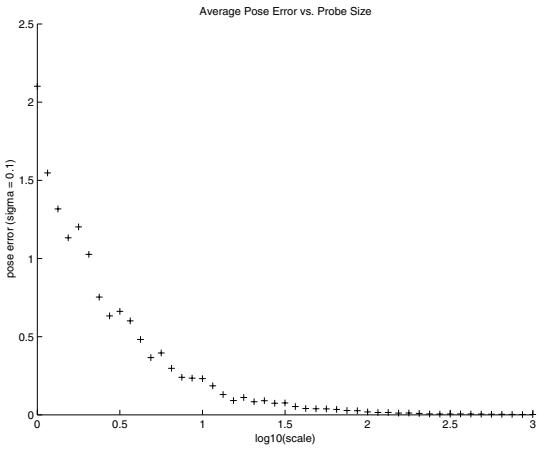

Figure 2. Pose Error vs. Probe Size

4.2 Experimental Verification

Based upon the simulations in Section 4.1, we applied the method for predicting pose error to two tracking probes in use within our laboratory. Probe 1 was a digitizing probe manufactured by Northern Digital Inc., shown in Figure 3 for use with the OPTOTRAK tracking system. Probe 2 was a head tracking probe designed in our laboratory for use with the OPTOTRAK system and fabricated via rapid prototyping. Probe 2 is shown in Figure 4.

For Probe 1, $\Delta x = 254\mu m$, given by the manufacturer, and $\Delta y = 0.15mm$, measured as the rms error in position of a single marker. The pose error predicted by the combination of Equations 12 and 16 was 0.105 mm. The actual pose error for Probe 1 was 0.1176 mm.

For Probe 2, $\Delta x = 0.25mm$, determined from the rapid prototyping data, and $\Delta y = 0.15mm$, again measured as

Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)
0-7695-2006-5/03 $17.00 © 2003 IEEE

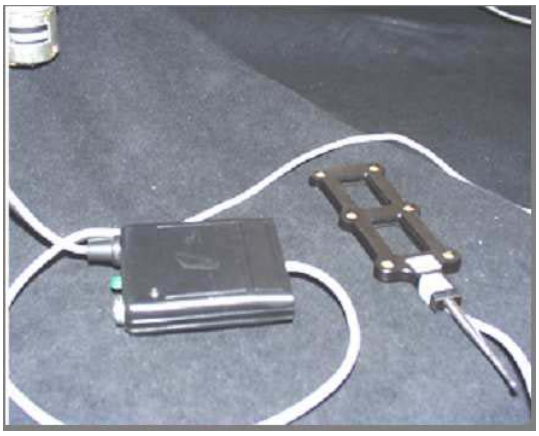

Figure 3. Probe 1: A six-marker digitizing probe

the rms error in position of a single marker. The pose error predicted for the tracking probe was 0.288 mm. The actual pose error for Probe 2 was 0.297 mm.

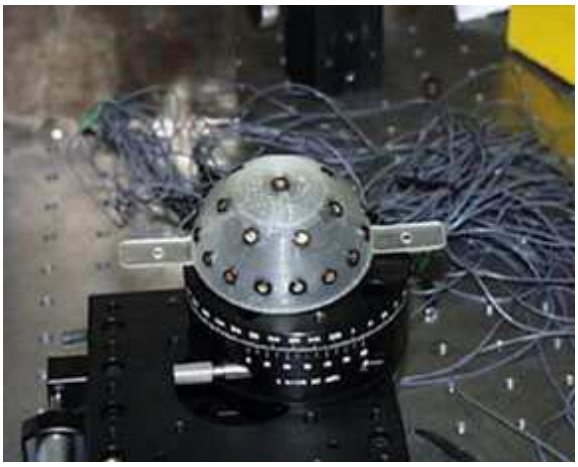

Figure 4. Probe 2: A custom designed tracking probe

5 Ongoing Work and Conclusions

While marker-based tracking has traditionally implied the use of optical tracking methods, the method presented generally applies to any tracking system that uses multiple, distinct markers to determine the pose of the tracked objects. The process presented for estimating pose error is applicable to general tracking probe topologies and robust. There are additional aspects to be considered for further improvement within the pose error estimation. For instance, the current model does not account for algorithmic error within the pose determination, which, while small, may be significant as the probe size decreases. Also, depending upon the tracker, errors may be dependent upon where the probe is located within the tracking volume. These two factors may contribute to the slight discrepancy seen between the predicted pose error and the measured pose errors.

Furthermore, the pose error prediction should include mechanisms for dynamic error estimation. As tracking probes are typically not static, the velocities and accelerations of the tracked objects should be accounted for within the error prediction. This may be facilitated by introducing motion estimation within our method. The possibility of expressing a relationship between individual marker errors and their effects on the probe pose is enticing. Thus, our future efforts will include an expansion to provide detailed quantification of the dependence of pose on the number of markers used as well as a graphical analysis of perturbations within the A and B matrices (which characterize the relationship). Similarly, an examination of the dependence of rotational errors about each axis as a function of marker error will be beneficial as well.

The research is presented as part of a larger framework for conformal tracking in virtual environments. Conformal tracking refers to the idea of tracking probes being constructed according to the shape of the objects of interest (i.e., conforming). Although the idea placing markers naturally on an object to track is not new, applying specific, quantitative design techniques to the process is. The proposed framework is general and will allow us to expand our quantification marker-based tracking performance from one based solely upon marker errors, to quantification with respect to marker placement, the number of markers, and the size of the marker distribution.

6. Acknowledgements

This research was supported by the U.S. Army Simulation, Training, and Instrumentation Command (STRICOM).

References

[1] Y. Argotti, L. Davis, V. Outters, and J. Rolland. Dynamic superimposition of synthetic objects on rigid and simple-deformable objects. *Computers and Graphics*, 26(6):919–930, 2002.

[2] L. Davis, J. Rolland, R. Parsons, and E. Clarkson. Methods for designing head-tracking probes. In *Proceedings of the Joint Conference on Information Sciences (JCIS)*, pages 498–502, March 2002.

Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)
0-7695-2006-5/03 $17.00 © 2003 **IEEE**

[3] F. Hamza-Lup, L. Davis, C. Hughes, and J. Rolland. Marker mapping techniques for augmented reality. In *Proceedings of the International Symposium on Computer and Information Sciences (ISCIS 2002)*, pages 152–156, 2002.

[4] R. M. Haralick, H. Joo, C. Lee, X. Zhuang, V. Vaidya, and M. Kim. Pose estimation from corresponding point data. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1426–1446, 1989.

[5] B. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, 1987.

[6] T. Morris and M. Donath. Using a maximum error statistic to evaluate measurement errors in 3d position and orientation tracking systems. *Presence: Teleoperators and Virtual Environments*, 2(4):314–343, 1993.

[7] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, Cambridge, MA, 2nd edition, 1992.

[8] C. Spoor and F. Veldpaus. Technical note: Rigid body motion calculated from spatial coordinates of markers. *Journal of Biomechanics*, 13:391–393, 1980.

[9] G. Strang. *Linear Algebra and its Applications*. Harcourt Brace Jovanovich, San Diego, 3rd edition, 1988.

[10] R. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323–344, 1987.

[11] S. Vogt, A. Khamene, F. Sauer, and H. Niemann. Single camera tracking of marker clusters: Multiparameter cluster optimization and experimental verification. In *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR '02)*, pages 127–136. IEEE and ACM, September 2002.

[12] H. Woltring, R. Huiskes, A. De Lange, and F. Veldpaus. Finite centroid and helical axis estimation from noisy landmark measurements in the study of human joint kinematics. *Journal of Biomechanics*, 18(5):379–389, 1985.

A Appendix

A.1 Summary of Least Squares 3D Pose Estimation

Given K markers, local marker coordinates, x_k, and global (tracker) marker coordinates, y_k, for $k = 1, \ldots, K$, we can express the following relationship

$$e(R,T) = \sum_{k=1}^{K} \parallel y_k - Rx_k - T \parallel^2 \ . \tag{17}$$

The R and T which minimize the error in Eq. 17 can be determined using the following method. If we define the centroids in the local and global coordinate frames, x and y, respectively, as

$$x = \frac{1}{K}\sum_{k=1}^{K} x_k \tag{18}$$

$$y = \frac{1}{K}\sum_{k=1}^{K} y_k \ ,$$

we can express the coordinates in each frame with respect to the centroids as

$$\begin{aligned} \widetilde{x_k} &= x_k - x \\ \widetilde{y_k} &= y_k - y \ . \end{aligned} \tag{19}$$

We can then define a matrix, H, given by

$$H = \sum_{k=1}^{K} \widetilde{x_k}\widetilde{y_k}^T \ . \tag{20}$$

The singular value decompositon (SVD) of H is expressed as

$$H = U\Gamma V^T \ . \tag{21}$$

The matrices U, Γ, and V are orthogonal (by definition of the SVD). From U and V, we can compute R as

$$R = VDU^T \ , \tag{22}$$

where

$$D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & det(VU^T) \end{bmatrix} \tag{23}$$

and $T = y - Rx$. The R and T produced minimize Eq. 17.

A.2 Determination of the A and B matrices

To determine the A and B matrices, we start by determining how the marker errors affect the matrix H. The SVD of H is defined as

$$H = U\Gamma V^T \ . \tag{24}$$

Γ is the matrix of singular values, which are the square roots of the eigenvalues of H, μ_m. Because the columns of U and V are the eigenvectors of HH^T and H^TH, respectively [9], we can express the columns of U and V with the eigenvector equations

$$\begin{aligned} H^THv_m &= \mu_m v_m \\ HH^Tu_m &= \mu_m u_m \ . \end{aligned} \tag{25}$$

Moreover, the columns of U and V are orthogonal, meaning

$$v_m{}^T v_n = u_m{}^T u_n = \delta_{mn} \tag{26}$$

and meaning that U and V are unitary. Propagating the errors to H,

$$H \rightarrow H + \Delta H \tag{27}$$

where

$$\Delta H \approx \sum_{k=1}^{K} (\Delta\widetilde{x_k}\widetilde{y_k}^T + \widetilde{x_k}\Delta\widetilde{y_k}^T) \ . \tag{28}$$

Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)
0-7695-2006-5/03 $17.00 © 2003 **IEEE**

To analyze the error propagation to H, we define a matrix Q as

$$Q = H^T H \ . \tag{29}$$

Then, with first order error propagation,

$$Q \rightarrow Q + \Delta Q \tag{30}$$

with

$$\Delta Q \approx \Delta H^T H + H^T \Delta H \tag{31}$$

which is a first order expansion of Q. Now, consider the eigenvector equations from Eq. 25. We propagate the error as

$$(Q+\Delta Q)(v_m+\Delta v_m) = (\mu_m+\Delta\mu_m)(v_m+\Delta v_m) \tag{32}$$

$$(Q^T+\Delta Q^T)(u_m+\Delta u_m) = (\mu_m+\Delta\mu_m)(u_m+\Delta u_m) \ , \tag{33}$$

which must be satisfied with the constraints

$$\begin{aligned} (v_m + \Delta v_m)^T (v_n + \Delta v_n) &= \delta_{mn} \\ (u_m + \Delta u_m)^T (u_n + \Delta u_n) &= \delta_{mn} \end{aligned} \tag{34}$$

because orthogonality must still hold, even with errors in the H matrix. Using the error-free eigenvector equations (Eqs. 25 and 26), and applying the previous constraints, Eqs. 32 and 33 can be expanded in the following manner:

$$\begin{aligned} Qv_m + \Delta Q v_m + Q\Delta v_m + \Delta Q \Delta v_m = \\ \mu_m v_m + \Delta\mu_m v_m + \mu_m \Delta v_m + \Delta\mu_m \Delta v_m \end{aligned} \tag{35}$$

$$\begin{aligned} Q^T u_m + \Delta Q^T u_m + Q^T \Delta u_m + \Delta Q^T \Delta u_m = \\ \mu_m u_m + \Delta\mu_m u_m + \mu_m \Delta u_m + \Delta\mu_m \Delta u_m \ . \end{aligned} \tag{36}$$

Because

$$\begin{aligned} Qv_m &= \mu_m v_m \\ Q^T u_m &= \mu_m u_m \\ \Delta Q \Delta v_m &\approx \Delta Q^T \Delta u_m \approx 0 \ \ and \\ \Delta\mu_m \Delta v_m &\approx \Delta\mu_m \Delta u_m \approx 0 \ , \end{aligned} \tag{37}$$

we can simplify Eq. 35 to

$$\begin{aligned} \mu_m v_m + \Delta Q v_m + Q\Delta v_m \approx \\ \mu_m v_m + \Delta\mu_m v_m + \mu_m \Delta v_m \\ \Rightarrow \Delta Q v_m + Q\Delta v_m \approx \Delta\mu_m v_m + \mu_m \Delta v_m \end{aligned} \tag{38}$$

and, likewise, we can simplify Eq. 36 to

$$\begin{aligned} \mu_m u_m + \Delta Q^T u_m + Q^T \Delta u_m \approx \\ \mu_m u_m + \Delta\mu_m u_m + \mu_m \Delta u_m \\ \Rightarrow \Delta Q^T u_m + Q^T \Delta u_m \approx \Delta\mu_m u_m + \mu_m \Delta u_m \end{aligned} \tag{39}$$

with the orthogonality constraints (derived by first order expansion of Eq. 26)

$$\begin{aligned} v_m^T \Delta v_n + v_n^T \Delta v_m &\approx 0 \\ u_m^T \Delta u_n + u_n^T \Delta u_m &\approx 0 \ . \end{aligned} \tag{40}$$

We solve this system for $\Delta\mu_m$ and Δv_m (the first order error of the eigenvalue and its associated eigenvector) in terms of ΔQ (the first order error in the matrix describing the relation between the two coordinate systems). In order to do this, we define the matrices A and B by the equations

$$\begin{aligned} \Delta v_m &= A v_m \ , \ m = 1, \ldots, M \\ \Delta u_m &= B u_m \ , \ m = 1, \ldots, M \end{aligned} \tag{41}$$

These equations define A and B uniquely since the singular vectors form basis $\mathbb{R}^M$. By substituting into Eqs. 38 and 39 with the previous equation, the system can be written in terms of these new matrices as

$$\Delta Q v_m + Q A v_m \approx \Delta\mu_m v_m + \mu_m A v_m \tag{42}$$

$$\Delta Q^T u_m + Q^T B u_m \approx \Delta\mu_m u_m + \mu_m B u_m \tag{43}$$

and the orthogonality constraints become

$$\begin{aligned} v_m^T A v_n + v_n^T A v_m &\approx 0 \\ u_m^T B u_n + u_n^T B u_m &\approx 0 \end{aligned} \tag{44}$$

By rearranging Eqs. 42 and 43, we have

$$(\mu_m - Q) A v_m \approx (\Delta Q - \Delta\mu_m) v_m \tag{45}$$

$$(\mu_m - Q^T) B u_m \approx (\Delta Q^T - \Delta\mu_m) u_m \tag{46}$$

We now take Eq. 45 and multiply through by ${v_n}^T$,

$${v_n}^T \Delta Q v_m - {v_n}^T \Delta\mu_m v_m \approx {v_n}^T \mu_m A v_m - {v_n}^T Q A v_m \tag{47}$$

Since

$$Q v_n = \mu_n v_n \Rightarrow (Q v_n)^T = (\mu_n v_n)^T \Rightarrow {v_n}^T Q^T = \mu_n {v_n}^T \ , \tag{48}$$

we can express Eq. 47 as

$${v_n}^T \Delta Q v_m - \Delta\mu_m {v_n}^T v_m \approx \mu_m {v_n}^T A v_m - \mu_n {v_n}^T A v_m \ . \tag{49}$$

Furthermore, because ${v_n}^T v_m = \delta_{mn}$,

$${v_n}^T \Delta Q v_m - \Delta\mu_m \delta_{mn} \approx (\mu_m - \mu_n) {v_n}^T A v_m \ . \tag{50}$$

When $m = n$, we get

$$\Delta\mu_m \approx {v_n}^T \Delta Q v_m \ . \tag{51}$$

When $m \neq n$, we get

$${v_n}^T A v_m \approx \frac{{v_n}^T \Delta Q v_m}{\mu_m - \mu_n} \ , \tag{52}$$

Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)
0-7695-2006-5/03 $17.00 © 2003 **IEEE**

which fully specifies A. Note that since $\Delta Q^T = \Delta Q$ (meaning Q is symmetric) , we have

$$ {v_m}^T \Delta Q v_n = {v_m}^T \Delta Q^T v_n = ({v_n}^T \Delta Q v_m)^T \ . \quad (53) $$

Moreover, because ${v_m}^T \Delta Q v_n$ is a scalar,

$$ ({v_n}^T \Delta Q v_m)^T = {v_n}^T \Delta Q v_m \ . \quad (54) $$

As a result, the orthogonality constraints on A are satisfied.

Now, we want to get everything in terms of ΔH. We start with the SVD of H

$$ H = U \Gamma V^T \ . \quad (55) $$

This implies that

$$ \begin{aligned} HV &= U\Gamma \quad (56) \\ U^T H &= \Gamma V \end{aligned} $$

leading to

$$ \begin{aligned} H v_m &= u_m \sqrt{\mu_m} = \sqrt{\mu_m} u_m \quad (57) \\ H^T u_m &= v_m \sqrt{\mu_m} = \sqrt{\mu_m} v_m \ . \end{aligned} $$

We can now develop two scenarios. The first is when $m = n$. We start with the relation from Eq. 51, expanding with the relationships from Eqs. 31 and 57:

$$ \begin{aligned} \Delta\mu_m &\approx {v_n}^T \Delta Q v_m \approx {v_m}^T \Delta Q v_m \ , m = n \quad (58) \\ &\approx {v_m}^T (\Delta H^T H + H^T \Delta H) v_m \\ &\approx {v_m}^T \Delta H^T H v_m + {v_m}^T H^T \Delta H v_m \\ &\approx {v_m}^T \Delta H^T \sqrt{\mu_m} u_m + \sqrt{\mu_m} {u_m}^T \Delta H v_m \\ &\approx \sqrt{\mu_m} ({v_m}^T \Delta H^T u_m + {u_m}^T \Delta H v_m) \\ &\approx 2\sqrt{\mu_m} {u_m}^T \Delta H v_m \ . \end{aligned} $$

The second scenario is when $m \neq n$. Staring with the relation from Eq. 52, expanding with the relationships from Eqs. 31 and 57:

$$ \begin{aligned} {v_n}^T A v_m &\approx \frac{{v_n}^T \Delta Q v_m}{\mu_m - \mu_n} \quad (59) \\ &\approx \frac{{v_n}^T (\Delta H^T H + H^T \Delta H) v_m}{\mu_m - \mu_n} \\ &\approx \frac{{v_n}^T \Delta H^T H v_m + {v_n}^T H^T \Delta H v_m}{\mu_m - \mu_n} \\ &\approx \frac{{v_n}^T \Delta H^T \sqrt{\mu_m} u_m + \sqrt{\mu_n} {u_n}^T \Delta H v_m}{\mu_m - \mu_n} \\ &\approx \frac{\sqrt{\mu_n} {u_n}^T \Delta H v_m + \sqrt{\mu_m} {u_m}^T \Delta H v_n}{\mu_m - \mu_n} \end{aligned} $$

By replacing H with H^T and interchanging v_k with u_k in Eq. 59, we get the following equation for B:

$$ {v_n}^T B v_m \approx \frac{\sqrt{\mu_m} {u_n}^T \Delta H v_m + \sqrt{\mu_n} {u_m}^T \Delta H v_n}{\mu_m - \mu_n} \quad (60) $$

We can determine A and B from Eqs. 59 and 60 element-by-element. The matrices are also anti-symmetric, meaning $A = -A^T$.

Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '03)
0-7695-2006-5/03 $17.00 © 2003 IEEE