MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A. I. Memo 855 **August, 1985**

Sensing Strategies for Disambiguating Among Multiple Objects in Known Poses

W. Eric L. Grimson

Abstract. The need for intelligent interaction of a robot with its environment frequently requires sensing of the environment. Further, the need for rapid execution requires that the interaction between sensing and action take place using as little sensory data as possible, while still being reliable. Previous work has developed a technique for rapidly determining the feasible poses of an object from sparse, noisy, occluded sensory data. In this paper, we examine techniques for acquiring position and surface orientation data about points on the surfaces of objects, with the intent of selecting sensory points that will force a unique interpretation of the pose of the object with as few data points as possible. Under some simple assumptions about the sensing geometry, we derive a technique for predicting optimal sensing positions. The technique has been implemented and tested. To fully specify the algorithm, we need estimates of the error in estimating the position and orientation of the object, and we derive analytic expressions for such error for the case of one particular approach to object recognition.

Acknowledgments. This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the Laboratory's Artificial Intelligence research is provided in part by a grant from the System Development Foundation, and in part by the Advanced Research Projects Agency under Office of Naval Research contracts N00014-80-C-0505 and N00014-82-K-0334.

Problem Definition

A robot often must recognize and locate objects in its workspace, or more informally, must use sensory information to determine *what* objects are *where*, in order to manipulate them. Since speed of operation is also an important consideration in robotics applications, the interaction of sensing and action should take place using a minimal amount of sensory data. This requires methods for optimally (or near optimally) selecting positions at which to obtain sensory data. Clearly, the notion of optimal selection of new data points will in part be tied to the specific recognition engine used to interpret those data points. In previous papers [Gaston and Lozano-Pérez 84; Grimson and Lozano-Pérez 84, 85a, 85b] we have presented a constraint based recognition and localization technique that uses as input, sparse, noisy, occluded measurements of the position and orientation of small patches of an object's surface obtained from any of several sensing modalities. Applying this recognition system to such sensory input data results in a small set of object poses, that is, a set of transformations taking a known object model from an intrinsic coordinate system into a coordinate system defined relative to the sensor. In this paper, we consider the problem of disambiguating from among this fixed set of object poses. Note that the set of poses could include poses corresponding to different objects. To disambiguate among a set of interpretations, we need to acquire sensory data that will clearly distinguish one pose of an object from another, using as few additional sensory points as possible. Thus, our problem is to optimally select places at which to obtain the needed sensory data.

While we use the recognition system developed in [Grimson and Lozano-Pérez 84, 85a, 85b] as the basis for investigating sensing strategies for disambiguation, we expect that some of the results of this investigation should have application in more general situations of recognition and localization. To illustrate this, we begin with a set of examples of the use of sensing strategies.

Example I: Disambiguating Multiple Interpretations

Suppose we are given a sparse set of sensory data points, each recording the position and orientation of a small patch of some surface in the workspace of a robot. Our goal is to determine what objects, from a set of known objects, are consistent with this data, together with the pose (position and orientation) of the object that leads to such a consistent interpretation. In the case of sensory data known to all lie on one object, we take *consistent* to mean that a rigid transformation of the object will cause all of the data points to lie on the object, with the correct surface orientation (to within some known error bounds). In the case of sensory data that may come from more than one object, we take *consistent* to mean that a maximum subset of the data satisfies the above condition. In this case, of course, other interpretations of consistent are possible.

In [Grimson and Lozano-Pérez 84, 85a, 85b] we described an efficient constrained search technique for matching the sensory data to faces of an object model, in order to find the interpretations of the data. The sensory data consist of measurements of the position and surface orientation of small patches of object surfaces. The objects are modeled by sets of planar faces equations. The technique uses efficient constraints

between data elements and model elements to determine the set of interpretations of the data consistent with the model, that is the set of poses of the object that agree with the input data. Empirical testing, as well as theoretical analysis [Grimson 84], indicates that in general, there will be only one consistent interpretation of the data. It is possible, however, that more than one pose of the object will be consistent with the data, even for non-symmetric objects and even if the object is known (see Figure 1). To determine the correct pose, we will need additional sensing.

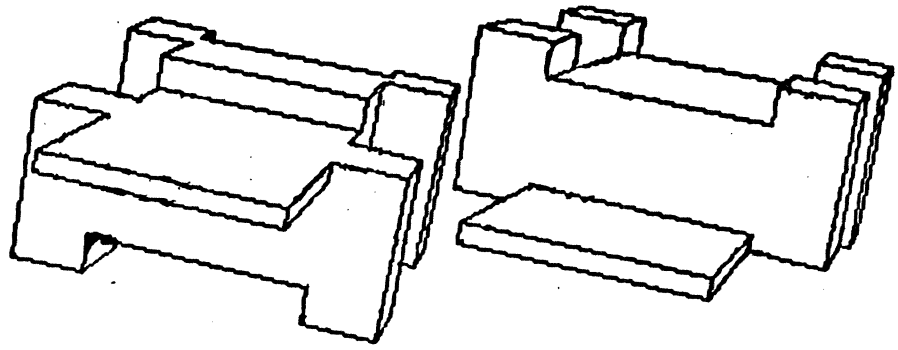

Figure 1. Example of multiple interpretations. Given only a sparse set of isolated data points, multiple interpretations such as those indicated may be possible.

The simplest method for obtaining the supplementary sensory data is to sample the object at random. If the sensing process is fast enough, and if, on average, only a few additional points are required in order to remove the ambiguity, then such a random sensing strategy could suffice. It is easy, however, to find situations in which a random sensing strategy would be ineffective in disambiguating between possible interpretations and in general one expects a random sensing strategy to have a very slow convergence. Moreover, some sensing modalities, for example, tactile sensing, are inherently sparse, and require considerable expense to obtain additional sensing points. In this case, it is particularly desirable to perform recognition with minimal sensory interaction.

No. of Points	9	10	11	12	13	14	15	16	17	18	19	20	>20
No. of Trials	59	15	6	5	4	1	1	2	1	1	2		3

Table 1 – Histograms of points needed for disambiguation. Each column indicates the number of sensory points needed to force a unique interpretation, and the number indicated in that column is the number of trials, out of 100, for which that number of data points was required.

For example, Table 1 lists histograms of the number of additional, randomly chosen, sensing points needed to uniquely disambiguate several consistent interpretations. We generated an initial set of 9 points of data, all lying on a single object (shown in Figure 1), and determined the set of consistent interpretations of that data, using the system

described in [Grimson and Lozano-Pérez 85a, 85b]. We then generated additional sense points until only one interpretation remained consistent with the data. This process was repeated for 100 trials, and the number of sense points needed to disambiguate the interpretations was histogrammed. The results are recorded in Table 1, where each entry is the number of trials terminating with the indicated number of sense points. The sensory data was generated by randomly choosing approach directions towards the object, and sensing for contact along them, much as might occur in tactile sensing. It can be seen that choosing sensing directions at random may have a slow convergence towards a unique interpretation, especially since in this case we are only dealing with the simple case of data from a single known object.

In general, one would expect a tradeoff between random sensing strategies and feature driven sensing strategies. Given two possible interpretations of the data, consider constructing the volume difference, consisting of all points contained in one but not both of the interpretations. If the size of this volume relative to the volume of the object is large, then in general, one would expect randomly generated additional sensing points to quickly disambiguate the situation. On the other hand, if the relative volume is small, one would expect that a large number of additional sense points would be needed before one of them struck this volume difference. In this case, a more directed sensing strategy is likely to be more effective.

Example II: Localization with Minimal Sensing

In the previous example, we discussed the problem of generating additional sensory data, given some initial set of data and the interpretations consistent with it. A related problem is to consider the optimal acquisition of all of the sensory data, rather than just that needed to disambiguate interpretations. For example, consider a situation in which a known object, with a fixed set of known stable positions is being sensed. This might be the case, for example, when considering objects in pallets, or feeders. We would like to determine the pose of the object with as few sensory points as possible. Here, the initial set of interpretations is the set of stable configurations of the object. Given this set of stable configurations, we want to determine the optimal sensing directions for distinguishing that set of configurations.

Example III: Simple Inspection

The problem of determining sensing positions can also arise in simple inspection tasks. Suppose we are given an object pose, and a set of distinctive points defined on the object model. In this case, we may be able to use the techniques developed below to choose the sensing rays needed to test that the designated distinctive model points are in fact present in the sensed object.

Assumptions

Thus, the problem to be addressed in this paper is finding effective and rigorous sensing strategies for deciding between a set of possible poses of an object, or multiple objects. We will assume that the following are given:

- Set of Interpretations — Some initial set of possible interpretations is assumed given. This could be either from the application of some recognition process to a set of initial sensed points, or from assumptions about the object to be sensed, in particular that it is lying in one of a known number of stable positions. In each case, the interpretation includes a computed transformation giving the pose of the model in sensor coordinates.
- Set of Sensing Directions — It is assumed that the initial sensory data were generated by sampling along a set of known directions. For example, in the case of visual sensing these could be given by the orientation of the cameras relative to the workspace. In general, determining optimal sensing rays is a four degree of freedom problem. In this paper, we assume that the two rotational degrees of freedom are restricted to a small set of possibilities by the sensing geometry, such as the given camera orientations. We then optimize over the remaining two degrees of freedom.
- Polyhedral Object Models — We assume that the objects to be sensed have been modeled as polyhedra, although the objects themselves need not be polyhedral. Any deviations between curved objects and their polyhedral models will simply contribute to a small amount of error in the sensory data, to which a recognition system should be insensitive.

The goal is to disambiguate between the set of interpretations by determining positions at which to obtain subsequent sensory information. These positions should be such that by sensing along one of the possible directions, the recorded information will disambiguate between the set of possible interpretations (or some subset of the interpretations) in the presence of possible error in the computed transformations associated with each of the interpretations.

In the examples given above, we assumed that we had available techniques for acquiring the sensory data, and techniques for solving the recognition and localization problem. There are, of course, many techniques for obtaining information about the three-dimensional positions of points on an object, as well as the local surface normals at those points. Typical examples of such measurement processes tactile sensing [e.g. Harmon 82, Hillis 82, Overton and Williams 81, Purbrick 81, Raibert and Tanner 82, Schneiter 82], binocular stereo [e.g. Baker and Binford 81, Barnard and Thompson 80, Grimson 81, 85, Marr and Poggio 79, Mayhew and Frisby 81, Ohta and Kanade 85], photometric stereo [e.g. Ikeuchi and Horn 79, Woodham 78, 80, 81], laser range-finding [e.g. Lewis and Johnston 77, Nitzan, Brain, and Duda 77], and structured-light systems [e.g. Popplestone, et al. 75, Shirai and Suwa 71]. These methods can provide information about the three-dimensional positions of points on the object, as well as the local surface normals at those points, usually with some error in the measurements.

A number of different techniques have been developed for model-based recognition and localization. If one views recognition as a search for a consistent match between

data elements and model elements, then much of the variation between existing recognition schemes can be accounted for by the choice of what descriptive tokens to match. Examples of techniques relying on sparse distinctive features include the use of a few extended features [Perkins 78, Ballard 81], the use of one feature as a focus, with the search restricted to a few nearby features [Tsuji and Nakamura 75, Holland 76, Sugihara 79, Bolles and Cain 82, Bolles, Horaud and Hannah 83], matching of high level descriptions [Nevatia 74, Nevatia and Binford 77, Marr and Nishihara 78, Brooks 81, Brady 82] and the use of geometric relationships between simple descriptors [Horn 83, Horn and Ikeuchi 83, Ikeuchi 83, Faugeras and Hebert 83, Gaston and Lozano-Pérez 84, Grimson and Lozano-Pérez 84, Stockman and Esteva 84, Brou 84]. The basis for the present work is the approach presented in [Gaston and Lozano-Pérez 84, Grimson and Lozano-Pérez 84, 85a, 85b].

For the purposes of this paper, we will assume that such techniques are available. Our concentration is on the problem of choosing optimal sensing strategies for interacting with such techniques.

An Algorithm For Computing Sensing Directions

To demonstrate the approach of computing sensing directions, we first look at an example in two dimensions (see Figure 2), where the object has three degrees of positional freedom (one rotational and two translational).

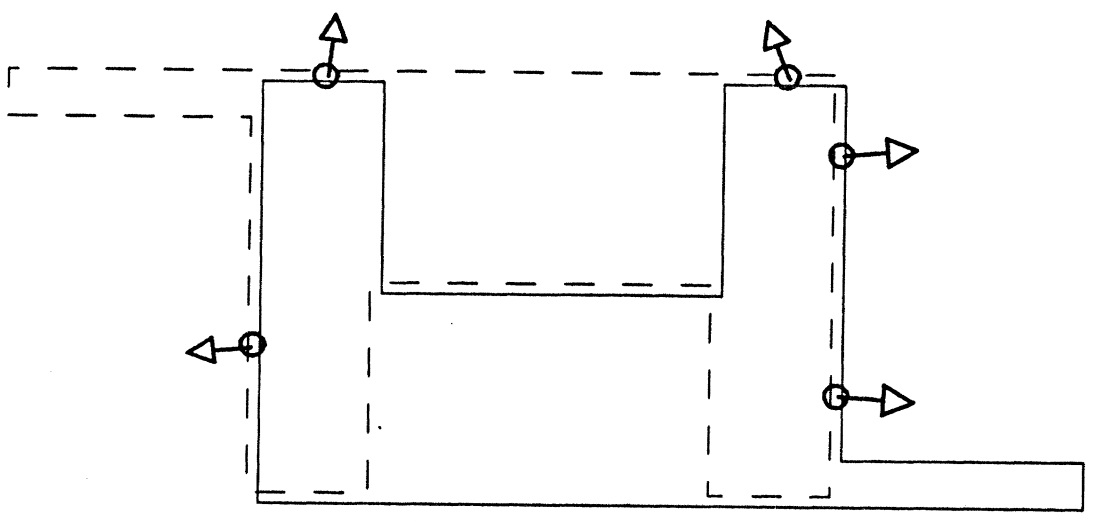

Figure 2. Two dimensional example of multiple poses. Both poses are consistent with the sensory data, indicated by the small surface normals and the points of contact.

After our recognition and localization process has been applied to a sparse set of data points, we are left with some set of poses of the object consistent with that data. We are given a set of sensing directions, that is, a set of unit vectors $\hat{\mathbf{s}}_i$ indexed over $i \in I$, such that sensing can occur along directions parallel to any of these unit vectors, for some set

of initial positions. For example, in Figure 3, if $\mathbf{o}$ is an offset vector, where $\mathbf{o} \cdot \hat{\mathbf{s}}_i = 0$, then we can sense along the ray $\mathbf{o} + \alpha\hat{\mathbf{s}}_i$ as α varies. Equivalently, we can think of this as having some finite portion of a plane perpendicular to $\hat{\mathbf{s}}_i$, such that for any point on the plane, we can sense along a ray through that point in the direction of $\hat{\mathbf{s}}_i$.

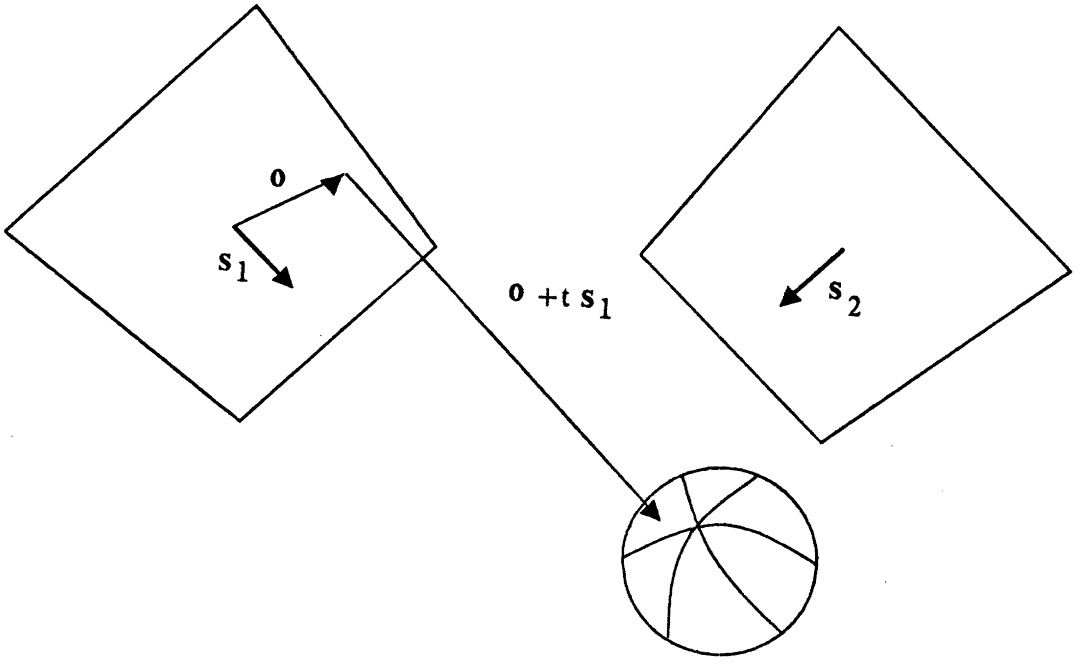

Figure 3. Examples of the sensing geometry. Each vector $\hat{\mathbf{s}}$ defines a sensing direction. The actual sensing ray is defined by specifying an offset vector $\mathbf{o}$ relative to the origin of the sensing plane through which the ray must pass, parallel to $\hat{\mathbf{s}}$.

We are also given some bounds on the sensitivity of the sensing device in measuring surface normals and surface positions. In particular, we define ϵ_n and ϵ_d in the following manner, illustrated in Figure 4. If $\hat{\mathbf{n}}_{true}$ is the surface normal at some point on an object, measured in sensor coordinates, and $\hat{\mathbf{n}}_{sense}$ is the normal measured by the sensing device, then

$$\hat{\mathbf{n}}_{sense} \cdot \hat{\mathbf{n}}_{true} > \epsilon_n.$$

If $\mathbf{p}_{true}$ is the actual position of a point on an object, measured in sensor coordinates, and $\mathbf{p}_{sense}$ is the position measured by the sensing device, then

$$|\mathbf{p}_{sense} - \mathbf{p}_{true}| < \epsilon_d.$$

Thus, ϵ_n and ϵ_d describe the range of uncertainty in the measurements of normals and distances, respectively.

The basic idea is that over the set of all given sensing directions $\{\hat{\mathbf{s}}_i | i \in I\}$, we want to find a particular direction $\hat{\mathbf{s}}_{i_0}$, and an offset position $\mathbf{o}$, such that sensing along the ray $\mathbf{o} + \alpha\hat{\mathbf{s}}_{i_0}$ will distinguish the poses. By distinguish, we mean that for all pairs of possible poses, either the difference in the expected normals of the faces that intersect the ray, or the difference in the expected positions of the points of intersection of the ray with the corresponding faces of the poses, is greater than the sensitivity of the sensing device.

We note that a sensing ray which does not intersect exactly one of the possible poses is acceptable. Indeed, in the case of two possible poses, sensing rays that would contact

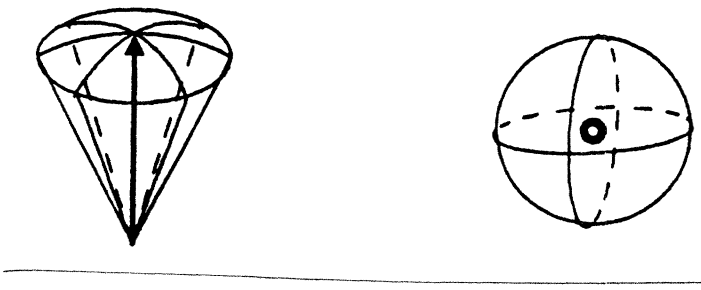

Figure 4. Error bounds. The true surface normal is known to lie within a specified cone of the measured normal, while the true position is known to lie within a specified ball about the measured position.

only one of the poses are likely to be among the best candidates for disambiguating the two poses. Secondly, we note that if there are many possible poses, it may not be possible to find one sensing ray that will distinguish between all of them. Instead, we may have to use a series of measurements to determine the correct pose. The number of such measurements will be bounded above by the number of poses, however.

The main problem to be faced in finding good sensing rays is the existence of error in the computed transformations associated with each pose. Thus, for the sensing strategy to be effective, the ray must both distinguish the poses, and be insensitive to errors in the position and orientation of the poses.

The proposed method is quite simple and is illustrated in Figure 5, in which two poses of the object are shown, one in solid lines, the other in hashed lines. The steps of the method are as follows.

1. Pick a particular sensing direction $\hat{\mathbf{s}}$ (we will assume the convention that $\hat{\mathbf{s}}$ points from the sensor towards the object). In the two-dimensional case, we can define a line perpendicular to the sensing direction, which we will call the *sensing line*, with origin at the point on the line closest to the origin of the sensor space. In three dimensions, this would be a *sensing plane*. This is shown in Figure 5a.
2. We fix the position of this line at some arbitrary reference point, for example by specifying the minimum distance of the line from the origin of the space to be d. This is shown in Figure 5b.
3. Now consider one of the poses, for example, the one shown in hashed lines in the figure. For each face f_i in the model, with corresponding model unit normal $\hat{\mathbf{n}}_{m,i}$, we let ${}^s\hat{\mathbf{n}}_{m,i}$ denote the unit normal rotated into sensor coordinates, i.e. corresponding to the orientation of the face relative to the pose of the object. If the face points towards the sensor ($\hat{\mathbf{s}} \cdot {}^s\hat{\mathbf{n}}_{m,i} < 0$), we project the boundaries of the face onto the sensing line, as shown for example in Figure 5c. In other words, each end point $\mathbf{e}$ of the edge is projected to a point on the sensing line, $\mathbf{e} + (d - \mathbf{e} \cdot \hat{\mathbf{s}})\hat{\mathbf{s}}$. In three dimensions, this would entail the projection of the edges of a face onto the sensing plane.
4. We can label the resulting segment of the $\hat{\mathbf{s}}$-line with the surface normal ${}^s\mathbf{n}_{m,i}$ and with the range of distances from the object face to the $\hat{\mathbf{s}}$-line. That is, if $\mathbf{v}$ is a point on the edge, in sensor coordinates, then $\mathbf{v} \cdot \hat{\mathbf{s}} - d$ is the distance from the point to the $\hat{\mathbf{s}}$-line. We let $\alpha_{\min}$ and $\alpha_{\max}$ denote the extreme values taken on by $\mathbf{v} \cdot \hat{\mathbf{s}} - d$ as $\mathbf{v}$ ranges over the edge, and the segment is labeled by

$$\left\{ {}^s\hat{\mathbf{n}}_{m,i}, \left\{ \alpha_{\min,i}, \alpha_{\max,i} \right\} \right\}.$$

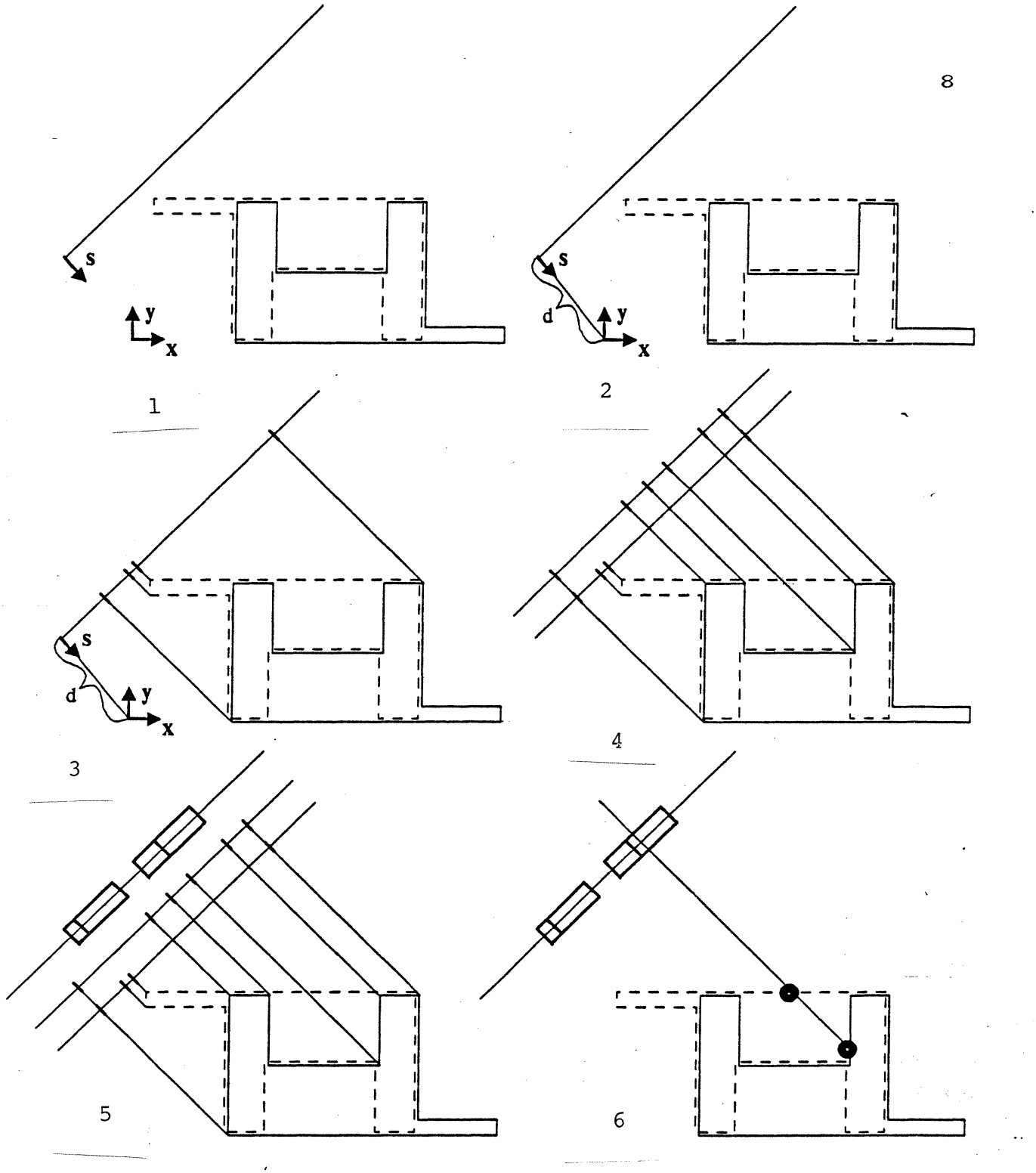

Figure 5. Projection of Poses onto the Sensing Line. In part (a) the sensing line is indicated, orthogonal to the sensing direction. In part (b) this line is fixed at a distance d from the origin. In part (c) the visible faces of one of the poses are projected onto the sensing line defined by the sensing ray. This projection defines a partition of the sensing line. Here $\hat{\mathbf{s}}$ is the sensing ray and d is the distance to the origin. In part (d), the visible faces of the second pose are also projected onto the sensing line. In part (e), the respective partitions are tested for distinguishability, based on differences in expected surface orientation and differences in expected position, and the distinguishable regions of the sensing line are marked. Using a sensing ray through the midpoint of either of the two marked regions would enable one to disambiguate the two poses, as shown in part (f), in which the expected sensory points are indicated.

5. When all the visible faces of a pose have been projected onto the $\hat{\mathbf{s}}$-line, we can perform hidden surface removal, to reduce the set of possibly overlapping segments of the $\hat{\mathbf{s}}$-line to a set of disjoint segments. Each segment will be labeled by the surface orientation of the corresponding face, in sensor coordinates, and the range of distances to points on the face.
6. We can perform this operation for each pose, obtaining a different disjoint partition of the $\hat{\mathbf{s}}$-line, labeled by the appropriate surface normals and distance ranges, as shown in Figure 5d (slightly offset for graphical clarity).
7. Next, we intersect the set of all such partitions. That is, we define a new partition of the $\hat{\mathbf{s}}$-line with two properties. First, each segment of this new partition lies within exactly one segment of each of the partitions of the $\hat{\mathbf{s}}$-line corresponding to a pose. Second, this new partition is the smallest (in terms of number of segments) such partition. The label associated with each segment of the new partition is the union of the labels of the corresponding segments of the individual partitions.
8. This partition can now be analyzed for distinguishability. More precisely, given a segment of the partition, the set of normals

$$\{\hat{\mathbf{n}}_j | j \in J\}$$

associated with that segment is *distinguishable* if

$$\max_{i,j \in I, i \neq j} (\hat{\mathbf{n}}_i \cdot \hat{\mathbf{n}}_j) < 2\epsilon_n.$$

In other words, given a measurement of the actual object in this region, we can uniquely determine to which pose it corresponds. Similarly, the set of distance measurements

$$\left\{ \left(\alpha_{\min,j}, \alpha_{\max,j} \right) | j \in J \right\}$$

is *distinguishable* if

$$\max_{i \neq j} \{ |\alpha_{\min,i} - \alpha_{\max,j}| \} > 2\epsilon_d.$$

We can collect all such distinguishable segments of the partition, thereby determining the set of possible sensing points along the particular choice of $\hat{\mathbf{s}}$. This is illustrated in Figure 5(e).

If there were no error in the transformations associated with the poses, we would be done, since any point in this set would disambiguate the poses, (see Figure 5(f) for an example). To account for possible error in the transformations associated with the poses, however, we need to be somewhat judicious in our choice of sensing point. The basic idea is to choose a point such that the face with which contact is made remains the same over small perturbations in the transformation. In two dimensions this is most easily done by choosing the midpoint of the longest segment. In three dimensions, the easiest way to choose such a point, from among the set of distinguishable polygons on the sensing plane, is by applying the notion of a Chebychev point, defined as follows. Suppose we are given a polygon on a plane, each of whose edges is defined by a pair $(\hat{\mathbf{n}}_j, d_j)$, where $\hat{\mathbf{n}}_j$ is a unit normal lying in the plane, and d_j is a constant such that points along the edge are defined by

$$\{\mathbf{v} | \mathbf{v} \cdot \hat{\mathbf{n}}_j - d_j = 0\}.$$

Then the distance from any point $\mathbf{v}$ to an edge is given by

$$\mathbf{v} \cdot \hat{\mathbf{n}}_j - d_j.$$

The Chebychev point of a polygon is the point which maximizes the minimum distance from the point to any edge of the polygon, that is, the point $\mathbf{v}$ that satisfies

$$\min_j (\mathbf{v} \cdot \hat{\mathbf{n}}_j - d_j) \geq \min_j (\mathbf{u} \cdot \hat{\mathbf{n}}_j - d_j) \qquad \forall \mathbf{u}$$

where the value taken by this expression at the Chebychev point is called the Chebychev value of the polygon. Clearly, the polygon with the maximum Chebychev value will be the least sensitive to perturbations in the computed transformations, and thus the Chebychev point with the maximum Chebychev value, as measured over the set of all distinguishable polygons, defines the best sensing position. Note that we can improve the reliability of the sensing strategy even further by choosing the maximum Chebychev point as measured over connected sequences of distinguishable segments.

9. We repeat this process over all sensing directions $\hat{\mathbf{s}}_i$, choosing the direction that best distinguishes the feasible poses.

While this analysis has been done in two dimensions, it clearly extends to the general three dimensional case. Here, the visible faces are projected into polygons on a sensing plane, and the intersection of the projections over all poses gives a partition of this plane, which can be tested for distinguishability.

An Implementation of the Technique

In testing the proposed algorithm, we have chosen a slightly modified implementation of the technique, that avoids some of the difficulties of performing hidden surface removal, and of intersecting polygonal partitions of a plane. One means of circumventing these difficulties is to use a regular grid tesselation of the plane.

In particular, suppose that we partition the $\hat{\mathbf{s}}$-plane with a rectangular grid whose elements have sides of length h. Rather than trying to compute polygonal regions on the $\hat{\mathbf{s}}$-plane that are distinguishable, we shall examine each grid segment within the bounds of the projected object, seeking those segments that are themselves distinguishable, and then we will piece these grid elements back together.

The steps of the new algorithm, many of which are identical to those of the previous solution, are sketched below.

- Initially, mark all grid segments as active.
- Given a pose, and a sense direction $\hat{\mathbf{s}}$, test each face for visibility. If the normal of the face, in sensor coordinates, is given by ${}^s\hat{\mathbf{n}}_{m,i}$, then a face is visible if $\hat{\mathbf{s}} \cdot {}^s\hat{\mathbf{n}}_{m,i} < 0$.
- For each visible face, project its vertices onto the $\hat{\mathbf{s}}$-plane, resulting in a set of new vertices that define a polygon on the plane.
- Given this polygon on the sensing plane, compute the smallest bounding rectangle composed of an integral number of grid elements which encloses the enscribed polygon. This rectangle has no intrinsic merit, but is simply a convenient means of restricting the search process.
- For each grid element lying in this enclosing rectangle, apply the following test. If the grid segment lies entirely outside of the polygon, nothing is done. If some edge of the polygon passes through the segment, this segment is marked as inactive. If the

grid segment is still active and lies entirely within the polygon, a label is attached to the grid segment. This label is composed of two elements. The first is the normal of the face whose projection resulted in the current polygon on the sensing plane, measured in sensor coordinates. The second is the range of possible positions that could be achieved by intersecting a sensing ray passing through a point in this grid segment with the face of the underlying interpretation. If the vector $\mathbf{o}$, lying in the $\hat{\mathbf{s}}$-plane, defines the midpoint of the grid segment, this range is given by

$$\alpha_0 \pm \frac{\sqrt{2}h}{2} \tan\theta$$

where α_0 is the value of α for which the ray $\mathbf{o} + \alpha\hat{\mathbf{s}}$ intersects the face of the pose, h is the size of the grid segment, θ is given by $\cos\theta = {}^s\hat{\mathbf{n}} \cdot \hat{\mathbf{s}}$, and ${}^s\hat{\mathbf{n}}$ is the normal of the face in sensor coordinates.

- Repeat this process for all visible faces. This results in a set of *active* grid segments, each of which is labeled by possibly several labels of the type described above. This set of labeled active grid segments represents the equivalent of the partition of the sensing plane described in the ideal solution. Note that we have avoided the hidden surface problem by incorporating multiple labels for a grid segment, from a single pose. This may reduce the number of distinguishable segments, by applying additional constraints on the criteria of distinguishability, but it also greatly reduces the computational expense of the process.

- Once a partition of the grid is obtained for each pose, test the grid segments for distinguishability. First, only grid segments that are active in all poses are considered. Such a segment is considered *distinguishable* if for all pairs of sets of labels, either all the face normals of one label are distinguishable from all the face normals of the other (in the sense defined in the previous section), or all the distance ranges of one label are distinguishable from all the distance ranges of the other (also in the sense defined in the previous section).

- Finally, collect the set of distinguishable grid segments into convex connected components.

- Compute the best sensing position as the center of the largest square (with sides an integral number of grid segments) that can be placed entirely within the set of distinguishable grid segments. Note that if the square has sides of size s then the Chebychev value for the segment is at least $s/2$. This process can be repeated over all sensing directions, and the midpoint of the largest such connected convex collection of distinguishable grid segments can be used to define the best sensing position. To save on computation, it is also possible to define a minimum size for an acceptable convex connected component, and to only apply this process until the first such acceptable component is obtained.

In Figure 6, we illustrate the above technique on the multiple poses of Figure 1. Note that each of the small circles denotes a point on the grid of the sensing plane that is distinguishable. We can then determine the best sensing position by finding the largest square area filled by such distinguishable points. The figure illustrates the computation for each of three different sensing rays.

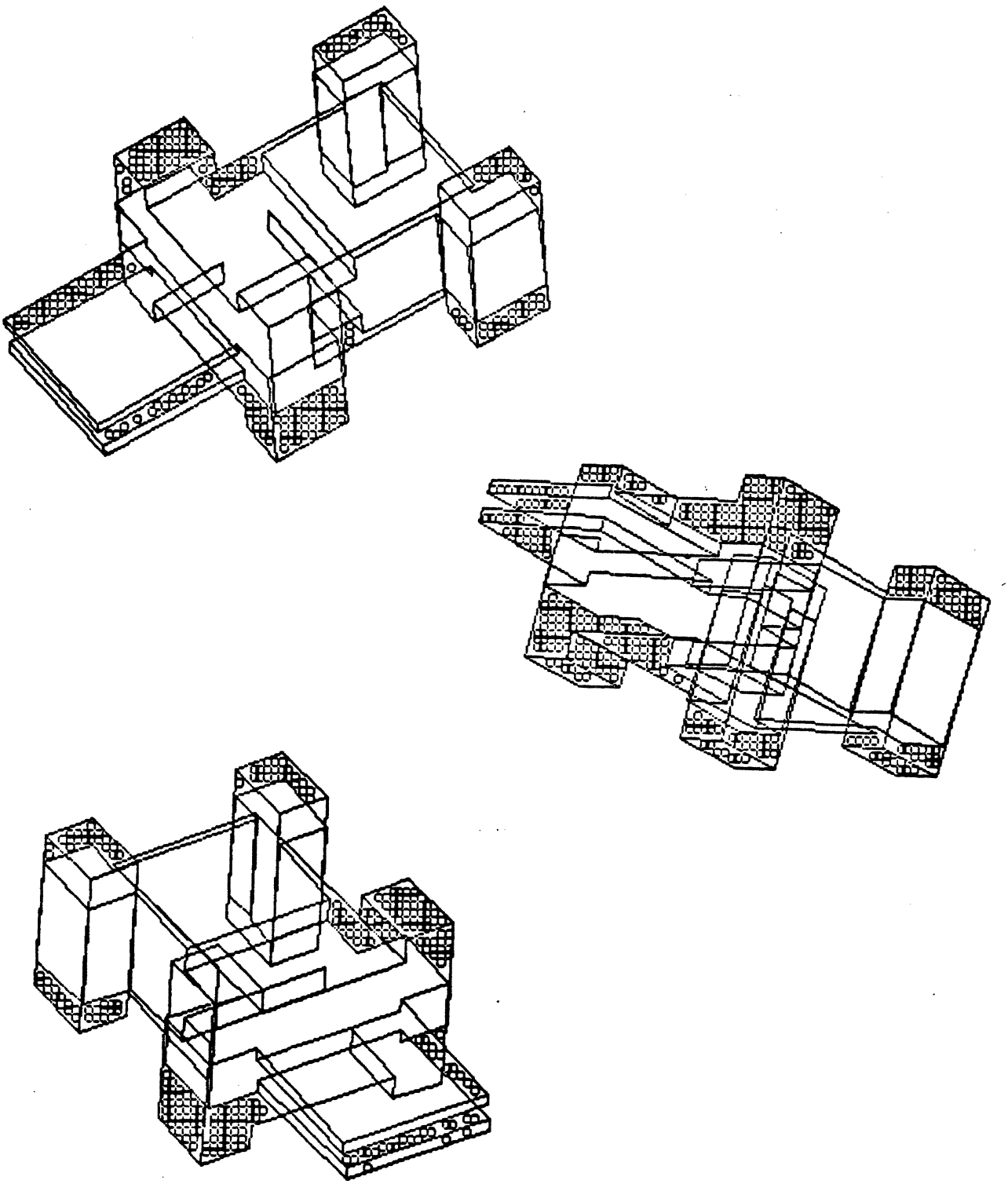

Figure 6. Example of distinguishable points. For the multiple poses of Figure 1, we show three different sensing plane projections. The small circles mark positions, on the defined grid, that are distinguishable in these poses.

Bounds on Transform Errors

In order to use such an algorithm, we need to determine values for two parameters. First, errors in the computed transformation associated with a pose will affect the threshold needed to determine distinguishability. For example, if there is no error in the computed transformation, then two surface normals are distinguishable if the angle between them exceeds the range of error in measuring such normals. When error is present in the transformation, its effect on the expected surface normals must be added to this threshold, thereby reducing the set of distinguishable normals. Second, we need a bound on the minimum Chebychev value (or its approximation) such that errors in the computed transformation will not affect the expected values of the sensor along the sensing ray. In order to deal with these parameters, in this section we will derive theoretical bounds on the possible errors in the computed transformations. In doing so, we will also derive criteria that can be imposed on the computation of the transformation from model coordinates to sensor coordinates in order to reduce the range of possible error. Depending on the sensor data available, it may not always be possible to satisfy these criteria, in which case higher possible errors will have to be tolerated.

Computing the Transform

There are many different methods for determining the transformation from model coordinates to sensor coordinates, and the errors associated with that computation will clearly be dependent on the specific method. To illustrate the disambiguation technique developed here, we choose one particular scheme, and derive specific error bounds on the model transformation for that scheme. This will then allow us to actually test our disambiguation algorithm. We being by reviewing the process used in [Grimson and Lozano-Pérez 84] for computing the transformation from model coordinates to sensor coordinates.

We are given a set of possible poses of the sensed data, each one consisting of a set of triples $(\mathbf{p}_i, \hat{\mathbf{n}}_i, f_i)$, where $\mathbf{p}_i$ is the vector representing the sensed position, $\hat{\mathbf{n}}_i$ is the vector representing the sensed normal, and f_i is the face assigned to this sensed data for that particular pose. We want to determine the actual transformation from model coordinates to sensor coordinates, corresponding to the pose [see also Grimson and Lozano-Pérez 84].

We assume that a vector in the model coordinate system is transformed into a vector in the sensor coordinate system by the following transformation:

$$\mathbf{v}_s = R\mathbf{v}_m + \mathbf{v}_0$$

where R is a rotation matrix, and $\mathbf{v}_0$ is some translation vector. We need to solve for R and $\mathbf{v}_0$.

Rotation Component

Suppose $\hat{\mathbf{n}}_{m,i}$ is the unit normal, in model coordinates, of face f_i, and $\hat{\mathbf{n}}_{s,i}$ is the corresponding unit normal in sensor coordinates. Given a two such pairs of model and sensor normals, an estimate of the direction of rotation $\hat{\mathbf{r}}_{ij}$ such that a rotation about that

direction would take $\hat{\mathbf{n}}_{m,i}$ into $\hat{\mathbf{n}}_{s,i}$ is given by the unit vector in the direction of

$$(\hat{\mathbf{n}}_{m,i} - \hat{\mathbf{n}}_{s,i}) \times (\hat{\mathbf{n}}_{m,j} - \hat{\mathbf{n}}_{s,j}).$$

If there were no error in the sensed normals, we would be done. With error included in the measurements, however, the computed rotation direction $\hat{\mathbf{r}}$ could be slightly wrong. One way to reduce the effect of this error is to compute all possible $\hat{\mathbf{r}}_{ij}$ as i and j vary over the faces of the pose, and then cluster these computed directions to determine a value for the direction of rotation $\hat{\mathbf{r}}$.

Once we have computed a direction of rotation $\hat{\mathbf{r}}$, we need to determine the angle θ of rotation about it. This is given by

$$\begin{aligned} \cos\theta &= 1 - \frac{1 - (\hat{\mathbf{n}}_{s,i} \cdot \hat{\mathbf{n}}_{m,i})}{1 - (\hat{\mathbf{r}} \cdot \hat{\mathbf{n}}_{s,i})(\hat{\mathbf{r}} \cdot \hat{\mathbf{n}}_{m,i})} \\ \sin\theta &= \frac{(\hat{\mathbf{r}} \times \hat{\mathbf{n}}_{s,i}) \cdot \hat{\mathbf{n}}_{m,i}}{1 - (\hat{\mathbf{r}} \cdot \hat{\mathbf{n}}_{s,i})(\hat{\mathbf{r}} \cdot \hat{\mathbf{n}}_{m,i})}. \end{aligned} \tag{1}$$

Hence, given $\hat{\mathbf{r}}$, we can solve for θ. Note that if $\sin\theta$ is zero, there is a singularity in determining θ, which could be either 0 or π. In this case, however, $\hat{\mathbf{r}}$ lies in the plane spanned by $\hat{\mathbf{n}}_{s,i}$ and $\hat{\mathbf{n}}_{m,i}$ and hence, only the $\theta = \pi$ solution is valid.

As before, in the presence of error, we may want to cluster the $\hat{\mathbf{r}}$ vectors, and then take the average of the computed values of θ over this cluster.

Finally, given values for both $\hat{\mathbf{r}}$ and θ, we can determine the rotation matrix R. Let r_x, r_y, r_z denote the components $\hat{\mathbf{r}}$. Then

$$R = \cos\theta \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} + (1 - \cos\theta) \begin{bmatrix} r_x^2 & r_x r_y & r_x r_z \\ r_y r_x & r_y^2 & r_y r_z \\ r_z r_x & r_z r_y & r_z^2 \end{bmatrix} + \sin\theta \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix}$$

Note that in computing the rotation component of the transformation, we have ignored the ambiguity inherent in the computation. That is, there are two solutions to the problem, $(\hat{\mathbf{r}}, \theta)$ and $(-\hat{\mathbf{r}}, -\theta)$. We assume that a simple convention concerning the sign of the rotation is used to choose one of the two solutions.

Translation Component

Next, we need to solve for the translation component of the transformation. Suppose we consider three triplets from the pose, $(\mathbf{p}_{s,i}, \hat{\mathbf{n}}_{s,i}, f_i)$, $(\mathbf{p}_{s,j}, \hat{\mathbf{n}}_{s,j}, f_j)$, and $(\mathbf{p}_{s,k}, \hat{\mathbf{n}}_{s,k}, f_k)$ such that the triple product $\hat{\mathbf{n}}_{m,i} \cdot (\hat{\mathbf{n}}_{m,j} \times \hat{\mathbf{n}}_{m,k})$ is non-zero, (i.e. the three face normals are independent). Then, it can be shown that the translation component of the transformation, $\mathbf{v}_0$, is given by

$$\begin{aligned} [\hat{\mathbf{n}}_{m,i} \cdot (\hat{\mathbf{n}}_{m,j} \times \mathbf{n}_{m,k})]\, \mathbf{v}_0 = &\left(\hat{\mathbf{n}}_{s,i} \cdot \mathbf{p}_{s,i} - d_i\right) (\hat{\mathbf{n}}_{s,j} \times \hat{\mathbf{n}}_{s,k}) \\ &+ \left(\hat{\mathbf{n}}_{s,j} \cdot \mathbf{p}_{s,j} - d_j\right) (\hat{\mathbf{n}}_{s,k} \times \hat{\mathbf{n}}_{s,i}) \\ &+ \left(\hat{\mathbf{n}}_{s,k} \cdot \mathbf{p}_{s,k} - d_k\right) (\hat{\mathbf{n}}_{s,i} \times \hat{\mathbf{n}}_{s,j}) \end{aligned}$$

As in the case of rotation, if there is no error in the measurements, then we are done. The simplest means of attempting to reduce the effects of error on the computation is to average $\mathbf{v}_0$ over all possible trios of triplets from the pose.

Errors in the computed transformation

We now consider possible errors in each of the parameters of the transformation, as a function of error in the sensor measurements. The results are summarized below, more explicit details may be found in the appendix.

Errors in $\hat{\mathbf{r}}$.

Let $\hat{\mathbf{n}}_{m,i}$ be the unit normal of face f_i, in model coordinates, let $\hat{\mathbf{n}}'_{m,i}$ be the associated unit normal transformed into sensor coordinates, and let $\hat{\mathbf{n}}_{s,i}$ be the actual measured unit normal, in sensor coordinates. Suppose that the sensitivity of the measuring device was ϵ_n, that is,

$$\hat{\mathbf{n}}'_{m,i} \cdot \hat{\mathbf{n}}_{s,i} \geq \epsilon_n.$$

Then an absolute bound on the possible error in the computed value for the direction of rotation, $\hat{\mathbf{r}}_c$, in relation to the true direction of rotation, $\hat{\mathbf{r}}_t$, is given by

$$\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_c \geq \frac{\delta_i \delta_j \sqrt{1-\eta^2}}{\sqrt{1-\left\{\delta_i\delta_j\eta - \sqrt{1-\delta_i^2}\sqrt{1-\delta_j^2}\right\}^2}},$$

where

$$\eta = \left(\frac{\hat{\mathbf{n}}_{m,i} - \hat{\mathbf{n}}'_{m,i}}{|\hat{\mathbf{n}}_{m,i} - \hat{\mathbf{n}}'_{m,i}|}\right) \cdot \left(\frac{\hat{\mathbf{n}}_{m,j} - \hat{\mathbf{n}}'_{m,j}}{|\hat{\mathbf{n}}_{m,j} - \hat{\mathbf{n}}'_{m,j}|}\right)$$

$$\delta_i \geq \sqrt{\frac{\epsilon_n - \hat{\mathbf{n}}_{m,i} \cdot \hat{\mathbf{n}}'_{m,i}}{1 - \hat{\mathbf{n}}_{m,i} \cdot \hat{\mathbf{n}}'_{m,i}}}.$$

Note that if γ_i is close to ϵ_n, then the error bound becomes increasingly large. This is to be expected, since in this case, $\hat{\mathbf{n}}_{m,i} \approx \hat{\mathbf{n}}'_{m,i}$ and thus small errors in the position of $\hat{\mathbf{n}}$ can lead to large errors in the position of $\hat{\mathbf{r}}$. Similarly, if η is near 1, large errors can also result. If we restrict our computation (where possible) to cases where γ_i and η are small, then we have an approximate bound on the error in computing the direction of rotation given by

$$\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_c \geq \epsilon_n.$$

This bound is supported by the results of the simulations reported in [Grimson and Lozano-Pérez 84].

Errors in θ.

We know that the angle of rotation θ is given by

$$\tan\theta = -\frac{\hat{\mathbf{n}}'_m \cdot (\hat{\mathbf{r}} \times \hat{\mathbf{n}}_m)}{(\hat{\mathbf{r}} \times \hat{\mathbf{n}}'_m) \cdot (\hat{\mathbf{r}} \times \hat{\mathbf{n}}_m)}$$

where $\hat{\mathbf{n}}_m$ is the unit normal of a face in model coordinates, $\hat{\mathbf{n}}'_m$ is the corresponding normal transformed into sensor coordinates, and $\hat{\mathbf{r}}$ is the direction of rotation.

If we let $\hat{\mathbf{r}}_t$ denote the true direction of rotation, $\hat{\mathbf{r}}_c$ denote the computed direction of rotation, and $\hat{\mathbf{n}}_s$ denote the measured surface normal corresponding to $\hat{\mathbf{n}}'_m$, then the

constraints on the error in computing the angle θ are that $\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_c \geq \cos\psi$ and $\hat{\mathbf{n}}'_m \cdot \hat{\mathbf{n}}_s \geq \epsilon_n = \cos\phi$. In the appendix we show that the correct value for θ is given by

$$\tan\theta_t = -\frac{1}{\sin\nu}\frac{\cos\alpha}{\cos\beta}$$

where

$$\cos\nu = \hat{\mathbf{n}}_m \cdot \hat{\mathbf{r}}_t$$

$$\cos\alpha = \left(\hat{\mathbf{n}}'_m \cdot \left(\frac{\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m}{|\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m|}\right)\right)$$

$$\cos\beta = \left(\left(\frac{\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}'_m}{|\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}'_m|}\right) \cdot \left(\frac{\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m}{|\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m|}\right)\right).$$

Furthermore, we show in the appendix that the worst case for the computed value of θ is given by

$$\tan\theta_c = -\frac{1}{\sin\omega}\frac{\cos(\alpha - [\phi + \xi])}{\cos(\beta + [\psi + \gamma])}$$

where

$$\cos\omega = \cos(\phi + \psi) - (1 - \cos\nu)\cos\phi\cos\psi$$

$$\cos\xi = \cos\psi\sqrt{\frac{1 - \cos^2\nu}{1 - \cos^2\psi\cos^2\nu}}$$

$$\cos\gamma = \cos\phi\cos\psi\sqrt{\frac{1 - \cos^2\nu}{1 - \cos^2\omega}}.$$

We could use these expressions to derive bounds on the possible variation in θ as a function of ϕ and ψ, but this is a rather messy task. Instead, we show in the appendix that if ϕ and ψ are small, then an estimate for $\Delta\theta$ such that

$$\tan(\theta_t + \Delta\theta) \approx \tan\theta_c$$

is given by

$$|\Delta\theta| \approx |\phi + \psi|.$$

This bound is supported by the results of simulations reported in [Grimson and Lozano-Pérez 84].

Errors in $R\mathbf{v}$

We have computed expressions for the possible error in $\hat{\mathbf{r}}$ and θ. In particular, we will denote the error in θ by $\Delta\theta$ and the vector error in $\hat{\mathbf{r}}$ by $\delta\hat{\mathbf{r}}$ such that $\hat{\mathbf{r}} \cdot \delta\hat{\mathbf{r}} = 0$. We now consider the problem of estimating bounds on the possible error in applying the computed rotation matrix to an arbitrary vector $\mathbf{v}$. We know that the rotational component of the transformation of $\mathbf{v}$ is given by

$$R(\hat{\mathbf{r}}, \theta)\mathbf{v} = \cos\theta\mathbf{v} + (1 - \cos\theta)(\hat{\mathbf{r}} \cdot \mathbf{v})\hat{\mathbf{r}} + \sin\theta(\hat{\mathbf{r}} \times \mathbf{v})$$

where $\hat{\mathbf{r}}$ and θ are the parameters determining the rotation.

We show in the appendix that if we ignore higher order terms, a Taylor series expansion yields the following bound on errors in the computed value of a rotation:

$$|R(\hat{\mathbf{r}} + \delta\hat{\mathbf{r}}, \theta + \Delta\theta)\mathbf{v} - R(\hat{\mathbf{r}}, \theta)\mathbf{v}| \leq (2|\delta\hat{\mathbf{r}}| + |\Delta\theta|)|\mathbf{v}|.$$

Now, if the errors ϕ and ψ are small, then we know that

$$|\Delta\theta| \leq |\phi + \psi|.$$

Furthermore,

$$|\delta\hat{\mathbf{r}}| = |\sin\psi| \approx |\psi|$$

and this implies a bound on variation in $\mathbf{v}$ of

$$|3\psi + \phi|\,|\mathbf{v}|\,.$$

Moreover, if we are careful to restrict our computation appropriately, then $\psi \approx \phi$, and thus

$$|R(\hat{\mathbf{r}} + \delta\hat{\mathbf{r}}, \theta + \Delta\theta)\,\mathbf{v} - R(\hat{\mathbf{r}}, \theta)| \leq |4\phi|\,|\mathbf{v}|\,.$$

Effective bounds on rotation errors

Unfortunately, this is still a fairly weak bound. For example, an error cone of radius $\frac{\pi}{12}$ about the measured surface normals would give rise to potential errors in the computed rotation on the order of the magnitude of the rotated vector. This is obviated to a large extent by the fact that we do not rely on a single measurement in computing the transformation parameters. Rather, we use several sets of measurements, and use the mean value as the result when computing $\hat{\mathbf{r}}$ and θ.

To see how this helps reduce the effective bound, consider the following argument. Suppose that the error in computing θ is uniformly distributed over the range $[-2\phi, 2\phi]$. If we take n measurements and average, then the distribution of error about the correct value θ_c should approach a normal distribution, by the Central Limit Theorem. If we assume a uniform distribution for the error in each measurement, then the variance in the error can be shown to equal

$$\frac{4\phi^2}{3}.$$

If there is no systematic error in the measurements, i.e. each measurement error can be considered independent of the others, then the distribution of average error is essentially a zero-mean normal distribution with variance

$$\frac{4\phi^2}{3n}$$

and hence with standard deviation

$$\sqrt{\frac{4\phi^2}{3n}}.$$

Similarly, if the magnitude of the error vector, $\delta\hat{\mathbf{r}}$, associated with the computation of the direction of rotation, $\hat{\mathbf{r}}$, is uniformly distributed over its possible range, and the measurements are independent, then the distribution of error in $2\,|\delta\hat{\mathbf{r}}|$ is given by an identical normal distribution, since the maximum error in $|\delta\hat{\mathbf{r}}|$ is essentially ϕ. By linearly combining the two distributions, the error in the computation of $R\mathbf{v}$ is given by a zero-mean normal distribution with variance

$$\frac{8\phi^2}{3n}.$$

While an absolute bound on the error in computing $R\mathbf{v}$ is given by $4\,|\phi|\,|\mathbf{v}|$, tighter, but less certain, bounds are possible. For example, if we impose a 0.95 probability that the error does not exceed the bound, then an expression for this bound is given by the normal distribution error function, and in this particular case, by

$$\frac{3.92\sqrt{2}}{\sqrt{3n}}\phi.$$

As the number of samples n increases, this bound becomes increasingly tighter.

Note that while we have assumed a uniform distribution of the errors in the individual measurements, this is not a critical assumption. Since we are only seeking estimates for the bounds in computational error, other distributions will give similar results.

In summary, given some lower bound on the number of samples to be used in computing the transformation from model coordinates to sensor coordinates, and given that the assumption of small errors in the measurements of surface orientation holds, then the error in the computed rotation of a vector $\mathbf{v}$ is given by a zero-mean normal distribution, scaled by the magnitude $|\mathbf{v}|$, with standard deviation

$$\sqrt{\frac{8}{3n}}\phi$$

where n is the number of measurement samples and ϕ is the angle of maximum error in the measurement of surface orientation at each sample point.

Errors in $\mathbf{v}_0$.

We know that the translation component of the transformation is given by

$$\begin{aligned}[\hat{\mathbf{n}}_{m,i}\hat{\mathbf{n}}_{m,j}\hat{\mathbf{n}}_{m,k}]\,\mathbf{v}_0 = &\left(\hat{\mathbf{n}}'_{m,i}\cdot\mathbf{p}_{s,i} - d_i\right)\left(\hat{\mathbf{n}}'_{m,j}\times\hat{\mathbf{n}}'_{m,k}\right)\\ &+\left(\hat{\mathbf{n}}'_{m,j}\cdot\mathbf{p}_{s,j} - d_j\right)\left(\hat{\mathbf{n}}'_{m,k}\times\hat{\mathbf{n}}'_{m,i}\right)\\ &+\left(\hat{\mathbf{n}}'_{m,k}\cdot\mathbf{p}_{s,k} - d_k\right)\left(\hat{\mathbf{n}}'_{m,i}\times\hat{\mathbf{n}}'_{m,j}\right)\end{aligned}$$

where $\hat{\mathbf{n}}_{m,i}$ is a face normal in model coordinates, $\hat{\mathbf{n}}'_{m,i}$ is the corresponding face normal transformed into sensor coordinates, $\mathbf{p}_{s,i}$ is the position vector of the contact point in sensor coordinates, and d_i is the constant offset for face i.

If the error in measured surface normals is given by $\epsilon_n = \cos\phi$ such that $\hat{\mathbf{n}}_s\cdot\hat{\mathbf{n}}'_m \geq \epsilon_n$ and the error in measured contact positions is bounded in magnitude by ϵ_d, then the error in each component

$$\left(\hat{\mathbf{n}}'_{m,k}\cdot\mathbf{p}_{s,k} - d_k\right)\left(\hat{\mathbf{n}}'_{m,i}\times\hat{\mathbf{n}}'_{m,j}\right)$$

is bounded in magnitude by

$$\sqrt{\left[s\sin\varsigma - (s+\Delta)\sin(\varsigma - 2\phi)\right]^2 + (s+\Delta)^2\sin(2\varsigma)\sin(4\phi)}$$

where

$$\begin{aligned}s &= \hat{\mathbf{n}}'_{m,k}\cdot\mathbf{p}_{s,k} - d_k\\ \Delta &\leq \epsilon_d + |\mathbf{p}_{s,k}|\sqrt{2}\sqrt{1-\epsilon_n}\\ \cos\varsigma &= \hat{\mathbf{n}}'_{m,i}\cdot\hat{\mathbf{n}}'_{m,j}.\end{aligned}$$

If we restrict our computation to cases in which the faces are nearly orthogonal, then this bound on the components of the translation vector reduces to

$$|s - (s+\Delta)\cos(2\phi)|.$$

Choosing the Parameters

We now consider deriving a formal definition of *distinguishability* as applied to surface normals and to distances. Consider first the case of distinguishing poses on the basis of measured surface normals. Suppose that α denotes the angle between surface normals associated with two different possible poses. What is the minimum size of α needed to distinguish these poses?

Clearly, the expected normals must differ by an amount that is bigger than the sensitivity of the measurements themselves. Thus, α must at least exceed $2\cos^{-1}\epsilon_n = 2\phi$. Since there is also some error associated with the computed transformations associated with each pose, the angle must also exceed this error. By the previous analysis, if we use a single measurement to determine the rotation matrix R, then this error is bounded by $|4\phi|$ and hence, we have the bound

$$\alpha \geq 2\phi + 2(4\phi) = 10\phi.$$

For most values of ϕ, this bound is far too large to be of much use.

If we use several measurements to compute R, however, then more effective bounds can be used. As shown previously, assuming no systematic error implies that the error in the computed surface normal associated with a transformed face is given by a zero-mean normal distribution with standard deviation

$$\sqrt{\frac{8}{3n}}\,\phi$$

where n is the number of measurement samples.

This gives us a tighter definition of distinguishable surface normals. In particular, if α denotes the angle between surface normals associated with two distinct poses, those poses are distinguishable if

$$\alpha \geq 2\phi + 2\rho\phi.$$

The first term denotes the range of possible error in the measurement of the surface normals, and the second term denotes the range of possible error in the expected values of the surface normals. Here, ρ is a scale factor that is a function of the reliability of the error bound. That is, $\rho(c)$ denotes the point in the normal distribution described above such that c percent of the weight of the distribution lies below the value ρ.

For example, if the cutoff on the reliability of the bound is 0.95, and the number of measurements involved in computing the transformation is at least 10, then $\rho \leq 1.01$ and thus the bound on two surface normals being distinguishable is

$$4.02\phi.$$

We can also derive a formal definition of *distinguishability* based on position measurements. We first note that if a face is defined by the pair $(\hat{\mathbf{n}}_m, d)$ in model coordinates, such that a point $\mathbf{v}$ lies on the plane of the face if

$$\mathbf{v} \cdot \hat{\mathbf{n}}_m - d = 0$$

then the same face, after transformation, is defined by the pair $(\hat{\mathbf{n}}'_m, d')$, where

$$\begin{aligned} \hat{\mathbf{n}}'_m &= R\hat{\mathbf{n}}_m \\ d' &= d + (\mathbf{v}_0 \cdot R\hat{\mathbf{n}}_m)\,. \end{aligned}$$

We let the error associated with the computed value of $\hat{\mathbf{n}}'_m$ be denoted by $\mathbf{w}$ such that $\mathbf{w} \cdot \hat{\mathbf{n}}'_m = 0$. The magnitude of $\mathbf{w}$ can be bounded above by

$$|\sin(\rho\phi)|,$$

by the discussion above. We also let the error in computing $\mathbf{v}_0$ be denoted by $\mathbf{u}$. We now seek a bound on the possible errors in computing the point of intersection of a sensing ray with an object face, due to errors in the computed transformation.

Suppose that the sensing ray is given by $\alpha\hat{\mathbf{s}} + \mathbf{o}$ where $\hat{\mathbf{s}}$ is a specified unit vector, $\mathbf{o}$ is a specified offset vector orthogonal to $\hat{\mathbf{s}}$, and α is the free parameter specifying position along the sensing ray. The correct parameter of intersection of the sensing ray with the transformed face is given by the value of α such that

$$(\alpha\hat{\mathbf{s}} + \mathbf{o}) \cdot \hat{\mathbf{n}}'_m - d' = 0$$

or

$$\alpha_t = \frac{d' - \mathbf{o} \cdot \hat{\mathbf{n}}'_m}{\hat{\mathbf{s}} \cdot \hat{\mathbf{n}}'_m}.$$

On the other hand, if we include the potential error in the computed transformation, then the point of intersection is given by

$$\alpha_m = \frac{\left[d' + \mathbf{u} \cdot \left(\hat{\mathbf{n}}'_m + \mathbf{w}\right) + \mathbf{v}_0 \cdot \mathbf{w}\right] - \mathbf{o} \cdot \left(\hat{\mathbf{n}}'_m + \mathbf{w}\right)}{\hat{\mathbf{s}} \cdot \left(\hat{\mathbf{n}}'_m + \mathbf{w}\right)}$$

and thus the difference is given by

$$r = \frac{\mathbf{u} \cdot \hat{\mathbf{n}}'_m + (\mathbf{u} + \mathbf{v}_0 - \mathbf{o}) \cdot \mathbf{w}}{\hat{\mathbf{s}} \cdot \hat{\mathbf{n}}'_m + \hat{\mathbf{s}} \cdot \mathbf{w}} - \frac{\hat{\mathbf{s}} \cdot \mathbf{w}}{\hat{\mathbf{s}} \cdot \hat{\mathbf{n}}'_m + \hat{\mathbf{s}} \cdot \mathbf{w}} \alpha_t.$$

As a consequence, we can bound the error in the expected intersection point of the sensing ray with the face by

$$r \leq \frac{|\mathbf{u}| + (|\mathbf{u}| + |\mathbf{v}_0 - \mathbf{o}|)|\mathbf{w}|}{|\hat{\mathbf{s}} \cdot \hat{\mathbf{n}}'_m| - |\mathbf{w}|} + \frac{|\mathbf{w}|}{|\hat{\mathbf{s}} \cdot \hat{\mathbf{n}}'_m| - |\mathbf{w}|} \alpha_t,$$

where

$\mathbf{w}$ is the error in computing $\hat{\mathbf{n}}'_m = R\hat{\mathbf{n}}_m$

α_t is the predicted intersection point

$\mathbf{u}$ is the error in $\mathbf{v}_0$

$\hat{\mathbf{s}}$ is the sensing direction

$\mathbf{o}$ is the sensing offset vector

$\hat{\mathbf{n}}'_m$ is the face normal in transformed coordinates

$\mathbf{v}_0$ is the computed translation.

Thus, given this bound, two poses are *distinguishable* if their expected points of intersection are large enough,

$$|\alpha_1 - \alpha_2| \geq 2\epsilon_d + r_1 + r_2.$$

As in the case of distinguishing on the basis of surface normals, the bounds for $\mathbf{w}$ and $\mathbf{u}$ may be too large to be practical. We can reduce these bounds by using several measurements to determine a value for $\mathbf{v}_0$. As in the previous case, this will lead to a zero-mean normal distribution of expected error, and the effective range of error will be reduced.

Finally, we need to place bounds on the minimum Chebychev values needed to guarantee that perturbations in the computed transformations will not cause the sensing ray to miss the intended face. Let c denote the Chebychev value associated with a

particular face, whose transformed unit surface normal is $\hat{\mathbf{n}}'_m$, and where the unit sensing ray is given by $\hat{\mathbf{s}}$. Then the modified Chebychev value in the sensing plane is given by $c\left|\hat{\mathbf{n}}'_m \cdot \hat{\mathbf{s}}\right|$. At the same time, the variation in the position of a point on the face, as a function of error in the computed transformation, is given by

$$\Delta\mathbf{p} = \mathbf{u} + \mathbf{q}$$

where, as above, $\mathbf{u}$ denotes the error in the computed value of $\mathbf{v}_0$ and $\mathbf{q}$ is the error in the computed value of $R\mathbf{p}$, where $\mathbf{p}$ is the Chebychev point in model coordinates. The magnitudes of these error vectors are bounded by the expressions derived above. Since the directions of the error vectors are arbitrary, the condition on the Chebychev value required to ensure contact with the face is

$$c \geq \frac{|\mathbf{u}| + |\mathbf{q}|}{\left|\hat{\mathbf{n}}'_m \cdot \hat{\mathbf{s}}\right|}.$$

Thus, we have derived conditions on the parameters of the disambiguation algorithm needed to guarantee the performance of the algorithm.

Discussion and Examples

When Do We Compute the Sensing Directions?

We have described a technique for determining optimal sensing directions. We have still to consider, however, how to interface such a technique with the general problem of recognition and localization. The simplest method is to obtain some initial set of sensory data points, apply our recognition technique, and then use the disambiguation process as required, based on the current set of consistent poses. For example, if there are several consistent poses, we could choose the first pair, compute an optimal sensing direction based on that pair and obtain a new data point. Then, we could determine which of the set of poses are also consistent with the new data point and iterate. This technique, while applicable to arbitrary sets of objects, has the disadvantage of high computational expense.

In situations in which a large number of objects are possible, we may not be able to do any better than to compute sensing points as needed, based on the current set of feasible poses. In situations involving a single object, however, there may be an alternative method for integrating the computation of sensing positions with the interpretation of the sensory data.

In particular, given the analysis developed here, one can precompute optimal sensing rays as a function of the difference in transformation associated with two poses. Take any pair of poses of an object. There exists a rigid transformation taking one pose into the other, which we can parameterize in some fashion. We compute the optimal sensing direction for this pair of poses, and insert it into a lookup table, whose dimensions are indexed by the parameters of the relative transformation. Since the workspace of the sensory system is bounded, this is a bounded table (that is, the translational degrees of freedom are not infinite in extent). The analysis can be used to compute an optimal sensing ray corresponding to each entry of the table, where the parameters of the transformation are quantized to some desired level.

Now, when attempting to disambiguate two possible poses, one simply computes the difference in the transformations, looks up of the precomputed sensing ray in the appropriate slot of the table, transforms that ray by the transformation associated with the first pose, and then senses along that ray to obtain an new data point. That data point is added to the current set of sensory data, and the recognition and localization process is applied. If a unique pose results, the process is stopped; if not, a new sensing ray is obtained and the process continues.

By precomputing the sensing rays, we can avoid the computational expense associated with finding a new sensing position, and at the same time take advantage of the efficiency of the technique is disambiguating multiple poses.

Avoiding False Negatives

We have seen in the previous discussion that the analytic error bounds on the computed transformations for any pose are probably too large to be practical. We argued that one way to reduce these bounds was to use several measurements in the computation of the transformation. This led to a normal distribution of error in each of the components of the transformation, and thus, given a level of desired confidence in the algorithm, tighter bounds on the parameters were possible. In this case, we would expect that in general the algorithm will succeed, and we need only consider alterations to the algorithm to deal with the infrequent case when the errors in the computed transformation do exceed the expected thresholds. There are two situations that can arise in this case. The first is that the perturbation in the transform causes a surface normal to be sensed, that does not agree with any of the expected normals. This is essentially a false negative, since it implies that the poses are not distinguishable. The more damaging case is a false positive, in which the perturbation in the transformation results in a sensor measurement that coincidentally agrees with the wrong pose.

The easiest solution is to use more than one sense point. In this manner, false negatives are easily handled, since the expectation is that not all sensed points will give inconsistent data. This will be especially true if several sensing directions are used, in particular if the sensing directions are orthogonal. As well, it is likely that false positives can also be detected, since the expectation is that the correct pose will be found by most sensor points, again especially if several directions are used, and a simple voting scheme will arrive at the correct answer.

Testing the Algorithm

We have implemented the described technique, and tested it on a number of examples. Because the worst case bounds are so large, we used the approximations described above, with the expectation that on occasion an incorrect decision would be made, but that such errors could be avoided by voting over several additional sensing points.

In particular, we ran the algorithm described in [Grimson and Lozano-Pérez 84] for an object in arbitrary orientation relative to the sensors and with simulated sensing from three orthogonal directions. Whenever there was an ambiguity in interpreting the sensed data, we used the following disambiguation technique. We used the analysis developed above to predict a sensing ray, and for each pose we predicted ranges of expected values

for the sensory data along that ray. We then acquired an additional sense point along the chosen sensing ray, and compared the recorded value with the expected ranges to choose a pose.

Using a variety of simulated sensing errors, the disambiguation technique was applied to 1000 ambiguous cases. It was found in 336 of these cases that, due to the large errors inherent in the sensory data, the algorithm could not distinguish reliably between the possible solutions. In all of these cases, the poses differed by the reassignment of one data point from one face to an adjacent face, and this resulted in nearly identical transformations associated with the poses. Relative to the error resolution of the sensing devices, these can be considered to be identical solutions. In 633 of the cases, the disambiguation algorithm was able to determine the correct pose with only a single additional sensory point. In the remaining 31 cases, the algorithm chose an incorrect pose from the set of consistent poses.

We also ran a second version of the disambiguation algorithm on the same set of data. In this case, rather than using predicted range of values to choose a pose, we simply used the technique to generate the next sensing direction, and then ran the RAF recognition algorithm [Grimson and Lozano-Pérez 84, 85a] with that sensory point added to the original set of sensory data. In this case, we found that the algorithm identified the correct pose in all 664 cases, with only from 1 to 3 additional sensory points required to complete the identification.

Acknowledgments

Tomás Lozano-Pérez was critical to the development of this work, both in terms of the underlying recognition technique and in terms of valuable comments and criticisms of the material presented here. John Hollerbach and Rodney Brooks provided useful comments on earlier drafts.

References

Baker, H. H. and T. O. Binford "Depth from Edge and Intensity Based Stereo" *Seventh International Joint Conference on Artificial Intelligence*, August 1981, 631–636

Ballard, D. H. " Generalizing the Hough transform to detect arbitrary shapes" *Pattern Recognition* 13(2) (1981) 111–122.

Barnard, S. T. and Thompson, W. B. "Disparity analysis of images," *IEEE Pattern Analysis and Machine Intelligence* **PAMI-2**, 4, (1980), 333-340.

Brady, M. "Smoothed local symmetries and frame propagation." *Proc. IEEE Pattern Recog. and Im. Proc.*. (1982)

Brooks, R. "Symbolic reasoning among 3-dimensional models and 2-dimensional images." *Artificial Intell.* 17 (1981) 285–349.

Brou, P. "Using the gaussian image to find orientations of objects." *Int. J. Robotics Res.* 3(4) (1984) 89–125.

Bolles, R. C., and Cain, R. A. "Recognizing and locating partially visible objects: The Local-Feature-Focus method" *Int. J. Robotics Res.* 1(3) (1982) 57–82.

Bolles, R. C., Horaud, P., and Hannah, M. J. "3DPO: A three-dimensional part orientation system" Paper delivered at First International Symposium of Robotics Research, Bretton Woods, N.H. 1983. (Also in Robotics Research: The First International Symposium, edited by M. Brady and R. Paul, MIT Press, 1984, pp. 413–424.)

Faugeras, O. D., and Hebert, M. "A 3-D recognition and positioning algorithm using geometrical matching between primitive surfaces." *Proc. Eighth Int. Joint Conf. Artificial Intell.* Los Altos: William Kaufmann, pp. 996–1002. (Aug. 1983, Karlsruhe, W. Germany).

Gaston, P. C. and T. Lozano-Pérez "Tactile Recognition and Localization Using Object Models: The case of polyhedra on a plane." *IEEE Pattern Analysis and Machine Intelligence* **PAMI-6**, 3, (1984), 257–265.

Grimson, W. E. L. "A Computer Implementation of a Theory of Human Stereo Vision" *Philosophical Transactions of the Royal Society of London, B* **292** (1981), 217–253

Grimson W. E. L. "The Combinatorics of Local Constraints in Model-Based Recognition and Localization from Sparse Data", *MIT Artificial Intelligence Laboratory Memo 763*, 1984.

Grimson, W. E. L. "Computational Experiments with a Feature-Based Stereo Algorithm" *IEEE Transactions on Pattern Analysis and Machine Intelligence* **7** (1985), 17–34.

Grimson, W. E. L. and T. Lozano-Pérez "Model-Based Recognition and Localization From Sparse Range or Tactile Data" *Int. Journ. Robotics Research* **3** (1984), 3–35.

Grimson, W. E. L., and T. Lozano-Pérez "Recognition and localization of overlapping parts from sparse data" AIM-841. Cambridge, Mass.:Massachusetts Institute of Technology Artificial Intelligence Laboratory. (1985a)

Grimson, W. E. L., and T. Lozano-Pérez "Recognition and localization of overlapping parts from sparse data in two and three dimensions" *Proc. IEEE Intern. Conf. on Robotics and Automation.* Silver Spring: IEEE Computer Society Press, pp. 61–66. (Mar. 1985b, St. Louis, MO).

Harmon, L. D "Automated Tactile Sensing" *Int. Journ. Robotics Research* **1** (1982) 3–32.

Hillis, W. D. "A High-Resolution Image Touch Sensor" *Int. Journ. Robotics Research* **1** (1982) 33–44.

Holland, S. W. "A programmable computer vision system based on spatial relationships." General Motors Publ. GMR-2078. Detroit: General Motors. (1976 Feb.)

Horn, B. K. P. "Extended Gaussian images." AIM-740. Cambridge, Mass.:M I T Artificial Intelligence Laboratory. (1983)

Horn, B. K. P., and Ikeuchi, K. "Picking parts out of a bin." AIM-746. Cambridge, Mass.:M I T Artificial Intelligence Laboratory. (1983)

Ikeuchi, K. "Determining attitude of object from needle map using extended gaussian image." AIM-714. Cambridge, Mass.:M I T Artificial Intelligence Laboratory. (1983)

Ikeuchi, K. and B. K. P. Horn "An Application of Photometric Stereo" *Sixth Intl. Joint Conf. on Artificial Intelligence* (1979) 413–415.

Lewis, R. A. and A. R. Johnston "A Scanning Laser Range Finder for a Robotic Vehicle" *Fifth Intl. Joint Conf. on Artificial Intelligence* (1977) 762–768.

Marr, D., and Nishihara, H. K. "Representation and recognition of the spatial organization of three-dimensional shapes." *Proc. R. Soc. Lond. B* 200 (1978) 269–294.

Marr, D. and T. Poggio "A Computational Theory of Human Stereo Vision" *Proc. R. Soc. Lond. B* **204** (1979) 310–328.

Mayhew, J.E.W. and J.P. Frisby "Psychophysical and Computational Studies towards a Theory of Human Stereopsis" *Artificial Intelligence* **17** (1981) 349–385.

Nevatia, R. "Structured descriptors of complex curved objects for recognition and visual memory." Ph.D. thesis, Stanford University. AIM 250. Stanford, Calif.: Stanford University Artificial Intelligence Laboratory. (1974)

Nevatia, R., and Binford, T. O. "Description and recognition of curved objects." *Artificial Intell.* 8 (1977) 77–98.

Nitzan, D., A. E. Brain, and R. O. Duda "The Measurement and Use of Registered Reflectance and Range Data in Scene Analysis" *Proc. of IEEE* **65** (February 1977) 206–220.

Ohta, Y. and Kanade, T. "Stereo by intra- and inter-scanline search using dynamic programming," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-7** 2 (1985) 139–155.

Overton, K.J. and T. Williams "Tactile Sensation for Robots" *Seventh Intl. Joint Conf. on Artificial Intelligence* (1981) 791–795.

Perkins, W. A. "A model-based vision system for industrial parts" *IEEE Trans. Comput.* C-27 (1978) 126–143

Popplestone, R. J., C. M. Brown, A. P. Ambler, and G. F. Crawford "Forming Models of Plane and Cylinder Faceted Bodies from Light Stripes" *Fourth Intl. Joint Conf. on Artificial Intelligence* Tbilisi, Georgia, USSR (September 1975) 664–668.

Purbrick, J. A. "A Force Transducer Employing Conductive Silicone Rubber" *First International Conference on Robot Vision and Sensory Controls* Stratford-upon-Avon, United Kingdom (April, 1981).

Raibert, M. H. and J. E. Tanner "Design and Implementation of a VLSI Tactile Sensing Computer" *Int. Journ. Robotics Research* **1** (1982) 3–18.

Schneiter, J. L. *An Optical Tactile Sensor for Robots* S.M. Thesis, Dept. of Mech. Engr., Massachusetts Institute of Technology, August 1982.

Shirai, Y. and M. Suwa "Recognition of Polyhedrons with a Range Finder" *Second Intl. Joint Conf. on Artificial Intelligence* (1971).

Stockman, G., and Esteva, J. C. "Use of geometrical constraints and clustering to determine 3D object pose." TR84-002. East Lansing, Mich.:Michigan State University Department of Computer Science. (1984)

Sugihara, K. "Range-data analysis guided by a junction dictionary." *Artificial Intell.* 12, (1979) 41–69.

Tsuji, S., and Nakamura, A. "Recognition of an object in a stack of industrial parts" *Proc. Fourth Int. Joint Conf. Artificial Intell.* Los Altos: William Kaufmann (1975) (Aug., Cambridge, Mass.) pp. 811–818

Woodham, R. J. "Photometric Stereo: A Reflectance Map Technique for Determining Surface Orientation from Image Intensity" *Image Understanding Systems and Industrial Applications, Proc. SPIE* **155** (1978).

Woodham, R. J. "Photometric Method for Determining Surface Orientation from Multiple Images" *Optical Engineering* **19** (1980) 139–144.

Woodham, R. J. "Analysing Images of Curved Objects" *Artificial Intelligence* **17** (1981) 117–140.

Appendix

In the appendix, we present a more detailed error analysis of the computation of the transformation from model to sensor coordinates.

Errors in $\hat{\mathbf{r}}$

We begin by considering the range of possible errors in the computation of the direction of rotation, $\hat{\mathbf{r}}$. By the analysis of [Grimson and Lozano-Pèrez 84], the rotation direction $\hat{\mathbf{r}}$ is computed by taking two pairs $(\hat{\mathbf{n}}_m, \hat{\mathbf{n}}'_m)$, where $\hat{\mathbf{n}}_m$ is the unit normal of a face of the model, and $\hat{\mathbf{n}}'_m$ is the same unit normal rotated into sensor coordinates, and letting $\hat{\mathbf{r}}$ be the unit vector in the direction of

$$(\hat{\mathbf{n}}_{m,i} - \hat{\mathbf{n}}'_{m,i}) \times (\hat{\mathbf{n}}_{m,j} - \hat{\mathbf{n}}'_{m,j}).$$

We assume that we are given $\hat{\mathbf{n}}_{m,i}, \hat{\mathbf{n}}'_{m,i}$ and that the sensitivity of the sensor to errors in surface orientation is given by ϵ_n. That is, if $\hat{\mathbf{n}}'_{m,i}$ is the correct surface normal transformed into sensor coordinates and $\hat{\mathbf{n}}_s$ is the actual measured (or sensed) surface normal, then

$$\hat{\mathbf{n}}_s \cdot \hat{\mathbf{n}}'_{m,i} \geq \epsilon_n. \tag{4}$$

We will consider two stages in deriving bounds on the error in computing $\hat{\mathbf{r}}$. If we let

$$\hat{\mathbf{v}}_i = \frac{\hat{\mathbf{n}}_{m,i} - \hat{\mathbf{n}}'_{m,i}}{|\hat{\mathbf{n}}_{m,i} - \hat{\mathbf{n}}'_{m,i}|},$$

and

$$\hat{\mathbf{u}}_i = \frac{\hat{\mathbf{n}}_{m,i} - \hat{\mathbf{n}}_{s,i}}{|\hat{\mathbf{n}}_{m,i} - \hat{\mathbf{n}}_{s,i}|},$$

then the correct value for $\hat{\mathbf{r}}$ is given by

$$\hat{\mathbf{r}}_t = \frac{\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j}{\sqrt{1 - (\hat{\mathbf{v}}_i \cdot \hat{\mathbf{v}}_j)^2}}$$

and the computed value is given by

$$\hat{\mathbf{r}}_c = \frac{\hat{\mathbf{u}}_i \times \hat{\mathbf{u}}_j}{\sqrt{1 - (\hat{\mathbf{u}}_i \cdot \hat{\mathbf{u}}_j)^2}}.$$

We will first derive bounds on $\hat{\mathbf{v}}_i \cdot \hat{\mathbf{u}}_i$ and then use the result to bound $\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_c$.

The vector $\hat{\mathbf{n}}_s$ can be represented by the following parameterization

$$\hat{\mathbf{n}}_s = \alpha\hat{\mathbf{n}}'_m + \beta\hat{\mathbf{n}}_m + \delta(\hat{\mathbf{n}}_m \times \hat{\mathbf{n}}'_m).$$

Then equation (4) produces the inequality

$$\alpha + \beta\gamma \geq \epsilon_n \tag{4'}$$

where $\gamma = \hat{\mathbf{n}}_m \cdot \hat{\mathbf{n}}'_m$. We will consider the worst case, in which equality holds. Furthermore, the fact that $\hat{\mathbf{n}}_s$ is a unit vector yields the following constraint

$$\alpha^2 + 2\alpha\beta\gamma + \beta^2 + \delta^2 = 1. \tag{5}$$

Given $\hat{\mathbf{n}}_m, \hat{\mathbf{n}}'_m$, and $\hat{\mathbf{n}}_s$, we first consider the range of possible values for $\hat{\mathbf{n}}_m - \hat{\mathbf{n}}_s$, relative to $\hat{\mathbf{n}}_m - \hat{\mathbf{n}}'_m$, that is, we want bounds on the range of possible values for

$$E = \hat{\mathbf{v}} \cdot \hat{\mathbf{u}} = \frac{(\hat{\mathbf{n}}_m - \hat{\mathbf{n}}'_m) \cdot (\hat{\mathbf{n}}_m - \hat{\mathbf{n}}_s)}{|\hat{\mathbf{n}}_m - \hat{\mathbf{n}}'_m|\,|\hat{\mathbf{n}}_m - \hat{\mathbf{n}}_s|}.$$

It is straightforward to show that

$$|\hat{\mathbf{n}}_m - \hat{\mathbf{n}}'_m| = \sqrt{2(1-\gamma)}.$$

Furthermore, using equation (5), one can show that

$$|\hat{\mathbf{n}}_m - \hat{\mathbf{n}}_s| = \sqrt{2\,(1-\beta-\alpha\gamma)}.$$

Finally, expanding out the dot product and substituting yields

$$E = \frac{(1-\beta+\alpha)(1-\gamma)}{2\sqrt{1-\gamma}\sqrt{1-\beta-\alpha\gamma}}.$$

By equation (4'), $\alpha = \epsilon_n - \beta\gamma$, and substitution yields

$$E = \frac{\sqrt{1-\gamma}\,[1+\epsilon_n-\beta(1+\gamma)]}{2\sqrt{1-\epsilon_n\gamma-\beta(1-\gamma^2)}}.$$

The first problem to consider is what is the minimum value for E as β varies. In particular, we find that

$$\frac{\partial E}{\partial \beta} = \frac{\sqrt{1-\gamma}\,(1+\gamma)^2}{4} \quad \frac{[\beta\,(1-\gamma)-(1-\epsilon_n)]}{(1-\epsilon_n\gamma-\beta\,(1-\gamma^2))^{\frac{3}{2}}}.$$

This is zero when

$$\beta = \frac{1-\epsilon_n}{1-\gamma} \tag{6}$$

and this is a valid value for β provided $\gamma < \epsilon_n$. Taking a second partial derivative of E, we find that the sign of $\partial^2 E/\partial\beta^2$ is given by the sign of

$$\beta\left(1-\gamma^2\right) + 3\left(\epsilon_n - \gamma\right) + \epsilon_n\gamma - 1.$$

Substituting equation (6), we find that the sign of the second partial derivative is given by the sign of

$$2\left(\epsilon_n - \gamma\right)$$

and this is positive, since $\gamma < \epsilon_n$. Hence, E achieves a minimum at the value of β given by equation (6), and this value is

$$E = \sqrt{\frac{\epsilon_n - \gamma}{1-\gamma}}. \tag{47}$$

If $\gamma > \epsilon_n$, then the minimum value for E occurs for β at the limit of its range, namely $\beta = 1$. In this case, $E \leq 0$, and is minimized when $\gamma = \sqrt{\epsilon_n}$, taking the value

$$E = \frac{-\left(1-\sqrt{\epsilon_n}\right)}{2}.$$

In general, we will try to restrict the computation of $\hat{\mathbf{r}}$ to those cases in which $\gamma < \epsilon_n$, in order to keep the magnitude of the possible error in $\hat{\mathbf{r}}$ small. Note that the minimum value for E is monotonic in γ, that is, the minimum E increases as γ decreases towards -1.

Thus, we obtain the bound

$$\begin{aligned}\hat{\mathbf{u}}_i \cdot \hat{\mathbf{v}}_i &\geq \sqrt{\frac{\epsilon_n - \gamma_i}{1 - \gamma_i}} \\ &= \delta_i\end{aligned}$$

where

$$\gamma_i = \left(\hat{\mathbf{n}}_{m,i} \cdot \hat{\mathbf{n}}'_{m,i}\right).$$

We are now interested in obtaining bounds on $\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_c$. In essence, we have two cones in the Gaussian sphere, centered about $\hat{\mathbf{v}}_i$ and $\hat{\mathbf{v}}_j$, of radius δ_i and δ_j respectively. The possible values of $\hat{\mathbf{r}}_c$ are given by the normalized cross products of vectors within these cones. Clearly, if the cones overlap, then the computation for $\hat{\mathbf{r}}$ is unstable. We avoid this case by requiring that the cones do not overlap.

Note that if all the error in the computation of either $\hat{\mathbf{u}}_i$ or $\hat{\mathbf{u}}_j$ lies in the plane spanned by $\hat{\mathbf{v}}_i$ and $\hat{\mathbf{v}}_j$, then the normalization of the cross product will result in the correct value $\hat{\mathbf{r}}_t = \hat{\mathbf{r}}_c$. Clearly the maximum deviation of $\hat{\mathbf{r}}_c$ from $\hat{\mathbf{r}}_t$ will occur when the error between $\hat{\mathbf{u}}_i$ and $\hat{\mathbf{v}}_i$ and the error between $\hat{\mathbf{u}}_j$ and $\hat{\mathbf{v}}_j$ lie maximally separated from this plane. This requires that we check two cases, one in which the errors lie on the same side of the plane and one in which the errors lie on opposite sides of the plane. We now consider the first case.

Let $\eta = \hat{\mathbf{v}}_i \cdot \hat{\mathbf{v}}_j$. Then

$$\hat{\mathbf{u}}_i = \delta_i \hat{\mathbf{v}}_i + \sqrt{\frac{1 - \delta_i^2}{1 - \eta^2}} \left(\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j\right)$$

$$\hat{\mathbf{u}}_j = \delta_j \hat{\mathbf{v}}_j + \sqrt{\frac{1 - \delta_j^2}{1 - \eta^2}} \left(\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j\right)$$

$$\hat{\mathbf{u}}_i \times \hat{\mathbf{u}}_j = \delta^2 \left(\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j\right) + \delta_j \sqrt{\frac{1 - \delta_i^2}{1 - \eta^2}} \left[\left(\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j\right) \times \hat{\mathbf{v}}_j\right] + \delta_i \sqrt{\frac{1 - \delta_j^2}{1 - \eta^2}} \left[\hat{\mathbf{v}}_i \times \left(\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j\right)\right].$$

Thus

$$\left(\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j\right) \cdot \left(\hat{\mathbf{u}}_i \times \hat{\mathbf{u}}_j\right) = \delta_i \delta_j \left[1 - \eta^2\right]$$

and

$$\hat{\mathbf{u}}_i \cdot \hat{\mathbf{u}}_j = \delta_i \delta_j \eta + \sqrt{1 - \delta_i^2} \sqrt{1 - \delta_j^2}.$$

Then, by substitution,

$$\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_c \geq \frac{\delta_i \delta_j \sqrt{1 - \eta^2}}{\sqrt{1 - \left\{\delta_i \delta_j \eta + \sqrt{1 - \delta_i^2} \sqrt{1 - \delta_j^2}\right\}^2}}. \tag{8}$$

In the second case, we change $\hat{\mathbf{u}}_j$ to

$$\hat{\mathbf{u}}_j = \delta_j \hat{\mathbf{v}}_j - \sqrt{\frac{1 - \delta_j^2}{1 - \eta^2}} \left(\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j\right)$$

and thus

$$\hat{\mathbf{u}}_i \times \hat{\mathbf{u}}_j = \delta_j \delta_i \left(\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j\right) + \delta_j \sqrt{\frac{1-\delta_i^2}{1-\eta^2}} \left[\left(\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j\right) \times \hat{\mathbf{v}}_j\right] - \delta_i \sqrt{\frac{1-\delta_j^2}{1-\eta^2}} \left[\hat{\mathbf{v}}_i \times \left(\hat{\mathbf{v}}_i \times \hat{\mathbf{v}}_j\right)\right].$$

Following through the same algebra leads to a bound on the dot product of

$$\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_c \geq \frac{\delta_i \delta_j \sqrt{1-\eta^2}}{\sqrt{1 - \left\{\delta_i \delta_j \eta - \sqrt{1-\delta_i^2}\sqrt{1-\delta_j^2}\right\}^2}}, \tag{9}$$

where

$$\eta = \hat{\mathbf{v}}_i \cdot \hat{\mathbf{v}}_j$$

$$\hat{\mathbf{v}}_i = \frac{\hat{\mathbf{n}}_{m,i} - \hat{\mathbf{n}}'_{m,i}}{\left|\hat{\mathbf{n}}_{m,i} - \hat{\mathbf{n}}'_{m,i}\right|}$$

$$\delta_i \geq \sqrt{\frac{\epsilon_n - \gamma_i}{1-\gamma_i}}$$

$$\gamma_i = \hat{\mathbf{n}}_{m,i} \cdot \hat{\mathbf{n}}'_{m,i}.$$

It is straightforward to show that the bound in equation (9) is in fact smaller than the one in equation (8).

Note that if γ_i is close to 1, then the error bound comes increasingly large. This is to be expected, since in this case, $\hat{\mathbf{n}}_{m,i} \approx \hat{\mathbf{n}}'_{m,i}$ and thus small errors in the position of $\hat{\mathbf{n}}'_m$ can lead to large errors in the position of $\hat{\mathbf{r}}$. Similarly, if η is near 1, large errors can also result. If we restrict our computation (where possible) to cases where γ_i and η are small, then we have an approximate bound on the error in computing the direction of rotation given by

$$\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_c \geq \epsilon_n.$$

This bound is supported by the results of the simulations reported in [Grimson and Lozano-Pérez 84].

Errors in θ

We now want to consider bounds on the possible error in computing the remaining parameter of the rotation component of the transformation, namely, the angle of rotation θ. Given the expressions in equation (1) for $\cos\theta$ and $\sin\theta$, the value of θ is given by

$$\tan\theta = -\frac{\hat{\mathbf{n}}'_m \cdot \left(\hat{\mathbf{r}} \times \hat{\mathbf{n}}_m\right)}{\left(\hat{\mathbf{r}} \times \hat{\mathbf{n}}'_m\right) \cdot \left(\hat{\mathbf{r}} \times \hat{\mathbf{n}}_m\right)}$$

where $\hat{\mathbf{n}}_m$ is the unit normal of a face in model coordinates and $\hat{\mathbf{n}}'_m$ is the corresponding normal transformed into sensor coordinates.

As in the previous section, we let $\hat{\mathbf{r}}_t$ denote the true direction of rotation and $\hat{\mathbf{r}}_c$ the computed direction of rotation. We will assume that the error in the computed direction of rotation is bounded by

$$\hat{\mathbf{r}}_t \cdot \hat{\mathbf{r}}_c \geq \delta_r$$

and that the measured value for $\hat{\mathbf{n}}'_m$ is given by $\hat{\mathbf{n}}_s$ such that

$$\hat{\mathbf{n}}'_m \cdot \hat{\mathbf{n}}_s \geq \epsilon_n.$$

We shall make use of the following four unit vectors:

$$\hat{\mathbf{w}} = \frac{\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m}{|\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m|}$$

$$\hat{\mathbf{u}} = \frac{\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}_m}{|\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}_m|}$$

$$\hat{\mathbf{s}} = \frac{\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}'_m}{|\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}'_m|}$$

$$\hat{\mathbf{t}} = \frac{\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}'_m}{|\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}'_m|}.$$

Given these definitions, it is straightforward to show that

$$\tan\theta_t = \frac{-1}{|\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}'_m|} \frac{(\hat{\mathbf{n}}'_m \cdot \hat{\mathbf{w}})}{(\hat{\mathbf{s}} \cdot \hat{\mathbf{w}})}$$

$$\tan\theta_c = \frac{-1}{|\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}_s|} \frac{(\hat{\mathbf{n}}_s \cdot \hat{\mathbf{u}})}{(\hat{\mathbf{t}} \cdot \hat{\mathbf{u}})}.$$

Our method in obtaining bounds on the deviation between these two expressions will be to bound $\hat{\mathbf{n}}_s \cdot \hat{\mathbf{u}}$ as a function of $\hat{\mathbf{n}}'_m \cdot \hat{\mathbf{w}}$, and to bound $\hat{\mathbf{t}} \cdot \hat{\mathbf{u}}$ as a function of $\hat{\mathbf{s}} \cdot \hat{\mathbf{w}}$. Once we have bounds on these expressions, they can be combined to bound the overall expression for $\tan\theta$.

First, we consider the range of values for

$$\hat{\mathbf{u}} \cdot \hat{\mathbf{w}} = \frac{(\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}_m)}{|\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}_m|} \cdot \frac{(\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m)}{|\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m|}.$$

In considering the range of values for $\hat{\mathbf{u}} \cdot \hat{\mathbf{w}}$ we note that because of the normalization of the vectors, any error in $\hat{\mathbf{r}}_c$ lying in the $\hat{\mathbf{r}}_t - \hat{\mathbf{n}}_m$ plane will have no effect on the dot product. Thus, the worst case occurs when all of the error lies perpendicular to this plane. Hence, we need only consider the cases where

$$\hat{\mathbf{r}}_c = \alpha\hat{\mathbf{r}}_t \pm \beta\,(\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m),$$

where

$$\alpha \geq \delta_r$$

$$1 = \alpha^2 + \beta^2\left(1 - \cos^2\nu\right)$$

$$\cos\nu = \hat{\mathbf{r}}_t \cdot \hat{\mathbf{n}}_m.$$

Now, the worst case will occur for $\alpha = \delta_r$, in which case,

$$\beta^2 = \frac{1 - \delta_r^2}{1 - \cos^2\nu}$$

so that the worst case will arise for

$$\hat{\mathbf{r}}_c = \delta_r\hat{\mathbf{r}}_t \pm \sqrt{\frac{1 - \delta_r^2}{1 - \cos^2\nu}}\quad (\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m).$$

In this case, the following expressions hold:

$$\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}_m = \delta_r\,(\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m) \pm \sqrt{\frac{1 - \delta_r^2}{1 - \cos^2\nu}}\quad (\hat{\mathbf{n}}_m \times (\hat{\mathbf{n}}_m \times \hat{\mathbf{r}}_t))$$

$$(\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}_m) \cdot (\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}_m) = 1 - \delta_r^2\cos^2\nu$$

$$(\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m) \cdot (\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m) = 1 - \cos^2\nu$$

$$(\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}_m) \cdot (\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}_m) = \delta_r\left(1 - \cos^2\nu\right).$$

Thus, we have the bound

$$\hat{\mathbf{u}} \cdot \hat{\mathbf{w}} \geq \mu$$

where

$$\mu = \delta_r \sqrt{\frac{1 - \cos^2 \nu}{1 - \delta_r^2 \cos^2 \nu}}.$$

At this stage, we have $\hat{\mathbf{n}}_s \cdot \hat{\mathbf{n}}'_m \geq \epsilon_n$ and $\hat{\mathbf{u}} \cdot \hat{\mathbf{w}} \geq \mu$. We can visualize this situation by considering two cones in the Gaussian sphere, one centered about $\hat{\mathbf{n}}'_m$ with radius ϵ_n, and one centered about $\hat{\mathbf{w}}$ with radius μ. We are essentially asking for bounds on the range of dot products between vectors lying within these two different cones. Assuming that the cones do not overlap, the maximum and minimum dot products will occur for the minimum and maximum angles between elements of the cones, respectively, and this clearly occurs for vectors lying in the cones and lying in the $\hat{\mathbf{n}}'_m - \hat{\mathbf{w}}$ plane.

Suppose we denote:

$$\begin{aligned}
\cos \alpha &= \left(\hat{\mathbf{n}}'_m \cdot \hat{\mathbf{w}}\right) \\
\cos \phi &= \epsilon_n \\
\cos \psi &= \delta_r \\
\cos \xi &= \mu
\end{aligned}$$

Clearly, the extremal angles for these two cones are given by

$$\alpha \pm [\phi + \xi].$$

Thus, the range of possible values for

$$\hat{\mathbf{n}}_s \cdot \hat{\mathbf{w}}$$

is bounded by

$$\cos\left(\alpha \pm [\phi + \xi]\right).$$

An analogous argument can be made for the dot product $\hat{\mathbf{t}} \cdot \hat{\mathbf{u}}$. If we let

$$\begin{aligned}
\cos \beta &= \left(\hat{\mathbf{s}} \cdot \hat{\mathbf{w}}\right) \\
\cos \gamma &= \rho \\
\cos \omega &= \delta_r \epsilon_n \cos \nu - \sqrt{1 - \delta_r^2}\sqrt{1 - \epsilon_n^2} \\
&= \cos(\phi + \psi) - (1 - \cos \nu) \cos \phi \cos \psi \\
&= \left(\hat{\mathbf{r}}_c \cdot \hat{\mathbf{n}}_s\right)
\end{aligned}$$

where ρ is the bound

$$\rho = \delta_r \epsilon_n \sqrt{\frac{1 - \cos^2 \nu}{1 - \cos^2 \omega}}$$

then the range of possible values for

$$\hat{\mathbf{t}} \cdot \hat{\mathbf{u}}$$

is bounded by

$$\cos\left(\beta \pm [\psi + \gamma]\right).$$

Finally,

$$\begin{aligned}
\left|\hat{\mathbf{r}}_t \times \hat{\mathbf{n}}'_m\right| &= \sin \nu \\
\left|\hat{\mathbf{r}}_c \times \hat{\mathbf{n}}_s\right| &= \sin \omega.
\end{aligned}$$

Thus, by gathering all these expressions together, we obtain the following worst case expressions:

$$\tan\theta_t = \frac{-1}{\sin\nu}\frac{\cos\alpha}{\cos\beta}$$

$$\tan\theta_c = \frac{-1}{\sin\omega}\frac{\cos(\alpha-(\phi+\xi))}{\cos(\beta+(\psi+\gamma))}.$$

We note that this expression for the computed value of $\tan\theta$ has the expected limiting case behavior. In particular, if the error parameters $\phi = \psi = 0$, then the entire expression for $\tan\theta_c$ reduces to that for $\tan\theta_t$.

We now seek an estimate for the error in the computed value of θ, in the special case of ϕ and ψ both small. In particular, we would like an expression for $\Delta\theta$ such that

$$\tan(\theta_t+\Delta\theta) \approx \tan(\theta_c).$$

In this way, we can place a bound on the possible error in the computation of θ.

In the limiting case of ϕ and ψ small, the bound $\mu \approx \delta_r$ so that $\xi \approx \psi$. Furthermore, $\cos\omega \approx \cos\nu$ so that $\cos\gamma \approx \cos\phi\cos\psi \approx \cos(\phi+\psi)$ and hence, $\gamma \approx \phi+\psi$. As a consequence, finding an approximation for the deviation in $\tan\theta$ reduces to comparing the worst case of deviation between

$$\tan(\theta_c) = \frac{\cos(\alpha-[\phi+\psi])}{\cos(\beta+[\phi+2\psi])}$$

and

$$\tan(\theta_t) = \frac{\cos\alpha}{\cos\beta}. \tag{10}$$

By expansion,

$$\tan(\theta_t+\Delta\theta) = \frac{\tan\theta_t+\tan\Delta\theta}{1-\tan\theta_t\tan\Delta\theta}$$

and if we substitute $\Delta\theta \approx \phi+\psi$, and use equation (10), then this expression can be expanded into

$$\frac{\cos(\alpha-\phi-\psi)+[\cos\beta-\sin\alpha]\sin(\phi+\psi)}{\cos(\beta+\phi+\psi)+[\sin\beta-\cos\alpha]\sin(\phi+\psi)}. \tag{11}$$

Now, if $\cos\beta \approx \sin\alpha$, then the second term in both the numerator and denominator can be ignored, especially since $\sin(\phi+\psi)$ is also small. Requiring this to be true is equivalent to requiring that

$$(\hat{\mathbf{s}}\cdot\hat{\mathbf{w}})^2 + (\hat{\mathbf{n}}'_m\cdot\hat{\mathbf{w}})^2 \approx 1$$

that is, that the component of the unit vector $\hat{\mathbf{w}}$ in the direction of $\hat{\mathbf{n}}'_m\times\hat{\mathbf{s}}$ be small. It is straightforward to show that

$$\left(\hat{\mathbf{w}}\cdot(\hat{\mathbf{n}}'_m\times\hat{\mathbf{s}})\right)^2 = [\cot\nu\,(\hat{\mathbf{n}}_m\cdot\hat{\mathbf{s}})]^2.$$

Since we have already indicated that we will restrict our computation of the transformation parameters to those cases in which $\hat{\mathbf{r}}\cdot\hat{\mathbf{n}}_m << 1$ it follows that $\cot\nu$ is small and the second terms in both the numerator and denominator in equation (11) can be disregarded.

By dropping these terms, we see that the remaining expression reduces to

$$\tan(\theta_t+[\phi+\psi]) \approx \frac{\cos(\alpha-[\phi+\psi])}{\cos(\beta+[\phi+\psi])}.$$

Thus, if ψ is small enough, it follows that the worst case deviation is given by $\theta_c \approx \theta_t+(\phi+\psi)$ and hence that a good approximation to the error, $\Delta\theta$, in the computed

value of the rotation, θ, is given by

$$\Delta\theta \approx |\phi + \psi| .$$

Errors in $R\mathbf{v}$

We have computed expressions for the possible error in $\hat{\mathbf{r}}$ and θ. In particular, we will denote the error in θ by $\Delta\theta$ and the vector error in $\hat{\mathbf{r}}$ by $\delta\hat{\mathbf{r}}$ such that $\hat{\mathbf{r}} \cdot \delta\hat{\mathbf{r}} = 0$. We now consider the problem of estimating bounds on the possible error in applying the computed rotation matrix to an arbitrary vector $\mathbf{v}$. We know that the rotational component of the transformation of $\mathbf{v}$ is given by

$$R(\hat{\mathbf{r}}, \theta)\mathbf{v} = \cos\theta\mathbf{v} + (1 - \cos\theta)(\hat{\mathbf{r}} \cdot \mathbf{v})\hat{\mathbf{r}} + \sin\theta(\hat{\mathbf{r}} \times \mathbf{v})$$

where $\hat{\mathbf{r}}$ and θ are the parameters determining the rotation.

We first consider the variation of this expression with respect to the angle of rotation. In particular, under the assumption that $\Delta\theta$ is small, the following holds:

$$\begin{aligned} R(\hat{\mathbf{r}}, \theta + \Delta\theta)\mathbf{v} - R(\hat{\mathbf{r}}, \theta)\mathbf{v} = |\mathbf{v}| \{ &(\cos(\theta + \Delta\theta) - \cos\theta)\hat{\mathbf{v}} \\ &+ (\cos\theta - \cos(\theta + \Delta\theta))(\hat{\mathbf{r}} \cdot \hat{\mathbf{v}})\hat{\mathbf{r}} \\ &+ (\sin(\theta + \Delta\theta) - \sin\theta)(\hat{\mathbf{r}} \times \hat{\mathbf{v}})\} \\ \approx |\mathbf{v}| \{ &-\Delta\theta\sin\theta\hat{\mathbf{v}} + \Delta\theta\sin\theta(\hat{\mathbf{r}} \cdot \hat{\mathbf{v}})\hat{\mathbf{r}} + \Delta\theta\cos\theta(\hat{\mathbf{r}} \times \hat{\mathbf{v}})\} . \end{aligned}$$

Straightforward algebraic manipulation shows that the magnitude of this term is given by

$$|\mathbf{v}| \Delta\theta \sqrt{1 - (\hat{\mathbf{r}} \cdot \hat{\mathbf{v}})^2}$$

and this is bounded above by $\Delta\theta |\mathbf{v}|$.

Next, we consider the variation with respect to $\hat{\mathbf{r}}$, so that

$$\begin{aligned} R(\hat{\mathbf{r}} + \delta\hat{\mathbf{r}}, \theta) - R(\hat{\mathbf{r}}, \theta) = |\mathbf{v}| \{ &(1 - \cos\theta)[(\hat{\mathbf{r}} \cdot \hat{\mathbf{v}})\delta\hat{\mathbf{r}} + (\delta\hat{\mathbf{r}} \cdot \hat{\mathbf{v}})[\hat{\mathbf{r}} + \delta\hat{\mathbf{r}}]] \\ &+ \sin\theta[\delta\hat{\mathbf{r}} \times \hat{\mathbf{r}}]\} . \end{aligned} \tag{2}$$

We consider the magnitude of the second term in the right hand side of this expression, by taking the dot product of this vector with itself. If we ignore terms in $(\delta\hat{\mathbf{r}} \cdot \delta\hat{\mathbf{r}})$, since the assumption of $\delta\hat{\mathbf{r}}$ small implies such terms are negligible, then the magnitude of the second term in equation (2) is given by

$$|(1 - \cos\theta)(\delta\hat{\mathbf{r}} \cdot \hat{\mathbf{v}}) + \sin\theta(\hat{\mathbf{v}} \cdot (\hat{\mathbf{r}} \times \delta\hat{\mathbf{r}}))| . \tag{3}$$

We now consider a bound for this expression. Suppose we let $\hat{\mathbf{k}}$ denote the unit vector in the direction of $\delta\hat{\mathbf{r}}$, and let $\left(\hat{\mathbf{k}} \cdot \hat{\mathbf{v}}\right) = \cos\varsigma$. Since the worst case will occur when $\hat{\mathbf{v}}$ lies entirely in the plane spanned by $\delta\hat{\mathbf{r}}$ and $\hat{\mathbf{r}} \times \delta\hat{\mathbf{r}}$, equation (3) reduces to

$$|\delta\hat{\mathbf{r}}| \, |(1 - \cos\theta)\cos\varsigma + \sin\theta\sin\varsigma| = |\delta\hat{\mathbf{r}}| \, |\cos\varsigma - \cos(\theta + \varsigma)| .$$

It is clear that the worst possible value for this expression is $2|\delta\hat{\mathbf{r}}|$. Thus, the maximum value for the magnitude of the second term in equation (2) is $2|\delta\hat{\mathbf{r}}|$ and overall, the maximum deviation due to a variation in $\hat{\mathbf{r}}$ is given by

$$2|\hat{\mathbf{v}}| \, |\delta\hat{\mathbf{r}}| .$$

Finally, we can piece together these two variations. By ignoring higher order terms, it is clear that a Taylor series expansion of $R(\hat{\mathbf{r}}, \theta)$ yields the following bound on errors in the computed value of a rotation:

$$|R(\hat{\mathbf{r}} + \delta\hat{\mathbf{r}}, \theta + \Delta\theta)\mathbf{v} - R(\hat{\mathbf{r}}, \theta)\mathbf{v}| \leq (2|\delta\hat{\mathbf{r}}| + |\Delta\theta|)|\mathbf{v}| .$$

Now, if the errors ϕ and ψ are small, then we know that

$$|\Delta\theta| \leq |\phi + \psi| .$$

Furthermore,

$$|\delta\hat{\mathbf{r}}| = |\sin\psi| \approx |\psi|$$

and this implies a bound on variation in $\mathbf{v}$ of

$$|3\psi + \phi|\,|\mathbf{v}| .$$

Moreover, if we are careful to restrict our computation appropriately, then $\psi \approx \phi$, and thus

$$|R(\hat{\mathbf{r}} + \delta\hat{\mathbf{r}}, \theta + \Delta\theta)\,\mathbf{v} - R(\hat{\mathbf{r}}, \theta)| \leq |4\phi|\,|\mathbf{v}| .$$

Errors in $\mathbf{v}_0$

We now consider bounds on the error associated with computing the translation component $\mathbf{v}_0$ of the transformation. Recall that the correct form for $\mathbf{v}_0$ is given by

$$\begin{aligned}[\hat{\mathbf{n}}_{m,i}\hat{\mathbf{n}}_{m,j}\hat{\mathbf{n}}_{m,k}]\,\mathbf{v}_0 = &\left(\hat{\mathbf{n}}'_{m,i}\cdot\mathbf{p}_{s,i} - d_i\right)\left(\hat{\mathbf{n}}'_{m,j}\times\hat{\mathbf{n}}'_{m,k}\right)\\ &+ \left(\hat{\mathbf{n}}'_{m,j}\cdot\mathbf{p}_{s,j} - d_j\right)\left(\hat{\mathbf{n}}'_{m,k}\times\hat{\mathbf{n}}'_{m,i}\right)\\ &+ \left(\hat{\mathbf{n}}'_{m,k}\cdot\mathbf{p}_{s,k} - d_k\right)\left(\hat{\mathbf{n}}'_{m,i}\times\hat{\mathbf{n}}'_{m,j}\right)\end{aligned}$$

where $\hat{\mathbf{n}}_{m,i}$ is a face normal in model coordinates, $\hat{\mathbf{n}}'_{m,i}$ is the transformed normal in sensor coordinates, $\mathbf{p}_{s,i}$ is the position vector of the contact point in sensor coordinates, and d_i is the constant offset for face i. We will consider error ranges for each of the components

$$\left(\hat{\mathbf{n}}'_{m,k}\cdot\mathbf{p}_{s,k} - d_k\right)\left(\hat{\mathbf{n}}'_{m,i}\times\hat{\mathbf{n}}'_{m,j}\right)$$

separately.

We let $s = \hat{\mathbf{n}}_{m,k}\cdot\mathbf{p}_{s,k} - d_k$ and $\mathbf{v} = \hat{\mathbf{n}}'_{m,i}\times\hat{\mathbf{n}}'_{m,j}$ so that the correct component is simply $s\,\mathbf{v}$ and the computed component is

$$(s + \Delta)\,(\xi\mathbf{v} + \eta\hat{\mathbf{u}})$$

where $\hat{\mathbf{u}}$ is a unit vector orthogonal to $\mathbf{v}$, and Δ, ξ and η are values to be determined. We assume that the measured position vector is given by $\mathbf{p}_i + \delta\mathbf{p}_i$, where $\delta\mathbf{p}_i$ is a vector of magnitude ϵ_d, and the measured normal is given by $\hat{\mathbf{n}}_{s,i}$ such that $\hat{\mathbf{n}}_{s,i}\cdot\hat{\mathbf{n}}'_{m,i} \geq \epsilon_n$.

First note that the magnitude of the error in computing the component of the translation is given by

$$|s\mathbf{v} - (s + \Delta)\,(\xi\mathbf{v} + \eta\hat{\mathbf{u}})| = \sqrt{[s\,(1 - \xi) - \Delta\xi]^2\,(\mathbf{v}\cdot\mathbf{v}) + \eta^2\,(s + \Delta)^2}. \tag{12}$$

Thus, we need to find bounds for $s, \Delta, (\mathbf{v}\cdot\mathbf{v}), \xi$ and η^2. We know that s is a given scalar value. If the angle between the face normals is given by $\hat{\mathbf{n}}_i\cdot\hat{\mathbf{n}}_j = \cos\varsigma$, then

$$\mathbf{v}\cdot\mathbf{v} = 1 - \cos^2\varsigma = \sin^2\varsigma.$$

It is straightforward to show that

$$\begin{aligned}\Delta &= \left|\left(\hat{\mathbf{n}}_s\cdot(\mathbf{p} + \delta\mathbf{p}) - d\right) - \left(\hat{\mathbf{n}}'_m\cdot\mathbf{p} - d\right)\right|\\ &\leq \epsilon_d + \left|\left(\hat{\mathbf{n}}_s - \hat{\mathbf{n}}'_m\right)\cdot\mathbf{p}\right|\\ &\leq \epsilon_d + |\mathbf{p}|\,\sqrt{2}\sqrt{1 - \epsilon_n}.\end{aligned}$$

Next, we consider bounds for ξ, η^2, where

$$\hat{\mathbf{n}}_{s,i}\times\hat{\mathbf{n}}_{s,j} = \xi\left(\hat{\mathbf{n}}'_{m,i}\times\hat{\mathbf{n}}'_{m,j}\right) + \eta\hat{\mathbf{h}} \tag{13}$$

for some unit vector $\hat{\mathbf{h}}$ orthogonal to $\left(\hat{\mathbf{n}}'_{m,i} \times \hat{\mathbf{n}}'_{m,j}\right)$. Now

$$\begin{aligned}\left(\hat{\mathbf{n}}_{s,i} \times \hat{\mathbf{n}}_{s,j}\right) \cdot \left(\hat{\mathbf{n}}'_{m,i} \times \hat{\mathbf{n}}'_{m,j}\right) &= \left(\hat{\mathbf{n}}_{s,i} \cdot \hat{\mathbf{n}}'_{m,i}\right)\left(\hat{\mathbf{n}}_{m,j} \cdot \hat{\mathbf{n}}'_{m,j}\right) - \left(\hat{\mathbf{n}}_{s,i} \cdot \hat{\mathbf{n}}'_{m,j}\right)\left(\hat{\mathbf{n}}_{s,j} \cdot \hat{\mathbf{n}}'_{m,i}\right) \\ &\geq \epsilon_n^2 - \left(\hat{\mathbf{n}}_{s,i} \cdot \hat{\mathbf{n}}'_{m,j}\right)\left(\hat{\mathbf{n}}_{s,j} \cdot \hat{\mathbf{n}}'_{m,i}\right).\end{aligned}$$

Moreover, the worst case (i.e. largest value) for $\hat{\mathbf{n}}_{s,i} \cdot \hat{\mathbf{n}}'_{m,j}$ occurs at $\cos\left(\varsigma - \cos^{-1}\epsilon_n\right)$ since this is the smallest angle possible between the cone of radius ϵ_n about $\hat{\mathbf{n}}'_{m,i}$ and the vector $\hat{\mathbf{n}}'_{m,j}$. (Note that we have assumed that the two cones do not overlap, i.e. $\varsigma > 2\cos^{-1}\epsilon_n$). As before, we let $\epsilon_n = \cos\phi$. Then, by substitution and expansion, we get

$$\left(\hat{\mathbf{n}}_{s,i} \times \hat{\mathbf{n}}_{s,j}\right) \cdot \left(\hat{\mathbf{n}}'_{m,i} \times \hat{\mathbf{n}}'_{m,j}\right) \geq \sin\varsigma \sin\left(\varsigma - 2\phi\right).$$

At the same time, from equation (13)

$$\left(\hat{\mathbf{n}}_{s,i} \times \hat{\mathbf{n}}_{s,j}\right) \cdot \left(\hat{\mathbf{n}}'_{m,i} \times \hat{\mathbf{n}}'_{m,j}\right) = \xi \sin^2\varsigma$$

so that we have the bound

$$\xi \geq \frac{\sin\left(\varsigma - 2\phi\right)}{\sin\varsigma}.$$

Now the length of $\left(\hat{\mathbf{n}}_{s,i} \times \hat{\mathbf{n}}_{s,j}\right)$ is given by

$$1 - \left(\hat{\mathbf{n}}_{s,i} \cdot \hat{\mathbf{n}}_{s,j}\right)^2$$

and to get a bound on η^2, we want to maximize this expression. As before, the worst case occurs when the $\hat{\mathbf{n}}_s$ vectors lie at the limits of their respective cones, and

$$\left(\hat{\mathbf{n}}_{s,i} \cdot \hat{\mathbf{n}}_{s,j}\right) = \cos\left(\varsigma + 2\phi\right).$$

We also have, however, from equation (13),

$$\begin{aligned}\left(\hat{\mathbf{n}}_{s,i} \times \hat{\mathbf{n}}_{s,j}\right) \cdot \left(\hat{\mathbf{n}}_{s,i} \times \hat{\mathbf{n}}_{s,j}\right) &= \xi^2 \sin^2\varsigma + \eta^2 \\ &\leq 1 - \cos^2\left(\varsigma + 2\phi\right).\end{aligned}$$

Substitution and expansion yield the following bound

$$\eta^2 \leq \sin\left(4\phi\right)\sin\left(2\varsigma\right).$$

We are now ready to bound the error in computing each component of the translation vector $\mathbf{v}_0$. From equation (12), the magnitude of the error is given by

$$\sqrt{\left[s\left(1-\xi\right) - \Delta\xi\right]^2\left(\mathbf{v}\cdot\mathbf{v}\right) + \eta^2\left(s+\Delta\right)^2}.$$

Substitution of the various bounds yields

$$\sqrt{\left[s\sin\varsigma - \left(s+\Delta\right)\sin\left(\varsigma - 2\phi\right)\right]^2 + \left(s+\Delta\right)^2\sin\left(2\varsigma\right)\sin\left(4\phi\right)}$$

where

$$\begin{aligned}s &= \hat{\mathbf{n}}_{s,k} \cdot \mathbf{p}_{s,k} - d_k \\ \Delta &\leq \epsilon_d + \left|\mathbf{p}_{s,k}\right|\sqrt{2}\sqrt{1-\epsilon_n} \\ \cos\varsigma &= \hat{\mathbf{n}}_{m,i} \cdot \hat{\mathbf{n}}_{m,j}.\end{aligned}$$

Note that as $\phi \mapsto 0$, this bound reduces to $\left|\Delta\sin\varsigma\right|$. Furthermore, as $\epsilon_d \mapsto 0$, this expressions tends to 0, so that the error in the computed translation vanishes as the error in the measurements do.

Typically, we will want to restrict our computations to cases in which the faces are roughly orthogonal, so that $\varsigma \approx \frac{\pi}{2}$. In this case, the bound reduces to the simple expression

$$\left|s - \left(s+\Delta\right)\cos\left(2\phi\right)\right|.$$