MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 1362 July 10, 1992

Recognizing 3D Objects from 2D Images: An Error Analysis

W. Eric L. Grimson, Daniel P. Huttenlocher[1] & T.D. Alter

Abstract. Many recent object recognition systems use a small number of pairings of data and model features to compute the 3D transformation from a model coordinate frame into the sensor coordinate system. In the case of perfect image data, these systems seem to work well. With uncertain image data, however, the performance of such methods is less well understood. In this paper, we examine the effects of two-dimensional sensor uncertainty on the computation of three-dimensional model transformations. We use this analysis to bound the uncertainty in the transformation parameters, as well as the uncertainty associated with applying the transformation to map other model features into the image. We also examine the effects of the transformation uncertainty on the effectiveness of recognition methods.

Acknowledgments: This report describes research done in part at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by an Office of Naval Research University Research Initiative grant under contract N00014-86-K-0685, and in part by the Advanced Research Projects Agency of the Department of Defense under Army contract number DACA76-85-C-0010 and under Office of Naval Research contract N00014-91-J-4038. WELG is supported in part by NSF contract number IRI-8900267. DPH is supported at Cornell University in part by NSF grant IRI-9057928 and matching funds from General Electric, Kodak and Xerox, and in part by the Air Force Office of Scientific Research under contract AFOSR-91-0328. TDA is supported in part by a DoD Graduate Fellowship.

[1] Dept. of Comp. Sci., Cornell University, Ithaca, NY

Recognizing 3D Objects from 2D Images: An Error Analysis

W. Eric L. Grimson
MIT AI Lab & Dept. of EE & CS, Cambridge Mass.
welg@ai.mit.edu 617-253-5346 fax 617-253-5060

Daniel P. Huttenlocher
Dept. of Comp. Sci., Cornell University, Ithaca, NY

T.D. Alter
MIT AI Lab & Dept. of EE & CS, Cambridge Mass.

Abstract. Many recent object recognition systems use a small number of pairings of data and model features to compute the 3D transformation from a model coordinate frame into the sensor coordinate system. In the case of perfect image data, these systems seem to work well. With uncertain image data, however, the performance of such methods is less well understood. In this paper, we examine the effects of two-dimensional sensor uncertainty on the computation of three-dimensional model transformations. We use this analysis to bound the uncertainty in the transformation parameters, as well as the uncertainty associated with applying the transformation to map other model features into the image. We also examine the effects of the transformation uncertainty on the effectiveness of recognition methods.

Acknowledgments: This report describes research done in part at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by an Office of Naval Research University Research Initiative grant under contract N00014-86-K-0685, and in part by the Advanced Research Projects Agency of the Department of Defense under Army contract number DACA76-85-C-0010 and under Office of Naval Research contract N00014-85-K-0124. WELG is supported in part by NSF contract number IRI-8900267. DPH is supported at Cornell University in part by NSF grant IRI-9057928 and matching funds from General Electric, Kodak and Xerox, and in part by the Air Force Office of Scientific Research under contract AFOSR-91-0328. TDA is supported in part by a DoD Graduate Fellowship.

1 Introduction

Object recognition is one of the most ubiquitous problems in computer vision, arising in a wide range of applications and tasks. As a consequence, considerable effort has been expended in tackling variations of the problem, with a particular emphasis recently on model-based methods. In the standard model-based approach, a set of stored geometric models are compared against geometric features that have been extracted from an image of a scene (cf. [4, 11]). Comparing a model with a set of image features generally involves finding a valid correspondence between a subset of the model features and a subset of the image features. For a correspondence to be valid, it is usually required that there exist some transformation of a given type mapping each model feature (roughly) onto its corresponding image feature. This transformation generally specifies the *pose* of the object – its position and orientation with respect to the image coordinate system. The goal thus is to deduce the existence of a legal transformation from model to image and to measure its 'quality'. In other words, the goal is to determine whether there is an instance of the transformed object model in the scene, and the extent of the model present in the data.

More formally, let $\{F_i | 1 \leq i \leq m\}$ be a set of model features measured in a coordinate frame $\mathcal{M}$, let $\{f_i | 1 \leq i \leq s\}$ be a set of sensory features measured in a coordinate frame $\mathcal{S}$, and let $\mathcal{T} : \mathcal{M} \rightarrow \mathcal{S}$ denote a legal transformation from the model coordinate frame to the sensor coordinate frame. The goal is to identify a correspondence, $I \subseteq 2^{m \times s}$, that pairs model features with sensor features. Each such correspondence I specifies some transformation $\mathcal{T}_I$ which maps each model feature close to its corresponding image feature.[2] That is

$$I = \{(m_i, s_j) | \rho(\mathcal{T}_I m_i, s_j) \leq \epsilon\},$$

where ρ is some appropriate measure (e.g. Euclidean distance in the case of point features, or maximum Euclidean separation in the case of line features) and ϵ is some bound on uncertainty. In general the quality of such an interpretation is measured in terms of the number of pairs of model and image features, or the cardinality of I, $|I|$. The goal of recognition is generally either to find the best interpretation, maximizing $|I|$, or all interpretations where $|I| > t$ for some threshold t.

Many of the approaches to the recognition problem can be distinguished by the manner in which they search for solutions. One class of methods focuses on finding the correspondence I, typically by searching a potentially exponential sized space of pairings of model and data

[2] A given interpretation I will in fact generally define a range of 'equivalent' transformations in the sense that there are a number of transformations that generate the same set I.

features (e.g. [5, 7, 10, 11]). A second class of methods focuses on finding the pose $\mathcal{T}$, typically by searching a potentially infinite resolution space of possible transformations, (e.g. [2, 8, 9, 18, 19, 20, 21, 22]). A third class of methods is a hybrid of the other two, in that correspondences of a small number of features are used to explicitly transform a model into image coordinates (e.g., [1, 3, 6, 16, 17]) guiding further search for correspondences.

We are primarily interested in methods in the second and third classes, because they compute transformations from model coordinate frame to image coordinate frame using a small number of feature pairs. When the sensor data can be measured exactly, the fact that a small number of features are used to compute a pose does not cause problems. For real vision systems, however, there is generally uncertainty in measuring the locations of data features, and resulting uncertainty in the estimated pose. In this paper we develop methods for bounding the degree of uncertainty in a three-dimensional transformation computed from a small number of pairs of model and image points. The specific pose estimation method that we investigate is that of [16, 17], however the results are very similar for a number of other methods (e.g., [23, 24]). This pose estimation method uses the correspondence of 3 model and image features to compute the three-dimensional position and orientation of a model with respect to an image, under a 'weak perspective' imaging model (orthographic projection plus scaling).

The central idea of most pose-based recognition methods (such as alignment, geometric hashing, generalized Hough transform) is to use a small number of corresponding model and image features to estimate the pose of an object acting under some kind of transformation. The methods then differ in terms of how the computed pose is used to identify possible interpretations of the model in the image. The pose clustering and pose hashing methods compute all (or many) possible transformations, and then search the transformation space for clusters of similar transformations. In contrast, the alignment approaches explicitly transform the model into the image coordinate frame. The effects of pose errors in these two cases will be different, and we analyze the two cases separately.

Implementation and testing of pose-based recognition methods has been reported in the literature, with good results. An open issue, however, is the sensitivity of such methods to noise in the sensory data. This includes both the range of uncertainty associated with a computed transformation, and the effect of this uncertainty on the range of possible positions for other aligned model features. The answers to these questions can be used to address other issues, such as analyzing the probability of false positive responses, as well as using this analysis to build accurate verification algorithms. In addressing these issues, we first derive expressions for the degree of uncertainty in computing the pose, given bounds on the degree of sensing uncertainty.

Then we apply these results to analyze pose clustering methods, and alignment methods. The big difference between these methods is that the former methods operate explicitly in the space of possible poses (or transformations), whereas the latter methods operate in the space of image measurements.

Previous Results

Some earlier work has been done on simpler versions of these questions, such as recognition that is restricted to 2D objects that can rotate and translate in the image plane [12] or recognition that is restricted to 2D objects that can be arbitrarily rotated and translated in 3D, then projected into the image plane [14]. While these are useful for analyzing particular cases of recognition, we are interested in extending these results to the full case of a non-planar 3D object undergoing arbitrary rigid motion (under a 'weak perspective' imaging model of orthographic projection plus scaling).

2 Computing 3D Pose from 2D data

The pose estimation technique that we evaluate in detail is that of [16, 17], though similar results hold for other techniques that use a small number of points to estimate three-dimensional pose. This method of computing the pose operates as follows: We are given three image points and three model points, each measured in their own coordinate system; the result is the translation, scale and three-dimensional rotation that position the three model points in space such that they map onto the image points under orthographic projection. The original specification of this method assumes exact measurement of the image points is possible. In contrast, our development here assumes that each image measurement is only known to within some uncertainty disc of a given radius, ϵ. We speak of the nominal measured image points, which are the centers of these discs. The measured points can be used to compute an exact transformation, and then we are concerned with the variations in this transformation as the locations of the image points vary within their respective discs.

Let one of the measured image points be designated the origin of the points, represented by the vector $\vec{o}$, measured in the image coordinate system. Let the relative vectors from this point to the other two points be $\vec{m}$ and $\vec{n}$, also measured in image coordinates. Similarly, let $\vec{O}, \vec{M}$ and $\vec{N}$ denote the three model vectors corresponding to $\vec{o}, \vec{m}$ and $\vec{n}$, measured in a coordinate system centered on the model. For convenience, we assume that the model coordinate origin is in fact at $\vec{O}$ (see Figure 1a). We also assume that the model can be reoriented so that $\vec{M}$ and $\vec{N}$ lie in a plane parallel to the image plane. Note that we use the notation $\vec{x}$ for general

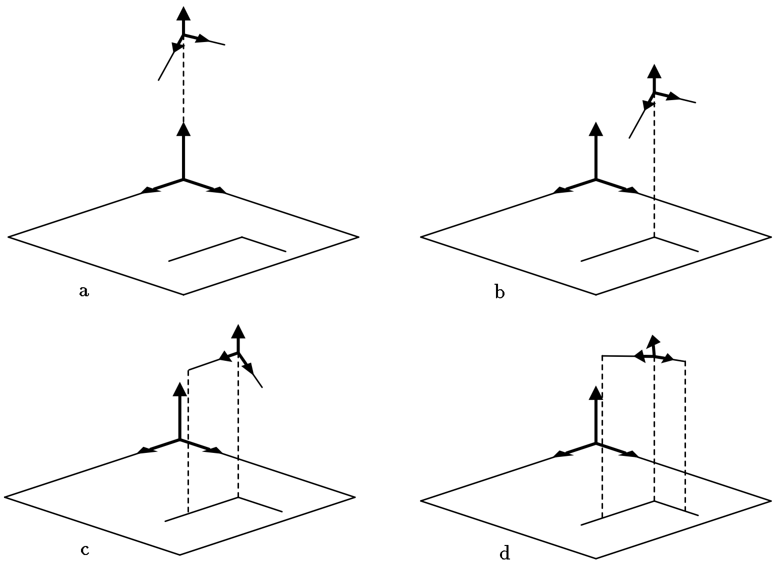


Figure 1: Computing the pose. Part a: A pair of basis vectors in the image have been selected, as well as a pair of model basis vectors. The origin of the model basis is assumed to project to the origin of the image coordinate system. Part b: After the translation, the origin point of the selected model basis projects to the origin of the selected image basis. Part c: After the first rotation, the first model axis projects along the first image axis. Part d: After the next two rotations, both model axes project along their corresponding image axis, such that the ratios of the lengths of the axes are the same in the projected model and the image.

vectors, and $\widehat{x}$ for unit vectors. Also note that we assume that the optic axis is along $\widehat{z}$.

Our version of the pose estimation algorithm is summarized below. The method described in [16] is similar, but for 2D objects. The method in [17], for 3D objects, is more direct and appears to be numerically more stable. The method used here, however, more readily lends itself to error analysis of the type desired (and the two methods are equivalent except for numerical stability issues). In particular, the method used here allows us to isolate each of the six transformation parameters into individual steps of the computation.

For the exact transformation specified by the measured sensory data, the steps are:

1. Translate the model so that the origins align. A point $\vec{P}$ is then transformed to $\vec{P}'$ by:

$$\vec{P}' = \vec{P} + \vec{o} - \Pi_{\widehat{z}}\vec{O}$$

where $\Pi_{\widehat{z}}$ denotes projection along the $\widehat{z}$ axis (see Figure 1b).

2. Rotate the model by an angle ψ about the axis parallel to $\widehat{z}$ and emanating from $\vec{O}'$ so that $\Pi_{\widehat{z}}\vec{M}$ lies along $\widehat{m}$, leading to the transformation

$$\vec{P}'' = R_{\widehat{z},\psi}\vec{P}'$$

where $R_{\widehat{z},\psi}$ denotes a rotation of angle ψ about the unit axis $\widehat{z}$ (see Figure 1c).

3. Rotate the model by an angle θ about the new $\widehat{M}''$, leading to the transformation
$$\vec{P}''' = R_{\widehat{M}'',\theta}\vec{P}''.$$

4. Rotate the model by an angle ϕ about $\widehat{m}^{\perp} \overset{\text{def}}{=} \widehat{z} \times \widehat{m}$ (Figure 1d), leading to
$$\vec{P}'''' = R_{\widehat{m}^{\perp},\phi}\vec{P}'''.$$

5. Scale by s so that
$$s\Pi_{\widehat{z}}\vec{M}'''' = \vec{m}.$$

The constraints on the process are that $\vec{N}''''$ should project along $\widehat{n}$ with scaled length $sN = n$, and $\vec{M}''''$ should project along $\widehat{m}$ with scaled length $sM = m$.

Now suppose that we don't know the image points exactly, but only to within a disc of radius ϵ. We want to know the effect of this on the computed transformation, i.e. what is the range of uncertainty in each of the transformation parameters if each of the image points is allowed to vary over an $\epsilon-$disc? We divide this analysis as follows. First we consider the transformation that aligns the model origin and the two model vectors with the image origin and the two image vectors. The translation is explicit in this computation, and we note that its error is simply bounded by the image uncertainty, ϵ. We then derive expressions for the remaining transformation parameters, ψ, θ, ϕ and s, which are only implicit in the alignment of the model vectors with the image vectors. Given these expressions we are then able to characterize the effects of sensor uncertainty on these parameters.

3 Aligning the Basis Vectors

First we note that the translation which brings the model origin into correspondence with the image origin, as specified in Step 1 of the method, simply has an uncertainty of ϵ, the sensor uncertainty.

We have some freedom in choosing the model coordinate system, and in particular, we choose the coordinate frame such that both $\vec{M}$ and $\vec{N}$ are orthogonal to the optic axis $\widehat{z}$. In this case, given the measured image data, the angle ψ is:

$$\left\langle \widehat{\Pi_{\widehat{z}} M}, \widehat{m} \right\rangle = \left\langle \widehat{M}, \widehat{m} \right\rangle = \cos\psi \tag{1}$$
$$\left\langle \widehat{\Pi_{\widehat{z}} M} \times \widehat{z}, \widehat{m} \right\rangle = -\left\langle \widehat{M}, \widehat{m}^{\perp} \right\rangle = \sin\psi. \tag{2}$$

Because there will be some uncertainty in computing ψ, due to the uncertainty ϵ in the image points, the first rotation, using Rodrigues' formula, transforms a vector into

$$\vec{P}'' = R_{\widehat{z},\psi+\delta\psi}\vec{P}' = \cos(\psi+\delta\psi)\vec{P}' + (1-\cos(\psi+\delta\psi))\left\langle \vec{P}', \widehat{z}\right\rangle \widehat{z} + \sin(\psi+\delta\psi)\left(\widehat{z}\times\vec{P}'\right). \quad (3)$$

By this we mean that ψ denotes the nominal correct rotation (i.e. the rotation that correctly aligns the model with the *measured* image data, without the uncertainty bounds) and $\delta\psi$ denotes the deviation in angle that could result from the ϵ-bounded uncertainty in measuring the position of the image point.

If we use the small angle approximation for $\delta\psi$, by assuming $|\delta\psi| \ll 1$, then we have

$$\vec{P}'' \approx R_{\widehat{z},\psi}\vec{P}' + \delta\psi R_{\widehat{z},\psi+\frac{\pi}{2}}\Pi_{\widehat{z}}\vec{P}'. \quad (4)$$

For the special case of $\vec{P}' = \widehat{M}$, we have

$$\begin{aligned} \widehat{M}'' &\approx R_{\widehat{z},\psi}\widehat{M} + \delta\psi R_{\widehat{z},\psi+\frac{\pi}{2}}\widehat{M} \\ &\approx \widehat{m} + \delta\psi\widehat{m}^{\perp} \end{aligned} \quad (5)$$

Note that the right hand side is not a unit vector, but to a first order approximation, the expression is sufficient, since we have assumed that $|\delta\psi| \ll 1$. Also note that this assumption is reasonable, so long as we place lower limits on the length of an acceptable image basis vector, i.e. we ensure that the length of the vector separating two basis points in the image is much greater than ϵ.

The second rotation has two components of uncertainty:

$$\vec{P}''' = R_{\widehat{m}+\delta\psi\widehat{m}^{\perp},\theta+\delta\theta}\vec{P}''. \quad (6)$$

We could expand this out using Rodrigues' formula, and keep only first order terms in $\delta\psi$ and $\delta\theta$, under a small angle assumption. Unfortunately, we have found experimentally that while we can safely assume that $\delta\psi$ is small, we cannot make the same assumptions about $\delta\phi$ or $\delta\theta$. Intuitively this makes sense, because the remaining two rotations ϕ and θ are the slant and tilt of the model with respect to the image, and small changes in the image may cause large changes in these rotations. Thus, we keep the full trigonometric expressions for $\delta\psi$ and $\delta\theta$:

$$\begin{aligned} \vec{P}''' \approx\ & R_{\widehat{m},\theta+\delta\theta}\vec{P}'' + \delta\psi(1-\cos(\theta+\delta\theta))\left[\left\langle \vec{P}'', \widehat{m}^{\perp}\right\rangle \widehat{m} + \left\langle \vec{P}'', \widehat{m}\right\rangle \widehat{m}^{\perp}\right] \\ & +\delta\psi\sin(\theta+\delta\theta)\left(\widehat{m}^{\perp}\times\vec{P}''\right) \end{aligned} \quad (7)$$

By a similar reasoning, the third rotation gives:

$$\vec{P}'''' = R_{\widehat{m}^{\perp},\phi+\delta\phi}\vec{P}'''. \quad (8)$$

Now suppose we decompose a point $\vec{P}'$ into the natural coordinate system:

$$\vec{P}' = \alpha\widehat{m} + \beta\widehat{m}^{\perp} + \gamma\widehat{z}.$$

Then this point is transformed to

$$\vec{P}'''' = \alpha\widehat{m}'''' + \beta\widehat{m}^{\perp''''} + \gamma\widehat{z}''''.$$

Thus, to see the representation for the transformed point, we simply need to trace through the transformations associated with each of the basis vectors.

Using equations (4), (7) and (8), we find that rotating the basis vectors, then projecting the result into the image plane (i.e. along $\widehat{z}$) yields (keeping only first order terms in $\delta\psi$)

$$\begin{aligned}
\Pi_{\widehat{z}}\widehat{m}'''' &= [\cos(\phi+\delta\phi)\left\{\cos\psi - \delta\psi\sin\psi\cos(\theta+\delta\theta)\right\} + \sin\psi\sin(\theta+\delta\theta)\sin(\phi+\delta\phi)]\,\widehat{m} \\
&\quad + [\delta\psi\cos\psi + \sin\psi\cos(\theta+\delta\theta)]\,\widehat{m}^{\perp}. && (9) \\
\Pi_{\widehat{z}}\widehat{m}^{\perp''''} &= [\cos(\phi+\delta\phi)\left\{-\sin\psi - \delta\psi\cos\psi\cos(\theta+\delta\theta)\right\} + \cos\psi\sin(\theta+\delta\theta)\sin(\phi+\delta\phi)]\,\widehat{m} \\
&\quad + [-\delta\psi\sin\psi + \cos\psi\cos(\theta+\delta\theta)]\,\widehat{m}^{\perp}. && (10) \\
\Pi_{\widehat{z}}\widehat{z}'''' &= [\delta\psi\cos(\phi+\delta\phi)\sin(\theta+\delta\theta) + \cos(\theta+\delta\theta)\sin(\phi+\delta\phi)]\,\widehat{m} - \sin(\theta+\delta\theta)\widehat{m}^{\perp}. && (11)
\end{aligned}$$

Thus far we have seen how to align the basis vectors of the model with the basis vectors measured in the image, by a combination of two-dimensional translation and three-dimensional rotation. Before we can analyze the effects of uncertainty on the rotation and scale parameters, we need to derive expressions for them.

4 Computing the Implicit Parameters

Now we consider how to compute the remaining parameters of the transformation, and characterize the effects of uncertainty on these parameters. There are some special cases of the transformation that are of particular importance to us. First, consider the case of $\vec{P}' = \vec{M}$. We have chosen the model coordinate system so that in this case

$$\alpha = M\cos\psi \qquad \beta = -M\sin\psi \qquad \gamma = 0$$

and thus

$$s\Pi_{\widehat{z}}\vec{M}'''' = sM\cos(\phi+\delta\phi)\widehat{m} + sM\delta\psi\widehat{m}^{\perp}, \tag{12}$$

where M is the length of the vector $\vec{M}$.

If we first consider the transformation that aligns the model with the measured image points (not accounting for the uncertainty in the image measurements), we want the scaled result to align properly, i.e.

$$s\Pi_{\hat{z}}\vec{M}''''|_{\delta\phi=0} = m\hat{m}.$$

This simply requires that

$$s = \frac{m}{M}\sec\phi. \tag{13}$$

Note that we only want positive scale factors, so we can assume that $\sec\phi \geq 0$, or that

$$-\frac{\pi}{2} \leq \phi \leq \frac{\pi}{2}.$$

The error vector associated with the transformed $\vec{M}$ is then

$$\vec{E}_M = m\sec\phi\delta\psi\hat{m}^{\perp} + m\left[\cos\delta\phi - \tan\phi\sin\delta\phi - 1\right]\hat{m}. \tag{14}$$

We can use this result to put constraints on the range of feasible transformations, and then on the range of feasible positions for other aligned points.

Since there is an ϵ-disc of error associated with the two endpoints of $\vec{m}$, there will be a range of acceptable projections. One way to constrain this range is to note that the magnitude of the error vector should be no more than 2ϵ. (This is because the translation to align origins has an uncertainty disc of ϵ and the actual position of the endpoint also has an uncertainty disc of ϵ.) This then implies that

$$\left|\vec{E}_M\right|^2 \leq 4\epsilon^2$$

or, with substitutions, that

$$\left[\cos\delta\phi - \tan\phi\sin\delta\phi - 1\right]^2 + (1+\tan^2\phi)(\delta\psi)^2 \leq 4\left(\frac{\epsilon}{m}\right)^2. \tag{15}$$

We could also derive slightly weaker constraints, by requiring that the components of the error vectors $\vec{e}$ in each of the directions $\hat{m}$ and $\hat{m}^{\perp}$ be of magnitude at most 2ϵ. In this case, we are effectively using a bounding box rather than a bounding disc of uncertainty. This leads to the following constraints:

$$\begin{aligned} |\delta\psi| &\leq \frac{2\epsilon}{m}|\cos\phi| && (16)\\ |\cos\delta\phi - 1 - \tan\phi\sin\delta\phi| &\leq \frac{2\epsilon}{m}. && (17) \end{aligned}$$

We need to do the same thing with $\vec{N}$ and $\vec{n}$. As noted earlier, we can choose the model coordinate system so that $\vec{N}$ has no $\hat{z}$ component. This means we can represent $\vec{N}$ by

$$\vec{N}' = N\cos\xi\widehat{M} + N\sin\xi\widehat{M}^{\perp}$$

where ξ is a known angle. Similarly, we can represent the corresponding image vector by

$$\vec{n} = n\cos\omega\widehat{m} + n\sin\omega\widehat{m}^{\perp}$$

where ω is a measurable angle. This means that the nominal (i.e. no error) projected transformation of $\vec{N}$ is:

$$\frac{m}{\cos\phi}\frac{N}{M}\left[\cos\xi\cos\phi + \sin\xi\sin\phi\sin\theta\right]\widehat{m} + \frac{m}{\cos\phi}\frac{N}{M}\left[\sin\xi\cos\theta\right]\widehat{m}^{\perp}. \tag{18}$$

But in principle this is also equal to

$$\vec{n} = n\cos\omega\widehat{m} + n\sin\omega\widehat{m}^{\perp}$$

and by equating terms we have

$$\begin{aligned} \frac{m}{n}\frac{N}{M}\left[\cos\xi\cos\phi + \sin\xi\sin\phi\sin\theta\right] &= \cos\phi\cos\omega & (19)\\ \frac{m}{n}\frac{N}{M}\left[\sin\xi\cos\theta\right] &= \cos\phi\sin\omega. & (20)\end{aligned}$$

These two equations define the set of solutions for the transformation parameters ϕ and θ. (Note that the set of solutions for ψ is given by equations (1) and (2).) There are several ways of solving these implicit equations in the variables of interest, namely ϕ and θ. One way is given below. First, let

$$\eta = \frac{m}{n}\frac{N}{M}.$$

We can rewrite these equations as explicit solutions for θ:

$$\begin{aligned} \cos\theta &= \frac{\sin\omega\cos\phi}{\eta\sin\xi}\\ \sin\theta &= \frac{\cos\phi(\cos\omega - \eta\cos\xi)}{\eta\sin\xi\sin\phi}. \end{aligned} \tag{21}$$

This gives us a solution for θ as a function of ϕ. To isolate ϕ, we use the fact that $\sin^2\theta + \cos^2\theta = 1$, and this leads to the following equation:

$$\sin^2\omega\cos^4\phi - \left[\eta^2 + 1 - 2\eta\cos\omega\cos\xi\right]\cos^2\phi + \eta^2\sin^2\xi = 0. \tag{22}$$

This is a quadratic in $\cos^2\phi$, as a function of known quantities, and hence the two solutions will yield a total of up to four solutions for $\cos\phi$. But since we want $s \geq 0$, we know that $\cos\phi \geq 0$, and so at most two of these solutions are acceptable:

$$\cos\phi = \sqrt{\frac{1 - 2\eta\cos\omega\cos\xi + \eta^2 \pm \sqrt{(1 - 2\eta\cos\omega\cos\xi + \eta^2)^2 - 4\eta^2\sin^2\omega\sin^2\xi}}{2\sin^2\omega}}. \tag{23}$$

Note that this gives real solutions only if

$$\cos\omega\cos\xi \leq \frac{1+\eta^2}{2\eta}. \tag{24}$$

Also note that we need $\cos\phi \leq 1$ so that if this does not hold in the equation, we can exclude the cases as being impossible. If there are two real solutions for ϕ, they can be used with equation (21) to find solutions for θ. Note that solutions to equation (22) will be stable only when $\sin\omega$ is not small. This makes sense, since the unstable cases correspond to image bases in which the basis vectors are nearly (anti-)parallel.

The complete version of transforming $\vec{N}$ is given by:

$$\begin{aligned} & sN \quad [\cos\xi\cos(\phi+\delta\phi)+\sin\xi\,\{\sin(\theta+\delta\theta)\sin(\phi+\delta\phi)-\delta\psi\cos(\theta+\delta\theta)\cos(\phi+\delta\phi)\}]\,\widehat{m} \\ + \; & sN \quad [\cos\xi\delta\psi+\sin\xi\cos(\theta+\delta\theta)]\,\widehat{m}^{\perp}. \end{aligned} \tag{25}$$

Similar to our analysis of the error associated with the transformed version of $\vec{M}$, we can set the magnitude of the difference between this vector and the nominal vector to be less than 2ϵ, or we can take the weaker constraints of requiring that the components of the error vector in two orthogonal directions each have magnitude at most 2ϵ. One natural choice of directions is $\widehat{n}$ and $\widehat{n}^{\perp}$ but a more convenient, and equally feasible, choice is $\widehat{m}$ and $\widehat{m}^{\perp}$.

In the latter case, bounding the component of the error in the direction of $\widehat{m}^{\perp}$ yields

$$\left|\frac{N}{M}\sec\phi\,[\delta\psi\cos\xi+\sin\xi\cos(\theta+\delta\theta)]-\frac{n}{m}\sin\omega\right| \leq \frac{2\epsilon}{m}. \tag{26}$$

The nominal transformation

To summarize, we compute the nominal transformation, which is when the nominal image points are correct, as follows:

1. Choose the coordinate system of the model so that the origin lies at $\vec{O}$, and so that $\vec{M}$ and $\vec{N}$ both lie in the $z=0$ plane.

2. Translate the model so that the origins align ($\Pi_{\widehat{z}}$ denotes projection along the $\widehat{z}$ axis):

$$\vec{P}' = \vec{P} + \vec{o} - \Pi_{\widehat{z}}\vec{O}.$$

3. Rotate the model by angle ψ about $\widehat{z}$ so that $\Pi_{\widehat{z}}\vec{M}$ lies along $\widehat{m}$. ψ is given by:

$$\begin{aligned} \left\langle \widehat{\Pi_{\widehat{z}}M}, \widehat{m} \right\rangle &= \left\langle \widehat{M}, \widehat{m} \right\rangle = \cos\psi \\ \left\langle \widehat{\Pi_{\widehat{z}}M} \times \widehat{m}, \widehat{z} \right\rangle &= -\left\langle \widehat{M}, \widehat{m}^{\perp} \right\rangle = \sin\psi. \end{aligned}$$

4. Rotate the model by angle θ about the newly resulting $\widehat{M}$, which in this case will be $\widehat{m}$.

5. Rotate the model by angle ϕ about $\widehat{m}^{\perp}$. The angles ϕ and θ are found by solving

$$\cos\phi = \sqrt{\frac{1 - 2\eta\cos\omega\cos\xi + \eta^2 \pm \sqrt{(1+\eta^2 - 2\eta\cos\omega\cos\xi)^2 - 4\eta^2\sin^2\omega\sin^2\xi}}{2\sin^2\omega}}$$

for ϕ, and then solving

$$\begin{aligned} \cos\theta &= \frac{\sin\omega\cos\phi}{\eta\sin\xi} \\ \sin\theta &= \frac{\cos\phi(\cos\omega - \eta\cos\xi)}{\eta\sin\xi\sin\phi} \end{aligned}$$

for θ, where

$$\eta = \frac{m}{n}\frac{N}{M}$$

and where only solutions for which $0 \leq \cos\phi \leq 1$ are kept.

6. Project into the image, and scale by

$$s = \frac{m}{M}\sec\phi.$$

5 Uncertainty in the Implicit Parameters

Now we turn to the problem of bounding the range of uncertainty associated with the rotation and scale parameters of the transformation, given that there are ϵ-bounded positional errors in the measurement of the image points.

To bound the rotational uncertainties, we will start with equation (15):

$$[\cos\delta\phi - \tan\phi\sin\delta\phi - 1]^2 + (1+\tan^2\phi)(\delta\psi)^2 \leq 4\left(\frac{\epsilon}{m}\right)^2.$$

From this, a straightforward bound on the uncertainty in ψ is

$$|\delta\psi| \leq \frac{2\epsilon}{m}|\cos\phi| = \frac{2\epsilon}{m}\cos\phi.$$

To solve for bounds on $\delta\phi$, we use the following analysis. Given a value for $\delta\psi$, we have the implicit equation

$$\cos\delta\phi - 1 - \tan\phi\sin\delta\phi = \mu \tag{27}$$

where μ can range over:

$$-\sqrt{\left(\frac{2\epsilon}{m}\right)^2 - \sec^2\phi\,(\delta\psi)^2} \leq \mu \leq \sqrt{\left(\frac{2\epsilon}{m}\right)^2 - \sec^2\phi\,(\delta\psi)^2}.$$

We can turn this into a quadratic equation in $\cos\delta\phi$ and into a quadratic equation in $\sin\delta\phi$, leading to two solutions for each variable, and hence a total of four possible solutions. By enforcing agreement with the implicit equation, we can reduce this to a pair of solutions:

$$\begin{aligned}\sin\delta\phi &= -(1+\mu)\sin\phi\cos\phi + \sigma\cos\phi\sqrt{1-(1+\mu)^2\cos^2\phi} && (28)\\ \cos\delta\phi &= (1+\mu)\cos^2\phi + \sigma\sin\phi\sqrt{1-(1+\mu)^2\cos^2\phi} && (29)\end{aligned}$$

where $\sigma = \pm 1$. Note that in order for these equations to yield a real result for $\delta\phi$, we need the argument to the square root to be positive. This yields additional constraints on the range of legitimate values for μ.

These two equations implicitly define a range of solutions for $\delta\phi$, as μ varies. Since $\mu \neq -1$ and $\phi \neq \pm\frac{\pi}{2}$ (since this case corresponds to $m = 0$), we can actually simplify this to:

$$\tan\delta\phi = \frac{-\tan\phi + \sigma\sqrt{\frac{1}{(1+\mu)^2\cos^2\phi} - 1}}{1 + \sigma\tan\phi\sqrt{\frac{1}{(1+\mu)^2\cos^2\phi} - 1}}. \tag{30}$$

By substituting for the two values for σ and by substituting for the limits on μ, we can obtain a range of values for $\delta\phi$. In fact, we get two ranges, one for the case of $\sigma = 1$ and one for the case of $\sigma = -1$. Only one of these two ranges, in general, will contain 0, and since this must hold, we can exclude the second range. In fact, when $\mu = 0$, $\tan\delta\phi = 0$, and if we substitute these values into equation (30), we find that

$$\sigma = \operatorname{sgn}(\tan\phi) = \begin{cases} 1 & \text{if } \tan\phi \geq 0; \\ -1 & \text{if } \tan\phi < 0. \end{cases}$$

Note that we can simplify the implicit expressions for $\delta\phi$. If we let

$$\nu = \arccos\left[(1+\mu)\cos\phi\right]$$

then

$$\delta\phi = \sigma\nu - \phi. \tag{31}$$

To solve for bounds on $\delta\theta$, we do a similar analysis, using equation (26). We have the implicit equation

$$\frac{N}{M}\sec\phi\left[\delta\psi\cos\xi + \sin\xi\cos(\theta+\delta\theta)\right] - \frac{n}{m}\sin\omega = \mu \tag{32}$$

where

$$-\frac{2\epsilon}{m} \leq \mu \leq \frac{2\epsilon}{m}.$$

We can write this equation in the form:

$$\cos\theta\cos\delta\theta - \sin\theta\sin\delta\theta = a$$

and using exactly the same kind of analysis, we can solve for $\sin\delta\theta$ and $\cos\delta\theta$:

$$\begin{aligned}\cos\delta\theta &= a\cos\theta + \sigma\sin\theta\sqrt{1-a^2} \qquad (33)\\ \sin\delta\theta &= -a\sin\theta + \sigma\cos\theta\sqrt{1-a^2} \qquad (34)\end{aligned}$$

where

$$\begin{aligned}\sigma &= \pm 1\\ a &= \frac{nM\sin\omega}{mN\sin\xi}\cos\phi - \frac{\delta\psi}{\tan\xi} + \frac{M\cos\phi}{N\sin\xi}\mu\\ -\frac{2\epsilon}{m}|\cos\phi| &\leq \delta\psi \leq \frac{2\epsilon}{m}|\cos\phi|\\ -\frac{2\epsilon}{m} &\leq \mu \leq \frac{2\epsilon}{m},\end{aligned}$$

and where μ is further constrained to keep the solutions real. Again, by substituting for the two values for σ, by substituting for the limits on μ, and by substituting for the limits on $\delta\psi$, we can obtain a range of values for $\delta\theta$.

Similar to the $\delta\phi$ case, we actually get two ranges, one for the case of $\sigma = 1$ and one for the case of $\sigma = -1$. Again, in general only one of these two ranges will span 0, and since this must hold, we can automatically exclude the second range.

Also similar to the $\delta\phi$ case, we can simplify the expression for $\delta\theta$. In particular, if we let

$$\nu = \arccos a$$

then

$$\delta\theta = b\nu - \theta. \qquad (35)$$

To bound the error in scale, we return to equation (12). If we let δs denote the multiplicative uncertainty in computing the scale factor s, i.e. the actual computation of scale is $s\delta s$, then by equation (12), one inequality governing this uncertainty is

$$\left[\frac{m\delta s\cos(\phi+\delta\phi)}{\cos\phi} - m\right]^2 + \left[\frac{m\delta s\delta\psi}{\cos\phi}\right]^2 \leq 4\epsilon^2. \qquad (36)$$

We can expand this out and solve for limits on δs, which are given by

$$\begin{aligned}\delta s &\geq \frac{\cos(\phi+\delta\phi)\cos\phi - \cos\phi\sqrt{\frac{4\epsilon^2}{m^2}\left(\cos^2(\phi+\delta\phi)+\delta^2\psi\right) - \delta^2\psi}}{\cos^2(\phi+\delta\phi)+\delta^2\psi}\\ \delta s &\leq \frac{\cos(\phi+\delta\phi)\cos\phi + \cos\phi\sqrt{\frac{4\epsilon^2}{m^2}\left(\cos^2(\phi+\delta\phi)+\delta^2\psi\right) - \delta^2\psi}}{\cos^2(\phi+\delta\phi)+\delta^2\psi} \qquad (37)\end{aligned}$$

Thus, given bounds on $\delta\psi$, from which we can get a range of values for $\delta\phi$, we can compute the range of values for δs.

In sum, our task was to obtain reasonable bounds on the errors in the six parameters of the transformation, namely translation, $\delta\psi, \delta\theta, \delta\phi$ and δs. The translation is known up to an ϵ-disc. For the rotations and scale, we used constraints that force $\vec{M}$ and $\vec{N}$ to project near the uncertainty regions surrounding $\vec{m}$ and $\vec{n}$, respectively. Specifically, let $\vec{E}_M$ be the error vector from $\vec{m}$ to the transformed and projected $\vec{M}$, and similarly for $\vec{E}_N$ and $\vec{n}$. We first used a constraint on the magnitude of $\vec{E}_M$ in the direction of $\vec{m}^{\perp}$ to get bounds on $\delta\psi$. Then, using the bounds on $\delta\psi$ plus a more general constraint on the magnitude of $\vec{E}_M$, we obtained a range of values for $\delta\phi$. Next, we used the bounds on $\delta\psi$ again plus a constraint on the magnitude of $\vec{E}_N$ in the direction of $\vec{m}^{\perp}$ to get a range of values for $\delta\theta$. Lastly, we bounded δs using the bounds on $\delta\psi, \delta\theta$, and the general constraint on the magnitude of $\vec{E}_M$.

Summary of Bounds on Parameters

To summarize, we have the following bounds on uncertainty in the transformation given ϵ uncertainty in the image measurements. The translation uncertainty is simply bounded by ϵ. The rotational uncertainty is:

$$|\delta\psi| \leq \frac{2\epsilon}{m} |\cos\phi| , \tag{38}$$

and

$$\begin{aligned} \sin\delta\phi &= -(1+\mu)\sin\phi\cos\phi + \operatorname{sgn}(\tan\phi)\cos\phi\sqrt{1-(1+\mu)^2\cos^2\phi} \\ \cos\delta\phi &= (1+\mu)\cos^2\phi + \operatorname{sgn}(\tan\phi)\sin\phi\sqrt{1-(1+\mu)^2\cos^2\phi} \end{aligned} \tag{39}$$

subject to the constraint that:

$$-\sqrt{\left(\frac{2\epsilon}{m}\right)^2 - \sec^2\phi\,(\delta\psi)^2} \leq \mu \leq \sqrt{\left(\frac{2\epsilon}{m}\right)^2 - \sec^2\phi\,(\delta\psi)^2},$$

and

$$\begin{aligned} \cos\delta\theta &= a\cos\theta + \sigma\sin\theta\sqrt{1-a^2} \\ \sin\delta\theta &= -a\sin\theta + \sigma\cos\theta\sqrt{1-a^2} \end{aligned} \tag{40}$$

where

$$\begin{aligned} \sigma &= \pm 1 \\ a &= \frac{nM\sin\omega}{mN\sin\xi}\cos\phi - \frac{\delta\psi}{\tan\xi} + \frac{M\cos\phi}{N\sin\xi}\mu \\ -\frac{2\epsilon}{m}|\cos\phi| &\leq \delta\psi \leq \frac{2\epsilon}{m}|\cos\phi| \\ -\frac{2\epsilon}{m} &\leq \mu \leq \frac{2\epsilon}{m}, \end{aligned}$$

and where μ is further constrained to keep the solutions real.

The uncertainty in scale is constrained by

$$\begin{aligned}
\delta s &\geq \frac{\cos(\phi+\delta\phi)\cos\phi - \cos\phi\sqrt{\frac{4\epsilon^2}{m^2}\left(\cos^2(\phi+\delta\phi)+\delta^2\psi\right)-\delta^2\psi}}{\cos^2(\phi+\delta\phi)+\delta^2\psi} \\
\delta s &\leq \frac{\cos(\phi+\delta\phi)\cos\phi + \cos\phi\sqrt{\frac{4\epsilon^2}{m^2}\left(\cos^2(\phi+\delta\phi)+\delta^2\psi\right)-\delta^2\psi}}{\cos^2(\phi+\delta\phi)+\delta^2\psi} \qquad (41)
\end{aligned}$$

We note that the bounds on the error for $\delta\theta$ and δs are overly conservative, in that we have not used all the constraints available to us in bounding these errors.

Constraints on the Analysis

All of the bounds computed are overestimates, up to a few reasonable approximations and with the possible exception of the scale factor. To bring everything together, we now list and discuss the approximations and overestimates.

In computing the formula for transforming and projecting a model point into the image, we assumed that $|\delta\psi| \ll 1$, so that we could use the small angle approximation, which gave $\cos\delta\psi \approx 1$ and $\sin\delta\psi \approx \delta\psi$, and so that we could drop higher order terms in $\delta\psi$.

Next, we list the sources of our overbounds on the parameters. First, we used a constraint that the error vector ($\vec{E}_M$) for projection of $\vec{M}$ has magnitude at most 2ϵ. This is a weaker constraint than requiring the destination point of the transformed and projected $\vec{M}$ to be within the ϵ-circle surrounding the image point at the destination point of $\vec{m}$.

The weak constraint on $\vec{E}_M$ was used directly to bound both $\delta\phi$ and δs, but an even weaker version was used to bound $\delta\psi$. The weaker version simply requires the magnitude of $\vec{E}_M$ in the direction of $\vec{m}^\perp$ to be at most 2ϵ. Similarly, $\delta\theta$ was bounded with a constraint that forces the magnitude of $\vec{E}_N$ in the direction of $\vec{m}^\perp$ to be at most 2ϵ. One indication that the constraint on $\vec{E}_N$ is weak is that it is independent of $\delta\phi$, the rotation about $\vec{m}^\perp$.

Further, it should be observed that another source of overbounding was the treatment of the constraints on $\vec{E}_M$ and $\vec{E}_N$ as independent. In actuality the constraints are coupled.

Finally, there is one place where we did not clearly overbound the rotation errors, which are $\delta\psi$, $\delta\theta$, and $\delta\phi$. In computing their ranges of values, we used the nominal value of the scale factor, whose differences from the extreme values of the scale factor may not be insignificant.

6 Using the Bounds

The bounds on the uncertainty in the 3D pose of an object, computed from three corresponding model and image points, have a number of applications. They can be used to design careful verification algorithms, they can be used to design error-sensitive voting schemes, and they can be used to analyze the expected performance of recognition methods. In this section, we consider all three such applications.

6.1 3D Hough transforms

We begin by considering the impact of the error analysis on pose clustering methods, such as the generalized Hough transform[2]. These methods seek to find solutions to the recognition problem by the following general technique:

1. Consider a pairing of k-tuples of model and image features, where k is the smallest such tuple that defines a complete transformation from model to image.
2. For each such pair, determine the associated transformation.
3. Use the parameters of the transformation to index into a hash space, and at the indexed point, increment a counter. This implies that the pairing of k-tuples is voting for the transformation associated with that index.
4. Repeat for all possible pairings of k-tuples.
5. Search the hash space for peaks in the stored votes, such peaks serving to hypothesize a pose of the object.

While the generalized Hough transform is usually used for matching 2D images to 2D objects undergoing rigid transformations in the plane, or for matching 3D objects to 3D data, it has also been applied to recognizing 3D objects from 2D images (e.g. [24, 25, 15]). In this case, each dimension of the Hough space corresponds to one of the transformation parameters. Under a weak perspective imaging model these parameters (as we saw above) are two translations, three rotations, and a scale factor. The method summarized at the end of Section 4 provides one technique for determining the values of these parameters associated with a given triple of model and image points.

In the case of perfect sensor data, the generalized Hough method generally results in correctly identified instances of an object model. With uncertainty in the data, however, in steps (2) and (3) one really needs to vote not just for the nominal transformation, but for the full range of transformations consistent with the pairing of a given k-tuple (in this case triple) of model and image points. Our analysis provides a method for computing the range of values in the transformation space into which a vote should be cast:

1. Consider a pairing 3-tuples of model and image features.
2. For each such pair, determine the associated transformation, using the method of Section 4 to determine the nominal values of the transformation, and using equations (38) (39) (40) and (41) to determine the variation in each parameter.
3. Use the parameters of the transformation, together with the range of variation in each parameter, to index into a hash space, and at the indexed point, increment a counter.
4. Repeat for all possible pairings of 3-tuples.
5. Search the hash space for peaks in the stored votes, such peaks serving to hypothesize a pose of the object.

6.2 Effects of Error Sensitivity on the Hough Transform

Unfortunately, allowing for uncertainty in the data dramatically increases the chances of false peaks in the vote in hash space, because each tuple of model and image points votes for a (possibly large) range of transformations.

Earlier analysis of this effect [12] has shown that the sensitivity of the Hough transform as a tool for object recognition depends critically on its redundancy factor, defined as the fraction of the pose space into which a single data-model pairing casts a vote. This previous analysis was done for 2D objects and 2D images, and for 3D objects and 3D images. Here we examine the impact of this effect on using the Hough transform for recognizing 3D objects from 2D images. We use the analysis of the previous section to determine the average fraction of the parameter space specified by a triple of model and image points, given ϵ-bounded uncertainty in the sensing data. (In [12], experimental data from [25] were used to analyze the behavior, whereas here we derive analytic values.)

To do this, we simply find the expected range of values for $\delta\psi$, as given by equation (16), for $\delta\phi$, as given by equation (30), and for $\delta\theta$, as given by equations (33) and (34). We could do this by actually integrating these ranges for some distribution of parameters. An easier way of getting a sense of the method is to empirically sample these ranges. We have done this with the following experiment. We created a set of model features at random, then created sets of image features at random. We then selected matching triples of points from each set, and used them to compute a transformation, and the associated error bounds. For each of the rotational parameters, we measured the average range of variation predicted by our analysis. The positional uncertainty in the sensory data was set to be $\epsilon = 1, 3$ or 5. The results are summarized in Table 1, where we report both the average range of uncertainty in angle (in

	$\delta\psi$	$\delta\theta$	$\delta\phi$	δs	b_r	b_t	b
Average	.0503	.2351	.1348	.1781			
Normalized	.0080	.0374	.0429	.1215	$1.284e^{-5}$	$1.257e^{-5}$	$1.961e^{-11}$
Average	.1441	.5336	.2708	.3644			
Normalized	.0229	.0849	.0862	.2485	$1.676e^{-4}$	$1.131e^{-4}$	$4.710e^{-9}$
Average	.2134	.7927	.3524	.4630			
Normalized	.0340	.1262	.1122	.3158	$4.814e^{-4}$	$3.142e^{-4}$	$4.777e^{-8}$

Table 1: Ranges of uncertainty in the transformation parameters. Listed are the average range of uncertainty in each of the rotation parameters and the range of uncertainty in the multiplicative scale factor. The ranges are also normalized to the total possible range for each parameter (see text for details). Also indicated are b_r, the redundancy in rotation, b_t the redundancy in translation, assuming that the image dimension is $D = 500$, and b, the overall redundancy. Tables are for $\epsilon = 1, 3$ and 5 respectively.

radians), and the average range normalized by 2π in the case of θ and ψ and by π in the case of ϕ (since it is restricted to the range $-\pi/2 \leq \phi \leq \pi/2$).

The product of the three rotation terms, which we term b_r, defines the average fraction of the rotational portion of the pose space that is consistent with a pairing of model and image 3-tuples.

To get the overall redundancy factor (the fraction of the transformation space that is consistent with a given pairing of model and sensor points), we must also account for the translational and scale parameters. If D is the size of each image dimension, then the fraction of the translational portion of pose space consistent with a given pairing of three model and image points is

$$b_t = \frac{\pi \epsilon^2}{D^2}.$$

In the examples reported in Table 1, we used a value of $D = 500$, where the largest possible distance between model features in the image was 176 pixels.

To estimate the range of uncertainty in scale, we use the following method. Since the scale factor is a multiplicative one, we use $\log s$ as the 'key' to index into the hash space, so that when we account for uncertainty in scale, $s\delta s$ is transformed into $\log s + \log \delta s$. If we assume that $s_{\max}$ and $s_{\min}$ denote the maximum and minimum allowed scale factors, then the fraction of the scale dimension covered by the computed uncertainty range is

$$b_s = \frac{\log \delta s_{\max} - \log \delta s_{\min}}{\log s_{\max} - \log s_{\min}}.$$

In the case of the experiments described in Table 1, we used $s_{\max} = .13$ and $s_{\min} = .03$.

	$\epsilon = 1$	3	5
Hough	$1.984e^{-15}$	$1.447e^{-12}$	$3.101e^{-11}$
Estimate	$1.961e^{-11}$	$4.710e^{-9}$	$4.777e^{-8}$

Table 2: Comparing fractions of the pose space consistent with a match of 3 image and model points. The Hough line indicates the average size of such regions for different amounts of sensor uncertainty. The Estimate line indicates the corresponding sizes using the uncertainty bounds on the transformation parameters derived in the previous section.

The overall redundancy (fraction of the transformation space consistent with a given triple of model and image points) is

$$b = b_r b_t b_s. \tag{42}$$

Values for b are reported in Table 1.

For this particular example, one can see that while the uncertainty in $\delta\psi$ is quite small, the uncertainty in the other two rotational parameters can be large, especially for large values of ϵ. The redundancy in translation is a bit misleading, since it depends on the relative size of the object to the image. The uncertainty in scale, normalized to the total range of scale can in principle be quite large, though this also depends on the total range of possible values.

Note how dramatically the redundancy jumps in going from $\epsilon = 1$ to $\epsilon = 3$.

Evaluating the analysis

We are overestimating the region of possible transformations, and one obvious question is how bad is this overestimate. We can explore this by the following alternative analysis. We are basically considering the following question: Given 3 model points and 3 matching image points, what is the fraction of the space of possible transformations that is consistent with this match, given ϵ uncertainty in the sensed data? This is the same as asking the following question: For any pose, what is the probability that that pose applied to a set of 3 model points will bring them into agreement, modulo sensing uncertainty, with a set of 3 image points? If D is the linear dimension of the image (or the fraction of the image being considered), then this probability, under a uniform distribution assumption on image points, is

$$\left[\pi \left(\frac{\epsilon}{D}\right)^2\right]^3, \tag{43}$$

because the probability of any of the transformed points matching an image point is just the probability that it falls within the ϵ error disc, and by uniformity this is just the ratio of the area of that disc to the area of the image region.

The expression in equation (43) defines the best that we could do, if we were able to exactly identify the portion of pose space that is consistent with a match. To see how badly we are overestimating this region, we compare the results of Table 1 with those predicted by this model, as shown in Table 2. One can see from this table that our approximate method overestimates by about a factor of 1000. Considering this is distributed over a 6 dimensional space, this implies we are overestimating each parameter's range by about a factor of 3. These numbers are potentially misleading since they depend on the relative size of the object to the image. Nonetheless, they give an informal sense of the difference between the ideal Hough case and the estimates obtained by this method.

Using the analysis

Although these values may seem like an extremely small fraction of the 6 dimensional space that is filled by any one vote for a pairing of 3-tuples, it is important to remember that under the Hough scheme, all possible pairings of 3-tuples cast votes into the space, and there are on the order of m^3s^3 such tuples, where m is the number of known model features, and s is the number of measured sensory features. By this analysis, each such tuple will actually vote for a fraction b of the overall hash space. Even in the ideal limiting case of infinitesimal sized buckets in the transformation space, there is likely to be a significant probability of a false peak.

To see this, we can apply the analysis of [12]. In particular, the probability that a point in the pose space will receive a vote of size j can be approximated by the geometric distribution,

$$p_j \approx \frac{\lambda^j}{(1+\lambda)^{j+1}} \tag{44}$$

where $\lambda = m^3s^3b$. The probability of at least ℓ votes at a point is then

$$p_{\geq \ell} = 1 - \sum_{j=0}^{\ell-1} p_j = \left(\frac{\lambda}{1+\lambda}\right)^{\ell}. \tag{45}$$

That is, this expression denotes the fraction of the cells in pose space that will have votes at least as large as ℓ. In most recognition systems, it is common to set a threshold, t, on the minimum size correspondence that will be accepted as a correct solution. Thus we are concerned with the probability of a false positive, i.e. a set of at least t feature pairings accidentally masquerading as a correct solution. Suppose we set this threshold by assuming that a correct solution will have pairings of image features for $t = fm$ of the model features, where f is some fraction, $0 \leq f \leq 1$. Since we are using triples to define entries into the hash table, there will be $\binom{x}{3} \approx x^3$ votes cast at any point that is consistent with a transformation

	$\epsilon = 1$			$\epsilon = 3$			$\epsilon = 5$		
	$f = .25$	$.5$	$.75$	$f = .25$	$.5$	$.75$	$f = .25$	$.5$	$.75$
$\delta = 10^{-2}$	557	1114	1672	90	179	269	41	83	124
10^{-3}	487	974	1460	78	157	235	36	72	109
10^{-4}	442	885	1327	71	142	213	33	66	99

Table 3:

Approximate limits on the number of sensory features, such that the probability of a false positive of size fm is less than δ, shown for different fractions f, and different thresholds δ. The tables are for errors of $\epsilon = 1, 3$ and 5 respectively.

aligning x of the model features with image features. Thus, if we want the probability of a false positive accounting for fm of the model features to be less than some bound δ, we need

$$\left(\frac{\lambda}{1+\lambda}\right)^{(fm)^3} \leq \delta. \tag{46}$$

Substituting for λ and rearranging the equation leads to the following bound on the number of sensory features that can be tolerated under these conditions:

$$s \leq \frac{1}{m}\left[\frac{1}{b}\frac{1}{\delta^{\frac{-1}{f^3m^3}} - 1}\right]^{\frac{1}{3}}. \tag{47}$$

Following the analysis of [12] we can use the series expansion

$$a^x = \sum_{j=0}^{\infty} \frac{(x \ln a)^j}{j!}$$

together with a Taylor series expansion, to approximate this bound on s by:

$$s_{\text{lim}} \approx \frac{f}{\left[b \ln \frac{1}{\delta}\right]^{\frac{1}{3}}}\left[1 - \frac{\ln \frac{1}{\delta}}{6f^3m^3}\right] \approx \frac{f}{\left[b \ln \frac{1}{\delta}\right]^{\frac{1}{3}}}. \tag{48}$$

Note that to a first order approximation, the limit on the number of sensory features that can be tolerated, while keeping the probability of a false positive of size fm below some threshold δ is independent of the number of model features m, and only depends on the redundancy b (and hence the uncertainty ϵ), the fraction f of the model to be matched, and the threshold δ. To get a sense of the range of values for s, we chart in Table 3 the limiting values for s based on equation (48) and using values for b from Table 1.

One can see from this that, except in the case of very small sensor uncertainty, the Hough space very rapidly saturates, largely because of the m^3s^3 number of cases that cast votes into the space. As a consequence, the 3D-from-2D Hough transform will perform well only if some other process pre-selects a subset of the sensor features for consideration. Note that these numbers are based on the redundancy values associated with our derived approximation for the volume of Hough space associated with each pairing of model and image triples of features. As we saw, in the ideal case, the actual volume of Hough space is smaller, by a factor of about 1000 (Table 2). This means that the values for the limits on sensor clutter in Table 3 in the ideal case will be larger by roughly a factor of 10. (Of course, this also requires that one can find an efficient way of exactly determining the volume of Hough space consistent with a pairing of image and model features.) For the larger uncertainty values, this still leaves fairly tight limits on the amount of sensory data that can be accomodated. This analysis supports our earlier work [12], in which we showed using empirical data, that the Hough transform works well only when there is limited noise and scene clutter.

6.3 3D Alignment

Alignment methods differ from pose clustering, or generalized Hough, methods in that the computed transformation is directly applied to the model, and used to check for additional corresponding model and image features. In order to analyze the effects of sensory uncertainty on this type of recognition method, we need to know what happens when a model point is transformed and projected into the image. That is, what is the range of positions, about the nominal correct position, to which a transformed model point can be mapped? Determining this allows us to design careful verification algorithms, in which minimal regions in the image are examined for supporting evidence.

In this section, we describe how to use our analysis to bound the range of possible positions of a given model point that is projected into the image. In addition, we illustrate how the bounding regions we compute compare to the true regions of uncertainty. Lastly, we look at the implications of using our bounds to perform alignment-based recognition. In particular, we compute the probability of a false positive match, which is when model points transformed under an incorrect transform are aligned to random image points up to error.

A Simple Verification Algorithm

In the Alignment Method, pairs of 3 model and image points are used to hypothesize the pose of an object in the image. In other words, a method such as the one described in section 4 is used to compute the transformation associated with this correspondence, and then that transformation is applied to all model features, thereby mapping them into the image. To verify such an hypothesis, we need to determine which of the aligned features have a match in the image. While this should really be done on edge features, here we describe a method for verification from point features (the edge case is the subject of forthcoming work). The key is to approximate the smallest sized region in which to search for a match for each image, such that the correct match, if it exists, will not be missed. We can approximate such regions as follows:

1. Decompose each model point $\vec{P}$ into the natural coordinate system, so that it transforms as:
$$\alpha\widehat{m} + \beta\widehat{m}^{\perp} + \gamma\widehat{z} \mapsto s\Pi_{\widehat{z}}\left[\alpha\widehat{m}'''' + \beta\widehat{m}^{\perp''''} + \gamma\widehat{z}''''\right].$$
The transformation of the basis vectors is given by equations (9), (10) and (11). This allows us to determine the position of the nominally transformed model point.

2. Select sample values for $\delta\psi$, at some spacing, subject to the conditions of equation (38).

3. For each value, use equations (39), (40) and (41) to compute bounds on the variation in the other error parameters. This leads to a set of extremal variations on each of the parameters.

4. For each collection of error values in this set, perturb the nominal transformation parameters, and compute a new position for the transformed model point. Take the difference from the nominal point to determine an error offset vector.

5. Expand each error offset vector outward from the nominal point by an additional offset of 2ϵ to account for the translational uncertainty and the inherent uncertainty in sensing the point.

6. Add each error vector to the nominal point in the image, and take the convex hull of the result to get a good estimate of the range of feasible positions associated with a projected model point.

7. Search over this region for a matching image point.

8. If sufficiently many projected model points have a match, accept the hypothesized pose.

An example of this is shown in Figure 2. The figure was created by taking a random set of 3D points as a model, arbitrarily rotating and translating them, projecting them into the image and scaling with a random scale factor between .05 and .1, and perturbing the result randomly with error vectors of magnitude at most ϵ, resulting in a set of corresponding data

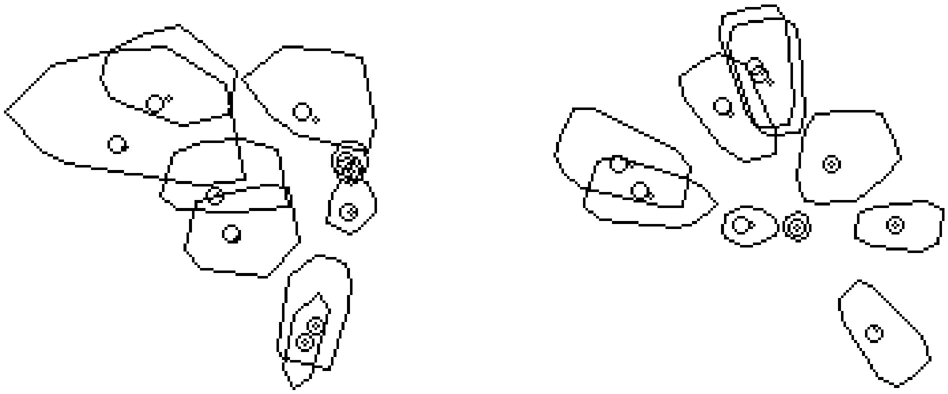


Figure 2:

Two examples of uncertainty regions, with perturbation of the data.

points. In this figure, the open circles represent a set of image points, each displayed as an ϵ-disc. The process is to match 3 model points to 3 data points whose positions are known up to ϵ-circles. Then, the match is used together with the parameters of the transformation to compute the uncertainty regions (displayed as polygons) and the crosses, which lie at the nominal location of the model. Note that the image points corresponding to the model points could fall anywhere within the polygons, so that simply searching an ϵ-circle for a match would not be sufficient.

One can see that the uncertainty regions vary considerably in size. To get a sense of this variation, we ran a series of trials as above, and collected statistics on the areas associated with each uncertainty region, over a large number of different trials of model and image points. We can histogram the areas of the observed discs, in terms of $\pi(2\epsilon)^2$ (the size of the basic disc of uncertainty). A sample histogram, normalized to sum to 1, is shown in Figure 3, and was based on 10000 different predicted regions of uncertainty. For the case of $\epsilon = 5$, the expected area of an uncertainty region is 2165 square pixels. For the case of $\epsilon = 3$, the expected area of an uncertainty region is 1028 square pixels. For $\epsilon = 1$, the expected area is 195 square pixels. In all cases, the maximum separation of image features was $m_{\max} = 176$.

Using the analysis

One advantage of knowing the bounds on uncertainty in computing a transform is that they can be used to overestimate the regions of the image into which aligned features project. This gives us a way of designing careful verification systems, in which we are guaranteed to find a

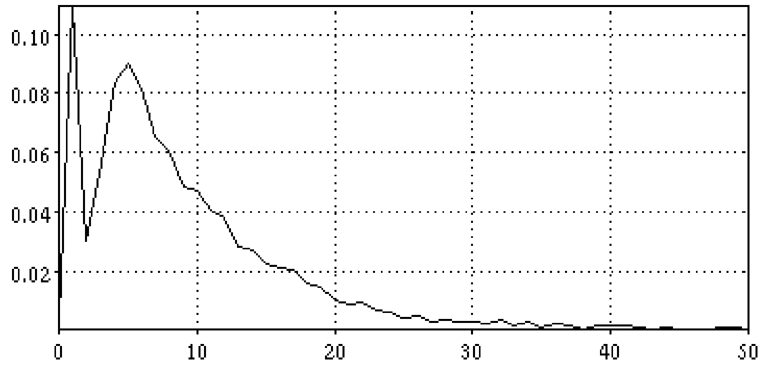


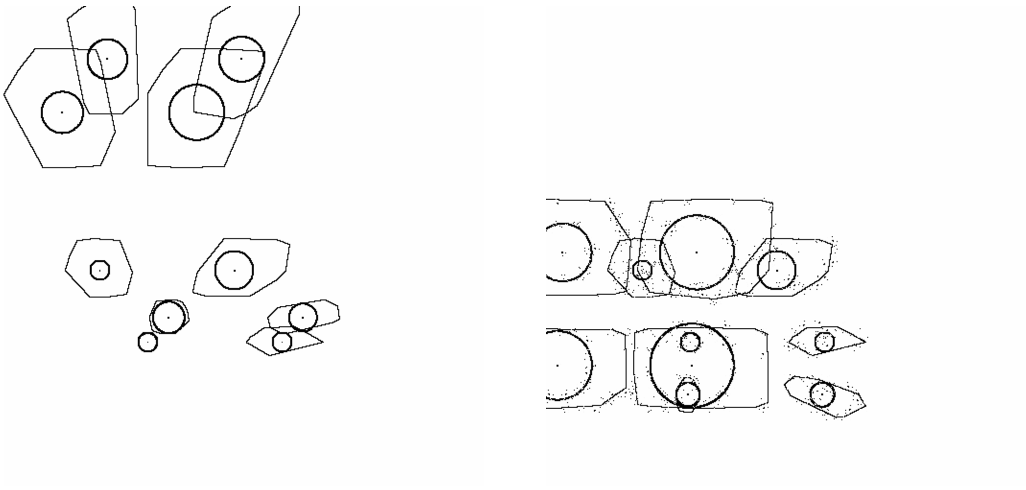
Figure 3:

Graph of the distribution of areas of uncertainty regions, as measured in the image, for the case of $\epsilon = 3$. The horizontal axis is in units of $\pi(2\epsilon)^2$. The vertical axis records the fraction of the distribution of uncertainty regions with that size.

correct corresponding image feature, if it exists, while at the same time keeping the space over which to search for such features (and thus the number of false matches) relatively small. Of course, we know that our estimates err on the high side, and it is useful to see how much our projected image regions overestimate the range of possible positions for matches.

To do this, we have run the following experiment. We took a 3D model of a telephone, and created an image of that model under some arbitrary viewing condition. We then chose a corresponding triple of image and model features and used the method described here both to determine the alignment transformation of the model and to determine our estimates of the associated uncertainty regions for each feature, based on assuming ϵ-discs of uncertainty in the image features. For comparison, we took a sampling of points on the boundary of the ϵ-disc around each of the basis images points, computed the associated alignment transformation, and projected each additional model features into the image. We collected the set of positions for each projected model point as we allowed the basis points to vary over their ϵ-discs, and used this to create regions of uncertainty about each aligned point. This should be a very close approximation to the actual region of uncertainty. We compare these regions to our estimated regions in Figure 4. One can see that our method does overestimate the uncertainty regions, although not drastically.

Finally, we can use our estimates of the uncertainty regions to estimate the probability that a random pairing of model and image bases will collect votes from other model points. That is, if we use a random alignment of the model and project the remaining transformed model points into the image, on average each such point will define an uncertainty region of



Example b

Figure 4:

Comparison of ideal uncertainty regions with estimated regions. Each feature in the model is projected according to the nominal transformation, as illustrated by the points. The dark circles show the ideal regions of uncertainty. The larger enclosing convex regions show the estimated uncertainty regions computed by our method. Two different solutions are shown.

the size computed above. If we consider an image of dimension $D = 500$, then the selectivity of the method (i.e. the probability that each model point will find a potentially matching image point in its uncertainty region) is 0.000781, 0.00411 and 0.00866 for $\epsilon = 1, 3$ and 5 respectively. By comparison, the selectivity for the case of a planar object in arbitrary 3D position and orientation, for the same level of sensor uncertainty is $0.000117, 0.001052$ and 0.002911 respectively (Table 1 of [14]). Although they represent overestimates, these results suggest that the selectivity of recognition methods applied to 3D objects should be only slightly worse than when applied to 2D objects.

To see this, we can use the analysis of [14] to estimate limits on the number of image features that can be tolerated, while maintaining a low false positive rate. Recapping from that earlier work, the false positive rate is computed by the following method:

1. The selectivity of the method is defined by the probability that the uncertainty region associated with a projected model point contains an image point, and this is just the redundancy (fraction of the transformation space that is consistent with a given triple of model and image points) b, as defined in equation (42).
2. Since each model point is projected into the image, the probability that a given model

point matches at least one image point is

$$p = 1 - (1 - \overline{b})^{s-3}$$

because the probability that a particular model point is not consistent with a particular image point is $(1 - \overline{b})$ and by independence, the probability that all $s - 3$ points are not consistent with this model point is $(1 - \overline{b})^{s-3}$.

3. The process is repeated for each model point, so the probability of exactly k of them having a match is

$$q_k = \binom{m-3}{k} p^k (1-p)^{m-3-k}. \tag{49}$$

Further, the probability of a false positive identification of size at least k is

$$w_k = 1 - \sum_{i=0}^{k-1} q_i.$$

Note that this is the probability of a false positive for a particular sensor basis and a particular model basis.

4. This process can be repeated for all choices of model bases, so the probability of a false positive identification for a given sensor basis with respect to any model basis is

$$e_k = 1 - (1 - w_k)^{\binom{m}{3}}. \tag{50}$$

Thus, we can compute limits on s such that $e_k \leq \delta$ where δ is some threshold on the false positive rate, and where k is taken to be fm for some fraction $0 \leq f \leq 1$. In Table 4, we list these limits, computed using equation (50) and values of b obtained from the ratio of areas described in equation (42).

While these results give a sense of the limits on alignment, they are potentially slightly misleading. What they say is that if we used the derived bounds to compute the uncertainty regions in which to search for possible matches, and we use no other information to evaluate a potential match, then the system saturates fairly quickly. As we know from Figure 4, however, our method overestimates the uncertainty regions, and a more correct method, such as that described earlier in which one uses sample of the basis uncertainty regions to trace out ideal uncertainty regions for other points, would lead to much smaller uncertainty regions, smaller values for the redundancy b, and hence a more forgiving verification system. Also, one could clearly augment the test described here to incorporate additional constraints on the pose and its uncertainty that can be obtained by using additional matches of model and sensory features

	$\epsilon = 1$			$\epsilon = 3$			$\epsilon = 5$		
	$f = .25$	$.5$	$.75$	$f = .25$	$.5$	$.75$	$f = .25$	$.5$	$.75$
$\delta = 10^{-2}$	149	480	1069	30	93	205	16	45	98
10^{-3}	139	457	1028	28	89	197	15	43	95
10^{-4}	130	437	991	27	85	190	14	42	90

Table 4:

Approximate limits on the number of sensory features, such that the probability of a false positive of size fm is less than δ, shown for different fractions f, and different thresholds δ. The tables are for errors of $\epsilon = 1, 3$ and 5 respectively, and for $m = 200$.

to further limit the associated uncertainty (e.g. given 4 pairs of matched points, use sets of 3 to compute regions of uncertainty in pose space, intersect these regions and use the result to determine the poses uncertainty).

7 Summary

A number of object recognition systems compute the pose of a 3D object from using a small number of corresponding model and image points. When there is uncertainty in the sensor data, this can cause substantial errors in the computed pose. We have derived expressions bounding the extent of uncertainty in the pose, given ϵ-bounded uncertainty in the measurement of image points. The particular pose estimation method that we analyzed is that of [12, 13], which determines the pose from 3 corresponding model and image points under a weak perspective imaging model. Similar analyses hold for other related methods of estimating pose.

We then applied this analysis in order to analyze the effectiveness of two classes of recognition methods that use pose estimates computed in this manner: the generalized Hough transform and alignment. We found that in both cases, the methods have a substantial chance of making a false positive identification (claiming an object is present when it is not), for even moderate levels of sensor uncertainty (a few pixels).

Acknowledgments

The development and presentation of these results were considerably improved by observations and comments from David Jacobs.

References

[1] Ayache, N. & O.D. Faugeras, 1986, "HYPER: A new approach for the recognition and positioning of two-dimensional objects," *IEEE Trans. Patt. Anal. & Mach. Intell.*, Vol. 8, no. 1, pp. 44–54.

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

[3] Basri, R. & S. Ullman, 1988, "The Alignment of Objects with Smooth Surfaces," *Second Int. Conf. Comp. Vision*, 482–488.

[4] Besl, P.J. & R.C. Jain, 1985, "Three-dimensional object recognition," *ACM Computing Surveys*, Vol. 17, no. 1, pp. 75–154.

[5] Bolles, R.C. & R.A. Cain, 1982, "Recognizing and locating partially visible objects: The Local Feature Focus Method," *Int. J. Robotics Res.*, Vol. 1, no. 3, pp. 57–82.

[6] Bolles, R.C. & M. A. Fischler, 1981, "A RANSAC-based approach to model fitting and its application to finding cylinders in range data," *Seventh Int. Joint Conf. Artif. Intell.*, Vancouver, B.C., Canada, pp. 637–643.

[7] Bolles, R.C. & P. Horaud, 1986, "3DPO: A Three-dimensional Part Orientation System," *Int. J. Robotics Res.*, Vol. 5, no. 3, pp. 3–26.

[8] Cass, T.A., 1988, "A robust parallel implementation of 2D model-based recognition," *IEEE Conf. Comp. Vision, Patt. Recog.*, Ann Arbor, MI, pp. 879–884.

[9] Cass, T.A., 1990, "Feature matching for object localization in the presence of uncertainty," MIT AI Lab Memo 1133.

[10] Faugeras, O.D. & M. Hebert, 1986, "The representation, recognition and locating of 3-D objects," *Int. J. Robotics Res.* Vol. 5, no. 3, pp. 27–52.

[11] Grimson, W.E.L., 1990, *Object Recognition by Computer: The role of geometric constraints*, MIT Press, Cambridge.

[12] Grimson, W.E.L. & D.P. Huttenlocher, 1990, "On the Sensitivity of the Hough Transform for Object Recognition," *IEEE Trans. PAMI* **12**(3), pp. 255–274.

[13] Grimson, W.E.L. & D.P. Huttenlocher, 1991, "On the Verification of Hypothesized Matches in Model-Based Recognition", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **13**(12) pp. 1201–1213.

[14] Grimson, W.E.L., D.P. Huttenlocher & D.W. Jacobs, 1992, "Affine Matching With Bounded Sensor Error: A Study of Geometric Hashing and Alignment", *Proceedings of the Second European Conference on Computer Vision*, pp. 291–306.

[15] Heller, A.J. & J.L. Mundy, 1990, "Benchmark Evaluation of a Model-Based Object Recognition System", *Proceedings DARPA Image Understanding Workshop*, pp. 727–741.

[16] Huttenlocher, D.P. and S. Ullman, 1987, "Object Recognition Using Alignment", *Proceedings of the First International Conference on Computer Vision*, pp. 102-111.

[17] Huttenlocher, D.P. & S. Ullman, 1990, "Recognizing Solid Objects by Alignment with an Image," *Inter. Journ. Comp. Vision* **5**(2):195–212.

[18] Lamdan, Y., J.T. Schwartz & H.J. Wolfson, 1988a, "On Recognition of 3-D Objects from 2-D Images," *IEEE Int. Conf. on Rob. Aut.* pp. 1407–1413.

[19] Lamdan, Y., J.T. Schwartz & H.J. Wolfson, 1988b, "Object Recognition by Affine Invariant Matching," *IEEE Conf. on Comp. Vis. and Patt. Recog.* pp. 335–344.

[20] Lamdan, Y., J.T. Schwartz & H.J. Wolfson, 1990, "Affine Invariant Model-Based Object Recognition," *IEEE Trans. Robotics and Automation*, vol. 6, pp. 578–589.

[21] Lamdan, Y. & H.J. Wolfson, 1988, "Geometric Hashing: A General and Efficient Model-Based Recognition Scheme," *Second Int. Conf. on Comp. Vis.* pp. 238–249.

[22] Lamdan, Y. & H.J. Wolfson, 1991, "On the Error Analysis of 'Geometric Hashing'," *IEEE Conf. on Comp. Vis. and Patt. Recog.* pp. 22–27.

[23] S. Linainmaa, D. Harwood, and L.S. Davis, "Pose Determination of a Three-Dimensional Object Using Triangle Pairs", CAR-TR-143, Center for Automation Research, University of Maryland, 1985.

[24] T. Silberberg, D. Harwood and L.S. Davis, "Object Recognition Using Oriented Model Points," *Computer Vision, Graphics and Image Processing*, Vol. 35, pp. 47-71, 1986.

[25] Thompson, D.W. & J.L. Mundy, 1987, "Three-dimensional model matching from an unconstrained viewpoint," Proc. IEEE Int. Conf. Robotics Autom., Raleigh, NC, pp. 208–220.