

# First order error propagation of the Procrustes method for 3-D attitude estimation

Leo Dorst `leo@science.uva.nl`

Informatics Institute, University of Amsterdam

Kruislaan 403, 1098 SJ Amsterdam, The Netherlands

## Abstract

The well known Procrustes method determines the optimal rigid body motion that registers two point clouds by minimizing the square distances of the residuals. In this paper we perform a complete error analysis of this method for the 3D case, fully specifying how directional noise in the point clouds affects the estimated parameters of the rigid body motion. These results are much more specific than the error bounds which have been established in numerical analysis. We provide an intuitive understanding of the outcome to facilitate direct use in applications.

**Index terms:** rigid body motion analysis, pose estimation, attitude estimation, Procrustes method, orthogonal Procrustes problem, perturbation analysis, error propagation, polar decomposition

# 1 Procrustes attitude estimation

## 1.1 Problem statement

There is a common need in vision and robotics to estimate the rigid body transformation between labeled sets of points. The problem is concisely formulated as follows. Let us assume that we have a set of labeled points  $\{\mathbf{X}'_i\}_{i=1}^n$ , and that this has been brought into a new position and orientation. The resulting point set is  $\{\mathbf{Y}'_i\}_{i=1}^n$ , where, approximately:

$$\mathbf{Y}'_i = \mathbf{R}\mathbf{X}'_i + \mathbf{T}.$$

Both the  $\mathbf{X}'_i$  and  $\mathbf{Y}'_i$  will be measured point locations, so both point sets may have some noise relative to this ideal equation (typically describable as some low-level non-isotropic Gaussian noise). To estimate  $\mathbf{R}$  and  $\mathbf{T}$ , one draws up the very reasonable cost function measuring the sum of the squared distances between corresponding points:

$$c^2 = \sum_{i=1}^n \|\mathbf{Y}'_i - (\mathbf{R}\mathbf{X}'_i + \mathbf{T})\|^2. \quad (1)$$

Minimization of this then gives sensible estimates of  $\mathbf{T}$  and  $\mathbf{R}$ . This is known as the *Procrustes problem* (estimation of  $\mathbf{R}$  when  $\mathbf{T} = 0$  is the ‘orthogonal Procrustes problem’), and the solution has long been known (and reinvented repeatedly), in various formulations in various fields from biomechanics [5](1952), satellite control [4][16](1960s) and statistical shape analysis [13](1978) to vision and robotics [1][7][8][17](late 1980s). This solution is

optimal (in the sense of least variance) if the data points have identical and isotropic Gaussian noise in the data points  $\mathbf{Y}'_i$  (if not, one should consider using the non-linear method of [12]).

An important question in a practical application: of the Procrustes method is: *how does noise in the data actually affect the computed rigid body motion parameters?* Some bounds are known (see e.g. [15]), but since the parametric noise depends rather strongly on the shape of the point cloud, one would prefer a more detailed understanding of the directional characteristics of the noise. We provide that in this paper, in the form of the calculation of the error propagation, for the 3-D estimation problem, as covariances of the rigid body motion parameters, computed to first order in the noise. We follow it by some rules-of-thumb to be used as a guide in practical usage of the Procrustes method, and illustrate the results by simulations.

## 1.2 Standard solution

We briefly summarize the standard solution of the orthogonal Procrustes problem of minimizing  $c^2$  in eq.(1), to introduce notation and representation. First of all, the cost criterion can be rewritten by considering the point clouds  $\mathbf{X}'_i$  and  $\mathbf{Y}'_i$  relative to their centroids  $\bar{\mathbf{X}}$  and  $\bar{\mathbf{Y}}$ . Let us denote the relative vectors as  $\mathbf{X}_i$  and  $\mathbf{Y}_i$ , then:

$$c^2 = \sum_{i=1}^n \|\mathbf{Y}'_i - (\mathbf{R}\mathbf{X}'_i + \mathbf{T})\|^2 = n \|\bar{\mathbf{Y}} - \mathbf{R}\bar{\mathbf{X}} - \mathbf{T}\|^2 + \sum_{i=1}^n \|\mathbf{Y}_i - \mathbf{R}\mathbf{X}_i\|^2 \quad (2)$$

We can minimize the first contribution using the translation estimate

$$\hat{\mathbf{T}} = \bar{\mathbf{Y}} - \mathbf{R}^* \bar{\mathbf{X}}, \quad (3)$$

once we have found the rotation estimate  $\mathbf{R}^*$ . The second term involves only the rotation and therefore produces the optimal  $\mathbf{R}^*$ . To describe its solution, we first use the column vectors  $\mathbf{X}_i$  and  $\mathbf{Y}_i$  to make two  $3 \times n$  matrices of the points relative to their centroids:  $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n] = [\mathbf{X}'_1 - \bar{\mathbf{X}}, \mathbf{X}'_2 - \bar{\mathbf{X}}, \dots, \mathbf{X}'_n - \bar{\mathbf{X}}]$  and  $\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_n] = [\mathbf{Y}'_1 - \bar{\mathbf{Y}}, \mathbf{Y}'_2 - \bar{\mathbf{Y}}, \dots, \mathbf{Y}'_n - \bar{\mathbf{Y}}]$ . Then we form

$$\mathbf{A} = \mathbf{Y}\mathbf{X}^\top.$$

The least variance estimate of the orthogonal transformation which aligns all  $\mathbf{X}_i$  and  $\mathbf{Y}_i$  is now

$$\mathbf{R}^* = \mathbf{A}(\mathbf{A}^\top \mathbf{A})^{-1/2} \quad (4)$$

if  $\mathbf{A}$  is non-singular. This has been shown (e.g. in [13]) to be the best orthogonal matrix (in the sense of the Frobenius norm) solving the correspondence between  $\mathbf{X}$  and  $\mathbf{Y}$ . This best transformation is a rotation if  $\det(\mathbf{R}^*) = 1$  or a rotoinversion (a.k.a. rotation/reflection) if  $\det(\mathbf{R}^*) = -1$ .

### 1.3 Alternative formulations

The solution eq.(4) can be given in various different but largely equivalent forms. In terms of a singular value decomposition (SVD) of  $\mathbf{A} = \mathbf{Y}\mathbf{X}^\top$  as  $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top$ , the optimal rotation

is ([16], [6]):

$$\mathbf{R}^* = \mathbf{U} \text{diag}(1, 1, \det[\mathbf{UV}^\top]) \mathbf{V}^\top. \quad (5)$$

The diagonal matrix ensures that this always produces the best rotation and never a rotoinversion, for it makes  $\det(\mathbf{R}^*) = 1$ . (This is a useful refinement over eq.(4).) This SVD formulation has been the basis for some error analysis approaches: in [9] it is used to find the covariance of fitting an orthonormal frame, with independent variations in the three vectors  $\mathbf{e}_i$  of that frame – which is unfortunately a rather special case.

The literature on the analysis of perturbation bounds (such as [14], [15]) often uses the more convenient *polar decomposition*, in which  $\mathbf{A}$  is written as the product of an orthogonal matrix and a symmetric matrix. It can be derived simply from the SVD of  $\mathbf{A}$  by rewriting that as  $\mathbf{A} = (\mathbf{UV}^\top) (\mathbf{V}\mathbf{S}\mathbf{V}^\top)$ : the first factor is an orthogonal matrix (and in fact equal to  $\mathbf{R}^*$  if we merely want to limit the solution to the best orthogonal mapping), the second factor is a symmetric matrix  $\mathbf{S}$  (even positive semi-definite). This is the decomposition we will use for our error analysis. In what follows we revert to the ‘best orthogonal matrix’ formulation, not including the diagonal matrix to find the best rotation. This is done for simplicity in the math: the error analysis for rotations and rotoinversion is exactly the same, but the non-linear flipping between the two possibilities as  $\det(\mathbf{UV}^\top)$  changes sign is hard to model. In practice, with enough non-degenerate points, that sign is stable under noise anyway, and will be positive if the data was known to be generated by a rotation.

As [3] has shown recently, all representations of the solution give the same numerically stable results (when the point clouds are not degenerate).

## 2 Perturbation analysis

### 2.1 Characterization of orthogonal transformation errors

For an orthogonal matrix  $\mathbf{R}$ , we have  $\mathbf{R}\mathbf{R}^\top = \mathbf{I}$ . A noisy perturbation of  $\mathbf{R}$  to  $\mathbf{R} + d\mathbf{R}$  should also be orthogonal to first order in the error  $d\mathbf{R}$  and therefore:

$$\mathbf{I} = (\mathbf{R} + d\mathbf{R})(\mathbf{R} + d\mathbf{R})^\top = \mathbf{R}\mathbf{R}^\top + d\mathbf{R}\mathbf{R}^\top + \mathbf{R}d\mathbf{R}^\top + O(d\mathbf{R}d\mathbf{R}^\top) \approx \mathbf{I} + d\mathbf{R}\mathbf{R}^\top + (d\mathbf{R}\mathbf{R}^\top)^\top.$$

Therefore, to first order,  $d\mathbf{R}\mathbf{R}^\top = -(d\mathbf{R}\mathbf{R}^\top)^\top$ . Defining the *relative transformation error*  $\delta\mathbf{R}$  as:

$$\delta\mathbf{R} \equiv d\mathbf{R}\mathbf{R}^{-1} = d\mathbf{R}\mathbf{R}^\top,$$

it is clear that this matrix must be skew-symmetric. For a more compact representation, remember that any skew-symmetric map  $\mathbf{M}$  in 3D is equivalent to the cross product with an appropriately chosen vector. That vector is in the 1D kernel (nullspace) of the mapping  $\mathbf{M}$ , with the appropriate magnitude. Let us simply denote it by  $\text{vec}(\mathbf{M})$ , then

$$\mathbf{M}\mathbf{z} = \text{vec}(\mathbf{M}) \times \mathbf{z}, \text{ for all } \mathbf{z}.$$

Vice versa, if the components of the axis are known as  $\text{vec}(\mathbf{M}) = \mathbf{m} = (m_1, m_2, m_3)^\top$ , we can construct the associated operator as  $\mathbf{M} = \mathbf{m}^\times$ , where the latter notation is defined

through:

$$\mathbf{M}\mathbf{z} = \mathbf{m} \times \mathbf{z} \equiv \mathbf{m}^\times \mathbf{z} = \begin{pmatrix} 0 & -m_3 & m_2 \\ m_3 & 0 & -m_1 \\ -m_2 & m_1 & 0 \end{pmatrix} \mathbf{z}. \quad (6)$$

Now we use these notations to characterize the perturbation. Let the orthogonal transformation  $\mathbf{R}$  be perturbed to the new orthogonal transformation  $\mathbf{R} + d\mathbf{R}$ . We rewrite this disturbance multiplicatively, so that the perturbed orthogonal transformation is obtained by a small additional rotation  $\mathbf{R}_{d\mathbf{a}}$  applied to  $\mathbf{R}$ :

$$\mathbf{R} + d\mathbf{R} = \mathbf{R}_{d\mathbf{a}}\mathbf{R}.$$

It follows that  $\mathbf{R}_{d\mathbf{a}}$  is a small disturbance of the identity rotation, expressible as a relative rotation error, which is a skew-symmetric map characterizable by a vector:

$$\mathbf{R}_{d\mathbf{a}} = \mathbf{I} + d\mathbf{R}\mathbf{R}^{-1} = \mathbf{I} + \delta\mathbf{R} \equiv \mathbf{I} + d\mathbf{a}^\times. \quad (7)$$

This defines  $d\mathbf{a}$  as

$$d\mathbf{a}^\times = \delta\mathbf{R} \quad \text{or, equivalently,} \quad d\mathbf{a} = \text{vec}(\delta\mathbf{R}). \quad (8)$$

It is easily verified from eq.(7) that  $d\mathbf{a}$  is in fact the axis of the small rotation  $\mathbf{R}_{d\mathbf{a}}$ , since the kernel of  $d\mathbf{a}^\times$  is the solution to the eigenvalue problem determining its axis:

$$\mathbf{x} = \mathbf{R}_{d\mathbf{a}}\mathbf{x} = (\mathbf{I} + d\mathbf{a}^\times)\mathbf{x} = \mathbf{x} + d\mathbf{a}^\times\mathbf{x}.$$

We therefore call  $\mathbf{da}$  the *error axis vector* of the perturbation. To reiterate precisely,  $\mathbf{da}$  is the axis vector of the extra rotation that needs to be done after  $\mathbf{R}$  to achieve the total perturbed transformation  $\mathbf{R} + \mathbf{dR}$ . Even if  $\mathbf{R}$  is a rotation, it is *not* correct to say that the new rotation axis is  $\mathbf{a} + \mathbf{da}$ , the actual relationship is more subtle (and may be gleaned from [10]).

We will characterize the perturbation using this vector  $\mathbf{da}$ , for instance computing its variance as a consequence of the noise variances in the data  $\mathbf{X}_i$  and  $\mathbf{Y}_i$ . This convenient characterization of the error is also used in [12] for rotations.

## 2.2 Perturbation of the orthogonal transformation

Let us characterize the errors in the  $\mathbf{X}_i$  and  $\mathbf{Y}_i$  as small additional matrices  $\mathbf{dX}$  and  $\mathbf{dY}$  representing the errors in  $\mathbf{X}$  and  $\mathbf{Y}$ . Then the error propagation into  $\mathbf{A} = \mathbf{YX}^\top$  is to first order

$$\mathbf{dA} = \mathbf{dYX}^\top + \mathbf{YdX}^\top. \quad (9)$$

Since there is an error in  $\mathbf{A}$ , there will be an error in the Procrustes estimator

$$\mathbf{R}^* = \mathbf{A}(\mathbf{A}^\top\mathbf{A})^{-1/2}. \quad (10)$$

Our goal is to give an explicit expression for this error, to first order, expressed in  $\mathbf{dX}$  and  $\mathbf{dY}$ . In principle, we do this through a first order Taylor-series expansion in a suitable representation of the estimator. The direct matrix representation eq.(10) is not easy to

work with: differentiating non-linear functions of matrices is not elementary. We also avoid the SVD of  $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top$ , since degenerate singular values mean that  $\mathbf{U}$  and  $\mathbf{V}$  do not have stable differentials (their eigenvector columns tend to permute with even a small amount of noise in  $\mathbf{A}$ ). Instead, we use the polar decomposition of  $\mathbf{A}$ , writing  $\mathbf{A}$  as the product of an orthogonal matrix  $\mathbf{R}$  and a symmetric matrix  $\mathbf{S}$

$$\mathbf{A} = \mathbf{R}\mathbf{S}. \tag{11}$$

As we saw in Section 1.3,  $\mathbf{R}$  is precisely the Procrustes estimation (from now on we denote it as  $\mathbf{R}$  rather than  $\mathbf{R}^*$ , to unclutter our formulas). As to the meaning of  $\mathbf{S}$ , if noise were absent  $\mathbf{Y}$  would equal  $\mathbf{R}\mathbf{X}$ , and then  $\mathbf{S} = \mathbf{R}^\top\mathbf{A} = \mathbf{R}^\top\mathbf{Y}\mathbf{X}^\top = \mathbf{R}^\top\mathbf{R}\mathbf{X}\mathbf{X}^\top = \mathbf{X}\mathbf{X}^\top$ . Thus in the noise-free case,  $\mathbf{S}$  is the *inertia tensor* of the point cloud (assuming unit mass for each point), and therefore characterizes its principal shape. (If one needs to compute  $\mathbf{S}$  in practice, one can only do so based on the actual measurements  $\mathbf{X}$  and  $\mathbf{Y}$ , and this causes some first order perturbation dependent on the amount of orthogonal transformation since now  $\mathbf{Y}$  does not exactly equal  $\mathbf{R}\mathbf{X}$ . Still the intuition of ‘inertia tensor’ is basically correct.)

Now do a first order perturbation of eq.(11):

$$d\mathbf{A} = (\mathbf{R} + d\mathbf{R})(\mathbf{S} + d\mathbf{S}) - \mathbf{R}\mathbf{S} \approx d\mathbf{R}\mathbf{S} + \mathbf{R}d\mathbf{S}.$$

Because they are defined through a polar composition,  $\mathbf{R}$  is orthogonal and  $\mathbf{S}$  is symmetric; and for the same reason,  $(\mathbf{R} + d\mathbf{R})$  is orthogonal and  $(\mathbf{S} + d\mathbf{S})$  is symmetric. It follows that

$dRR^T$  is skew, and that  $dS$  is symmetric. Therefore the relative error  $\delta A$  in  $A$  equals

$$\delta A \equiv dAA^{-1} \approx dRR^T + R dS S^{-1} R^T = \delta R + R dS S^{-1} R^T \quad (12)$$

By transposition,

$$(\delta A)^T \approx -\delta R + R S^{-1} dS R^T.$$

We use this second equation to express  $dS$  in terms of  $\delta R$  and  $(\delta A)^T$ :

$$dS \approx S R^T ((\delta A)^T - (\delta R)^T) R$$

and substituting this in eq.(12) we get an equation free of  $dS$ :

$$\bar{S} \delta R - (\delta R)^T \bar{S} \approx \bar{S} \delta A - (\delta A)^T \bar{S}, \quad (13)$$

where we defined the shorthand  $\bar{S} \equiv R S^{-1} R^T$ . Solving this equation for  $\delta R$  requires quite a bit of fairly ingenious algebraic manipulation, see appendix ???. The solution is most easily specified as the error axis vector  $\mathbf{da} = \text{vec}(\delta R)$ :

$$\mathbf{da} = \text{vec}(\delta R) = R(\text{tr}(S)I - S)^{-1} R^T \left( \sum_i (R\mathbf{X})_i \times d\mathbf{Y}_i + (Rd\mathbf{X})_i \times \mathbf{Y}_i \right) \quad (14)$$

for  $A = YX^T = RS$ . This is the central result of this paper.

## 2.3 Interpretation of error formula

The expression for  $\mathbf{d}\mathbf{a} = \text{vec}(\delta\mathbf{R})$  is not too hard to interpret. Ideally,  $\mathbf{R}\mathbf{X}_i$  and  $\mathbf{Y}_i$  should have the same direction and length. The cross product term accumulates those parts of the errors in  $\mathbf{X}$  and  $\mathbf{Y}$  that are perpendicular to the desired direction. Only those will affect the error in the orthogonal transformation (the other parts would go into a scaling error, if we were using the general Procrustes method).

This measure of mismatch between  $\mathbf{R}\mathbf{X}$  and  $\mathbf{Y}$  is weighted by a factor related to the shape of the point cloud as encoded mainly in the symmetric factor  $\mathbf{S}$  of the polar decomposition of  $\mathbf{A} = \mathbf{Y}\mathbf{X}^\top$ . As we have seen,  $\mathbf{S}$  is closely related to the inertia tensor of the cloud shape, and this suggests an interpretation for the term  $\mathbf{R}(\text{tr}(\mathbf{S})\mathbf{I} - \mathbf{S})^{-1}\mathbf{R}^\top$ . It is the  $\mathbf{R}$ -transformed version of a term we will denote  $\mathbf{H}$ :

$$\mathbf{H} \equiv (\text{tr}(\mathbf{S})\mathbf{I} - \mathbf{S})^{-1}$$

It is insightful to relate the eigenvalue structure of  $\mathbf{S}$  to  $\mathbf{H}$ . We derive that when

$$\mathbf{S} = \mathbf{V} \text{diag}(\lambda_1, \lambda_2, \lambda_3) \mathbf{V}^\top \quad \text{with } \lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$$

then

$$\mathbf{H} = \mathbf{V} \text{diag}\left(\frac{1}{\lambda_2 + \lambda_3}, \frac{1}{\lambda_3 + \lambda_1}, \frac{1}{\lambda_1 + \lambda_2}\right) \mathbf{V}^\top \quad \text{with } \frac{1}{\lambda_2 + \lambda_3} \geq \frac{1}{\lambda_3 + \lambda_1} \geq \frac{1}{\lambda_1 + \lambda_2} \geq 0.$$

Therefore the weighting matrix  $\mathbf{H}$  has a similar ordering to the inertial shape of the cloud, but the terms are ‘harmonized’ in a manner which mutes their ratios. For example, if the point cloud is very elongated as in  $\mathbf{S} = \text{diag}(1, 0.01, 0.01)$ , then  $\mathbf{H} \approx \text{diag}(50, 1, 1)$ , half as elongated (but 50 times larger). If the point cloud is very flat and circular as in  $\mathbf{S} = \text{diag}(1, 1, 0.01)$ , then  $\mathbf{H} \approx \text{diag}(1, 1, 0.5)$  is much less flat, even almost spherical. Of course a spherical cloud  $\mathbf{S} = \text{diag}(1, 1, 1)$  remains spherical, with  $\mathbf{H} = \text{diag}(0.5, 0.5, 0.5)$ .

If the two smallest singular values of  $\mathbf{S}$  are small (as happens for the line-like point cloud in the example), there may be stability problems in the estimation. This corresponds to the results on perturbation bounds for the Procrustes method in numerical analysis literature such as [15]. We will discuss the correspondence in Section 2.7.

## 2.4 Error covariance propagation

We are interested in a measure of the distribution of  $\mathbf{da}$ . Note that  $\mathbf{da}$  involves the sum of  $n$  error terms. For sufficiently large  $n$ , the central limit theorem tells us that  $\mathbf{da}$  will be distributed normally if the  $\{\mathbf{dX}_i\}$  and  $\{\mathbf{dY}_j\}$  are uncorrelated (and even when they have some small correlation). Under this assumption, the first and second moments are therefore propagated through the linear mapping eq.(14) of the errors  $\mathbf{dX}_i$  and  $\mathbf{dY}_i$  to  $\mathbf{da}$ .

Determine the first moment is straightforward from eq.(14). Note that if the noise  $\mathbf{dX}_i$  and  $\mathbf{dY}_i$  has zero mean, so does  $\mathbf{da}$ .

For the second moment, the contribution to the covariance of  $\mathbf{da}$  of the covariances  $\mathbf{C}_X$

in the  $\mathbf{X}$ -error and  $\mathbf{C}_Y$  in the  $\mathbf{Y}$ -error are immediate from the linear relationship of eq.(14):

$$\mathbf{C}_{\text{da}} = -\mathbf{R}\mathbf{H}\mathbf{R}^\top \left( \sum_i (\mathbf{R}\mathbf{X})_i^\times \mathbf{C}_Y (\mathbf{R}\mathbf{X})_i^\times + \sum_i \mathbf{Y}_i^\times (\mathbf{R}\mathbf{C}_X \mathbf{R}^\top) \mathbf{Y}_i^\times \right) \mathbf{R}\mathbf{H}\mathbf{R}^\top \quad (15)$$

where the minus sign is due to the skewness of the  $\times$ -operators, see eq.(6). If you prefer, we can rewrite this into matrix form as (see appendix A for a derivation):

$$\begin{aligned} \mathbf{C}_{\text{da}} = & \det(\mathbf{C}_Y) \mathbf{R}\mathbf{H} \left( \text{tr}[\mathbf{R}^\top \mathbf{C}_Y^{-1} \mathbf{R} \mathbf{X} \mathbf{X}^\top] \mathbf{R}^\top \mathbf{C}_Y^{-1} \mathbf{R} - \mathbf{R}^\top \mathbf{C}_Y^{-1} \mathbf{R} \mathbf{X} \mathbf{X}^\top \mathbf{R}^\top \mathbf{C}_Y^{-1} \mathbf{R} \right) \mathbf{H}\mathbf{R}^\top \\ & + \det(\mathbf{C}_X) \mathbf{R}\mathbf{H} \left( \text{tr}[\mathbf{C}_X^{-1} \mathbf{R}^\top \mathbf{Y} \mathbf{Y}^\top \mathbf{R}] \mathbf{C}_X^{-1} - \mathbf{C}_X^{-1} \mathbf{R}^\top \mathbf{Y} \mathbf{Y}^\top \mathbf{R} \mathbf{C}_X^{-1} \right) \mathbf{H}\mathbf{R}^\top. \end{aligned} \quad (16)$$

It is this consequence of the main result that is most useful in applications.

In a typical usage of the Procrustes method, the  $\mathbf{X}_i$  may encode the standard points of an object, and therefore be much less noisy than the  $\mathbf{Y}_i$  which are the measurements of their positions after displacement, for instance by some imaging system. Then the second term involving  $\mathbf{C}_X$  in eq.(15) is negligible, and only the first term in eq.(16) remains.

## 2.5 Two prototypical cases of transformation errors

Two special cases of the error covariance are easily computed. If we have uncorrelated and isotropic noise, then  $\mathbf{C}_Y = \sigma_Y^2 \mathbf{I}$  and  $\mathbf{C}_X = \sigma_X^2 \mathbf{I}$ . If that noise is small, we may set  $\mathbf{H} \approx (\text{tr}(\mathbf{X}\mathbf{X}^\top) \mathbf{I} - \mathbf{X}\mathbf{X}^\top)^{-1}$  and  $\mathbf{Y} \approx \mathbf{R}\mathbf{X}$ . We obtain after simplification

$$\mathbf{C}_{\text{da}} \approx (\sigma_X^2 + \sigma_Y^2) \mathbf{R}\mathbf{H}\mathbf{R}^\top. \quad (17)$$

So for small isotropic noise, the covariance of the transformation  $\mathbf{R}$  is like the  $\mathbf{R}$ -transformed harmonized inertia tensor of the point cloud. In this isotropic case, the least squares method produces the optimal estimator.<sup>1</sup>

As a second special case, take an elongated rod-like object with  $\mathbf{H}_r \propto \text{diag}(1, 0, 0)$  (for convenience we placed the principal axis in the  $x$ -direction). Then observe that eq.(15) is of the form  $\mathbf{C}_{\text{da}} = \mathbf{R}\mathbf{H}_r\mathbf{N}\mathbf{H}_r\mathbf{R}^\top$  with  $\mathbf{N}$  a matrix depending on the noise model. With the particular form of  $\mathbf{H}_r$  for the rod, this always simplifies to

$$\mathbf{C}_{\text{da}} = n_{11} \mathbf{R}\mathbf{H}_r\mathbf{R}^\top. \quad (18)$$

(where  $n_{11}$  is the (1,1)-element of  $\mathbf{N}$ ). This clearly shows that whatever the noise, the major transformation error will be in the direction of the  $\mathbf{R}$ -transformed principal axis.

The simulations in Section 3 will confirm these special cases. The bottom line of the analysis will be that understanding the  $\mathbf{H}$  of the point cloud provides good insight in the behavior of the errors.

## 2.6 Translational errors

The error analysis of the translational part of the estimation

$$\mathbf{T} = \bar{\mathbf{Y}} - \mathbf{R}\bar{\mathbf{X}}$$

---

<sup>1</sup>It should therefore coincide with the renormalization estimator of [12]; but since the results in that reference are not given in closed form, this is not obvious.

(this is eq.(3)) propagates the error of the transformation, as well as of the centroid computations. So we obtain to first order

$$d\mathbf{T} = d\bar{\mathbf{Y}} - \mathbf{R}d\bar{\mathbf{X}} - d\mathbf{R}\bar{\mathbf{X}} = d\bar{\mathbf{Y}} - \mathbf{R}d\bar{\mathbf{X}} - (d\mathbf{R}\mathbf{R}^{-1})\mathbf{R}\bar{\mathbf{X}} = d\bar{\mathbf{Y}} - \mathbf{R}d\bar{\mathbf{X}} + (\mathbf{R}\bar{\mathbf{X}}) \times d\mathbf{a}, \quad (19)$$

where  $d\mathbf{a}$  is the axis error vector of the estimated transformation. We introduce  $\bar{\mathbf{r}} \equiv \mathbf{R}\bar{\mathbf{X}}$  and the ‘form factor’  $\mathbf{F} \equiv \mathbf{R}\mathbf{H}\mathbf{R}^\top$ , and can now write

$$d\mathbf{T} = \frac{1}{n} \sum_i (d\mathbf{Y}_i - \mathbf{R}d\mathbf{X}_i) + \sum_i \bar{\mathbf{r}}^\times \mathbf{F}(\mathbf{R}\mathbf{X})_i^\times d\mathbf{Y}_i - \sum_i \bar{\mathbf{r}}^\times \mathbf{F}\mathbf{Y}^\times (\mathbf{R}d\mathbf{X})_i. \quad (20)$$

Then assuming uncorrelated noise, we get the covariance. We express it in terms of matrices using the same techniques as in the previous section, but it is convenient to define some shorthand to see the structure of the result. Let  $\hat{\mathbf{Y}} \equiv \mathbf{F}^{-1}\mathbf{Y}$ ,  $\mathbf{C}_{\hat{\mathbf{Y}}} = \mathbf{F}^{-1}\mathbf{C}_{\mathbf{Y}}\mathbf{F}^{-1}$ ,  $\hat{\mathbf{X}} \equiv \mathbf{F}^{-1}\mathbf{R}\mathbf{X}$ ,  $\mathbf{C}_{\hat{\mathbf{X}}} = \mathbf{F}^{-1}\mathbf{R}\mathbf{C}_{\mathbf{X}}\mathbf{R}^\top\mathbf{F}^{-1}$ , then

$$\begin{aligned} \mathbf{C}_{d\mathbf{T}} &= \frac{1}{n}(\mathbf{C}_{\mathbf{Y}} + \mathbf{R}\mathbf{C}_{\mathbf{X}}\mathbf{R}^\top) \\ &+ \det(\mathbf{F})^2 (\bar{\mathbf{r}}^\top \mathbf{C}_{\hat{\mathbf{Y}}}\bar{\mathbf{r}} \hat{\mathbf{X}}\hat{\mathbf{X}}^\top - \mathbf{C}_{\hat{\mathbf{Y}}}\bar{\mathbf{r}}\bar{\mathbf{r}}^\top \hat{\mathbf{X}}\hat{\mathbf{X}}^\top - \hat{\mathbf{X}}\hat{\mathbf{X}}^\top \bar{\mathbf{r}}\bar{\mathbf{r}}^\top \mathbf{C}_{\hat{\mathbf{Y}}} + \bar{\mathbf{r}}^\top \hat{\mathbf{X}}\hat{\mathbf{X}}^\top \bar{\mathbf{r}} \mathbf{C}_{\hat{\mathbf{Y}}}) \\ &+ \det(\mathbf{F})^2 (\bar{\mathbf{r}}^\top \mathbf{C}_{\hat{\mathbf{X}}}\bar{\mathbf{r}} \hat{\mathbf{Y}}\hat{\mathbf{Y}}^\top - \mathbf{C}_{\hat{\mathbf{X}}}\bar{\mathbf{r}}\bar{\mathbf{r}}^\top \hat{\mathbf{Y}}\hat{\mathbf{Y}}^\top - \hat{\mathbf{Y}}\hat{\mathbf{Y}}^\top \bar{\mathbf{r}}\bar{\mathbf{r}}^\top \mathbf{C}_{\hat{\mathbf{X}}} + \bar{\mathbf{r}}^\top \hat{\mathbf{Y}}\hat{\mathbf{Y}}^\top \bar{\mathbf{r}} \mathbf{C}_{\hat{\mathbf{X}}}). \end{aligned} \quad (21)$$

The expression becomes much simpler when  $\bar{\mathbf{r}} = 0$ , i.e. when the original point set  $\mathbf{X}'$  is centered. In the motivating application of section 1, in which we match observations  $\mathbf{Y}'$  to the standard pose of an object coded in  $\mathbf{X}'$ , this seems simple enough to achieve.

But it is not always possible to maintain such a registration. If occlusion occurs, the set of measured  $\mathbf{Y}$  may not contain the same number of points as the original  $\mathbf{X}$ . Before doing Procrustes we should remove those points, and the remaining set of  $\mathbf{X}$ -points may no longer have a zero centroid. We then have the choice between re-centering and applying Procrustes – which means we have to regauge the outcome if we want to compare it with different estimates from a different position of the camera – or doing Procrustes with a non-centered point cloud. In either case, we will encounter the additional noise term in the translation covariance when merging measurements from different locations (for instance using a Kalman filter).

## 2.7 Perturbation bounds

Our results are a refinement of the perturbation results for the Procrustes problem, as computed in numerical analysis literature. The analyses are often on the  $n$ -dimensional Procrustes problem, for which directional dependence is hard to formulate. Perhaps because of that, the focus is on *perturbation bounds*. The numerical stability of the estimation process is typically also studied.

Some directly applicable results for 3D may be found in [15]. They show that the bound on the transformation error in the 3D orthogonal Procrustes problem is proportional to the condition number

$$\kappa_{\mathbf{X}} = (\xi_2^2 + \xi_3^2)^{-1/2},$$

where  $\xi_i$  is the  $i$ -th ordered singular value of  $\mathbf{X}$ . This  $\kappa_{\mathbf{X}}^2$  is, as [15] remarks, the inverse of

the summed of squared distances to the major axis of the point cloud.

The correspondence to our results is clear. With the observation that  $\mathbf{S}$  is approximately the inertia tensor  $\mathbf{X}\mathbf{X}^\top$  (see Section 2.2), we can relate the singular values  $\xi_i$  of  $\mathbf{X}$  and  $\lambda_i$  of  $\mathbf{S}$  as  $\xi_i^2 \approx \lambda_i$ . The dominant term in the isotropic noise approximation for the covariance in eq.(17) is caused by the highest eigenvalue of  $\mathbf{H}$ , which is  $1/(\lambda_2 + \lambda_3)$ . The main contribution to the standard deviation is therefore proportional to:  $(\lambda_2 + \lambda_3)^{-1/2} \approx \kappa_{\mathbf{X}}$ . So our detailed analysis of the directional effects in errors agrees with the coarser (because non-directional) analysis on perturbation bounds.

In this context, the latter can be seen as bounding the  $\mathbf{H}$  matrix ellipsoid by the smallest sphere surrounding it, specifying only the worst effects that may occur. Since the  $\mathbf{H}$  matrix is a harmonized version of the inertia tensor, this is actually not that bad for isotropic noise if the point clouds are not too elongated. But for the effects of non-isotropic noise (such as that of stereo vision) or cloud elongation, we need the more precise directional description.

### 3 Characteristics of the noise propagation

The equations eq.(16) and eq.(21) are straightforward to compute, but it is much better to get an intuition of the kind of errors that might occur in various situations. We therefore evaluate the consequences of some typical scenarios for objects and types of noise. We remark here that the theoretical results have been compared to simulations of these situations, typically for 1000 trials each, and that the estimated covariances match the predicted covariances as well as can be expected for trials of that size. Even with noise

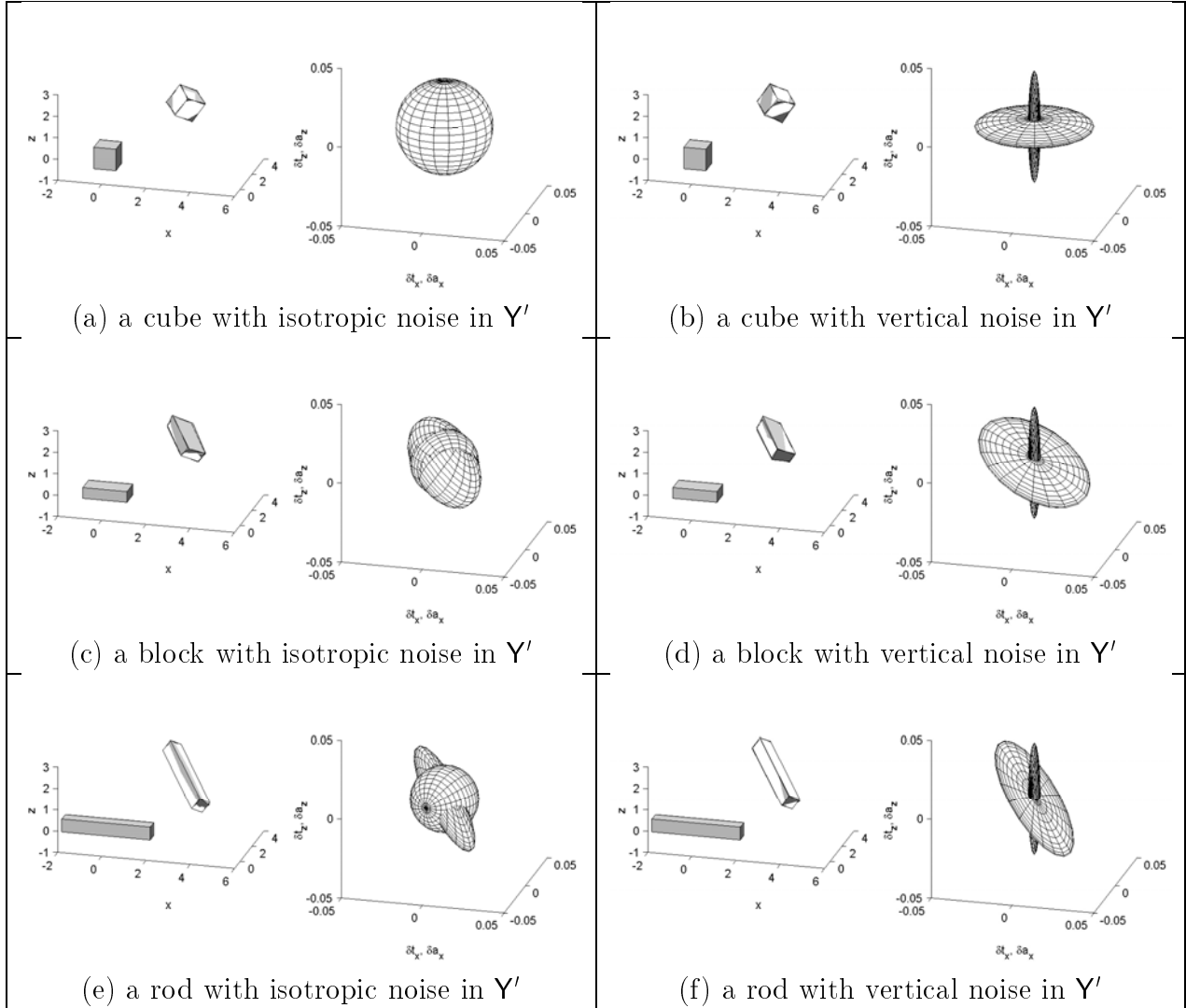


Figure 1: *The effect of shape and kind of  $Y'$ -noise on the translational and rotational errors in the Procrustes method (see text).*

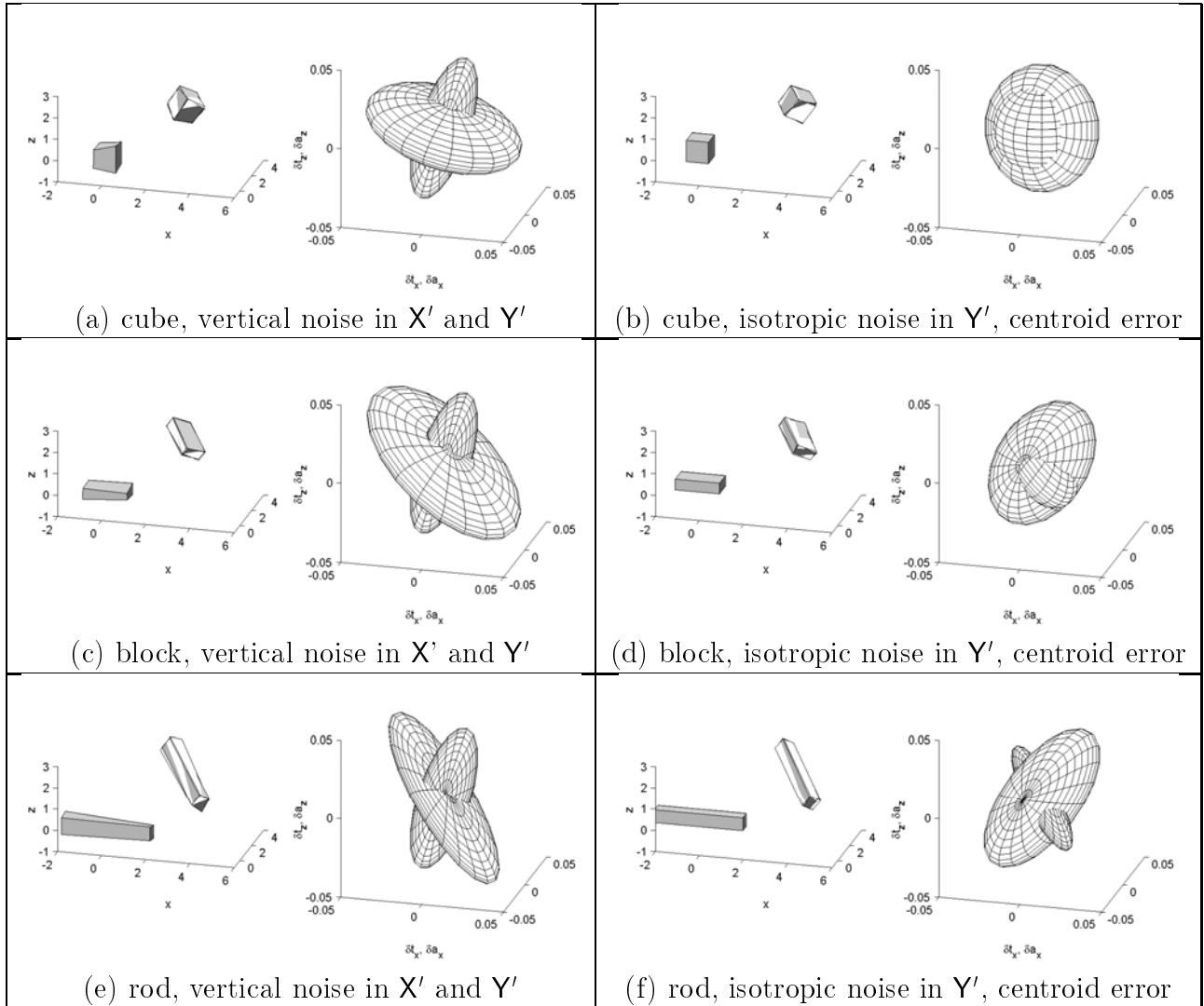


Figure 2: *The effect of noise both before and after the motion (first column) and of a centroid error.*

levels of  $\sigma = 0.5$  (about half the typical size of the objects!), the covariance as predicted by eq.(15) is an accurate estimate of the actual value in the simulations. The error propagation formulas are therefore apparently quite robust for all circumstances.

Since only the second order inertial shape of the cloud matters to our computations, it is sufficient to study a cube, a block and a rod as a sensible representation of the range of shapes. (We saw that a flat block with  $\mathbf{S} = \text{diag}(1, 1, 0)$  has  $\mathbf{H} = \text{diag}(1, 1, 0.5)$  and is therefore intermediate between a block or a sphere, sufficiently similar not to merit special treatment.) For the cube, the singular values of  $\mathbf{S}$  are all similar; for the block we took them in a ratio  $4 : 2 : 1$ , and for the rod  $10 : 1 : 1$ .

For the noise, we are interested in having it present in  $\mathbf{Y}'$  (which mimics the situation where  $\mathbf{X}'$  is a fiducial reference object with little noise), or both  $\mathbf{X}'$  and  $\mathbf{Y}'$  (for a relative orientation estimation). Types of noise of practical interest are isotropic Gaussian noise, and noise in one preferential direction, to approximate e.g. the effect of a stereo vision observation of the point set. All noise will have zero mean, but we will study the effect of a non-centralized point set, leading to extra errors in the translation estimation.

To be specific, in all cases, we do a rotation over -60 degrees around the axis  $(1, -2, 3)^\top$  through the origin, followed by a translation over  $(3, 2, 2)^\top$ . We may offset the  $\mathbf{X}'$  data by  $(-0.5, 1, 0)^\top$  to obtain an impression of the non-centered contributions to the translational noise. The isotropic noise has a standard deviation of 0.1 in all directions, the vertical noise of 0.1 in the vertical direction.

### 3.1 The effect of shape

Let us first study the effect of the object shape, under isotropic noise. The first column of Figure 1 shows the simulations. The subpictures in all figures are composed in the same manner, at the same scales. On the left of each sub-picture, an instantiation in task space is shown: in grey the object data (with the possibly noisy  $X'$  in the lower left near the origin and the noisy  $Y'$  in grey at the top right); in white the reconstruction applying the parameters of the Procrustes estimation to  $X'$ . In all cases, the result is close enough to  $Y'$  (the polyhedral faces of data and reconstruction intersect in all figures). At the right of each sub-picture, we show the error covariance ellipsoids. The rotation ellipsoid is typically oriented with the major axis of the rotated object  $Y'$ , NW-SE; the translational error ellipsoid will tend to be elongated (in many of our examples) and run NE-SW in most of our pictures (a consequence of our chosen parameters). Showing them both in one figure provides relative size and orientation and gives a slightly better appreciation of their spatial nature.

First look at Figure 1(e) for the rod, as a clear example to help you read the pictures. The rotational error is the cigar-shaped ellipse, the translational error is the sphere. The cigar shape indicates that the major error in rotation is in the direction of the major axis of the rod (see the task space picture). This is what eq.(18) predicted. It means that we quite probably have to do a sizable correction of the rotation by an extra little turn around an axis in this direction, over a small angle proportional to the size of the major axis of the cigar. This makes sense, since a rotation over this axis is more sensitive to noise: the

points of  $\mathbf{X}'$  and  $\mathbf{Y}'$  have small distance to the principal axis and their moments relative to it are easily disturbed by noise, severely affecting rotation estimates. In perpendicular directions, the elongation of the object implies that a perturbation of its points affects its estimated orientation much less. And indeed, the cigar is rather thin in those directions, so little adjustment of the estimate is likely.

The covariance ellipsoid of the translational error is a sphere, indicating that the error is isotropic. This is to be expected, since it is merely the noise in the classical estimation of a centroid under isotropic noise (this is eq.(21) for  $\bar{\mathbf{r}} = 0$ ).

In Figure 1(c), we see that the effects are similar for a block. The rotation error is more isotropic than for the rod, since  $\mathbf{H}$  is more isotropic, but it still maintains alignment with  $\mathbf{H}$  as it should according to eq.(17). The translation error is not affected by the shape.

For the cube in Figure 1(a), the rotation error is totally isotropic, i.e. the rotational correction is equally likely to be the same in all directions. In that figure, the translational error is not visible, but is of course identical to that of Figures 1(a),(c).

### 3.2 Non-isotropic noise

In the second column, we have given  $\mathbf{Y}'$  vertical noise instead of isotropic noise, to mimic the effect of the noise of stereovision which tends to be in a predominant direction. The effect on the translational error is clearly that it now lies mainly in that direction; in fact, the shape directly mimics the covariance of the noise in  $\mathbf{Y}'$ . The rotational error is more interesting. Comparing Figure 1(a) and (b), we see that the rotational covariance is very

small in the vertical direction, i.e. almost no rotation around the vertical axis needs to be made to correct the estimate. This makes sense, since with hardly any noise in the non-vertical components, the rotation parameter *around* the vertical axis is almost exactly what it should be. In perpendicular directions, however, the disk-shaped covariance shows that a rather large correction may be required due to the vertical errors in the points.

Similar effects can be observed for the block and rod in Figure 1(d) and (f). For (f), we expect an alignment with the principal axis in agreement with eq.(18). One can characterize the net effect for all cases by saying that the isotropic rotation covariances are flattened in the vertical direction.

### 3.3 Noise before and after the motion

The first column of Figure 2 studies the effect of noise in both  $\mathbf{X}'$  and  $\mathbf{Y}'$ . This is what might happen when the Procrustes method is used to determine a *relative* rotation rather than an *absolute* orientation (relative to a standard pose). We have taken vertical noise in both to mimic the effects of stereo vision. When we compare this column to column 2 of Figure 1, we should correct for the increase of about  $\sqrt{2}$  which we would expect in the magnitude of the noise. We then see that the flattened rotational ellipsoids have thickened, and the cigar-shaped translational ellipsoids have become flattish ellipses. Both are the consequence of the addition of distributions, which corresponds to the convolution of their covariance ellipsoids. For the translation, a pure  $z$ -oriented cigar due to the vertical noise in  $\mathbf{Y}'$  needs to be convolved with the  $\mathbf{R}$ -rotated  $z$ -cigar due to the vertical noise in  $\mathbf{X}'$ . The

result is a flat ellipsoid in the plane of rotation, with a principal axis in the direction ‘ $z$ ’ rotated by half the angle of  $\mathbf{R}$ . This is independent of the shape of the point cloud.

The thickening and tilt of the rotational ellipsoids is due to the same effect: they are the result of convolution of the rotational ellipsoids in column two of Figure 1 with their  $\mathbf{R}$ -rotated versions. This has a large effect for the cube: since its rotational ellipsoid tilts most upon rotation, it thickens most. It has the smallest effect for the rod since eq.(18) shows that its orientation tends not to depend on the noise. And indeed, the rotational ellipsoid in Figure 2(e) is approximately a  $\sqrt{2}$ -scaled version of the one in Figure 1(f)

### 3.4 Incorrect centroid

In column two of Figure 2, we have offset the centroid of  $\mathbf{X}'$  to  $\bar{\mathbf{X}} = (-0.5, 1, 0)^\top$ , while  $\mathbf{Y}'$  has isotropic noise. This offset is rather large, to show the effects. (In practice one would not expect the effect to be more than a fraction of the dimensions of the point cloud, at most about half when optical occlusion occurs in a solid object.)

Equation (19) shows that the extra term in the error is perpendicular to  $\mathbf{R}\bar{\mathbf{X}}$ , and proportional to the magnitude of the projections of the  $\mathbf{d}\mathbf{a}$  contributions on this plane (as well as perpendicular to them). This extra error is added to the uniform distribution of the translation error which we saw in column one of Figure 1. This implies that the distributions are convolved, and we therefore expect to see ellipsoids mostly in the plane perpendicular to  $\mathbf{R}\bar{\mathbf{X}}$ , with an orientation perpendicular to the projections onto it of the rotational ellipsoids for each shape. This is indeed what column two of Figure 2 shows.

There is no effect on the rotational errors, which are the same as Figure 1, column one.

## 4 Conclusion

In deriving the sensitivity of the Procrustes method in estimating orientation and position (or, equivalently, rotation and translation), we were able to get a full understanding of its accuracy. We summarize our findings as follows.

- The exact error propagation formulas are given by eq.(14) (orthogonal transformation) and eq.(19) (translation). Their effects on the covariance under uncorrelated noise are eq.(15) and eq.(20). The matrix forms of those are eq.(16) and eq.(21).
- The transformed harmonized inertia tensor makes its influence felt on all errors and is the clue to understanding the behavior of Procrustes estimation. Only the second order shape of a point cloud is relevant. But the inertia tensor  $\text{diag}(\lambda_1, \lambda_2, \lambda_3)$  itself is *not* what matters – it needs to be harmonized to  $\text{diag}(1/(\lambda_2 + \lambda_3), \text{and cyclic})$ .
- The isotropic noise approximation eq.(17) and the rod shape approximation eq.(18) are sufficient to ‘explain’ most features of the directional noise distribution.
- Noise in one direction, such as typical of stereo vision, can ‘flatten’ the covariances considerably and lead to strong directional effects in the orientation estimation.
- The translational noise is affected by whether or not the point cloud is centered; occlusion may make this a significant contribution in practice.

We re-emphasize that the Procrustes method is only optimal when the noise on the data points is isotropic, identical and independently Gaussian with zero mean. If one's circumstances significantly deviate from this, one should consider the non-linear optimal method of [12].

We are currently developing a Kalman filter for pose estimation, taking into account the special nature of orientation parameters. It computes the optimal combination of estimated poses, based on the covariances of their parameters [2]. We are using the result of the present paper to do 'Kalman on Procrustes', and hope to report on the results in applications soon.

## Acknowledgement

My sincere thanks to valued colleague Nikos Vlassis for his very helpful comments on this paper.

## References

- [1] K.S. Arun, T.S. Huang, S.D. Blostein, *Least-square fitting of Two 3-D Point Sets*, IEEE-PAMI vol.9, no.5, 1987.
- [2] L. Dorst, *Combining orientation estimates*, submitted to Phil. Trans. Royal Soc. London, 2002.

- [3] D.W. Eggert, A. Lorusso, R.B. Fisher, *Estimating 3-D Rigid Body Transformations: A Comparison of Four Major Algorithms*, Machine Vision & Applications, no.9, pp.272-290, 1997.
- [4] J.L. Farrell and J.C. Stülpnagel, *An exact solution of absolute orientation*, Photogrammetria, vol.16, no.1, pp.34-37, 1960.
- [5] B.F. Green, *The orthogonal approximation of an oblique structure in factor analysis*, Psychometrika, vol. 17, pp. 429–440, 1952.
- [6] R.J. Hanson, M.J. Norris, *Analysis of measurements based on the singular value decomposition*, SIAM J. Sci. Stat. Comput., vol.2, no.3, 1981.
- [7] B.K.P. Horn, *Closed Form Solution of Absolute Orientation using Unit Quaternions*, J.Opt.Soc.Am., A, Vol. 4, No. 4, pp. 629–642, April 1987.
- [8] B.K.P. Horn, H. M. Hilden and S. Negahdaripour, *Closed Form Solution of Absolute Orientation using Orthonormal Matrices*, J.Opt.Soc.Am., A, vol. 5, no. 7, pp. 1127–1135, July 1988.
- [9] K. Kanatani, *Analysis of 3-D Rotation Fitting*, IEEE PAMI, vol.16, no.5, 1994.
- [10] F.A. McRobie and J. Lasenby, *Simo-Vu Quoc rods using Clifford Algebra* Int. Journal Num. Methods Eng., 45,377-398, 1999.
- [11] Matlab, The Language of Technical Computing, The Math Works Inc., 1984.

- [12] N. Ohta and K. Kanatani, *Optimal Estimation of Three-Dimensional Rotation and Reliability Evaluation*, IEICE Trans. Information and Systems, vol. E82-D, no.11, 1998, pp.1247–1252.
- [13] R. Sibson, *Studies in the Robustness of Multidimensional Scaling: Procrustes Statistics*, J. Royal Statistical Society, Series B (Methodological), vol.40, no.2, 1978, pp.234–238. Available at [www.jstor.org](http://www.jstor.org).
- [14] I. Söderkvist, *Perturbation Analysis of the Orthogonal Procrustes Problem*, BIT vol 33, 1993, pp.687–694.
- [15] I. Söderkvist and P.-Å. Wedin, *On condition numbers and algorithms for determining a rigid body movement*, BIT vol.34, pp.424–436, 1994.
- [16] P.H. Schönemann, *A generalized solution of the orthogonal Procrustes problem*, Psychometrika, vol.31, pp.1-10, 1966.
- [17] S. Umeyama, *Least-squares Estimation of Transformation Parameters Between Two Point Patterns*, IEEE PAMI, vol.13, no.4, 1991.

## A Derivation of error propagation result

To solve eq.(13) for  $\delta\mathbf{R}$ , note that it and the right hand side are  $3 \times 3$  skew matrices. They can therefore be characterized as cross products with certain vectors. Since both sides are of the same form, let us first derive the following property of  $3 \times 3$  matrices  $A$  and  $B$ :

$$\text{vec}(B^\top A - A^\top B) = \sum_k A_k^\top \times B_k^\top, \quad (22)$$

where  $A_k$ , with  $A$  a matrix, denotes the  $k$ -th column of  $A$ , and we define  $A_k^\top \equiv (A^\top)_k$ . To derive eq.(22), we first observe that a cross product acting after a cross product can be written as a linear mapping:

$$(a \times b) \times c = b(a \cdot c) - a(b \cdot c) = (ba^\top - ab^\top)c \quad (23)$$

so that

$$\text{vec}(ba^\top - ab^\top) = a \times b.$$

By rewriting an anti-symmetric product of matrices in this form we can thus find its characterizing vector.

$$\begin{aligned} (B^\top A - A^\top B)_{ij} &= \sum_k (B^\top)_{ik} (A)_{kj} - (A^\top)_{ik} (B)_{kj} \\ &= \sum_k \left( (B_k^\top)_i (A_k^\top)_j - (A_k^\top)_i (B_k^\top)_j \right) \\ &= \sum_k \left( B_k^\top (A_k^\top)^\top - A_k^\top (B_k^\top)^\top \right)_{ij} \end{aligned}$$

and from comparison with eq.(23), the result of eq.(22) follows.

Using eq.(22), we can determine the characterizing vector of the left hand side of eq.(13).

We rewrite the result using  $a \times (b \times c) = (c^\top a - ca^\top) b$ , which is a variation of eq.(23):

$$\begin{aligned}
\text{vec}(\bar{\mathbf{S}} \delta \mathbf{R} - (\delta \mathbf{R})^\top \bar{\mathbf{S}}) &= \sum_k \delta \mathbf{R}_k^\top \times \bar{\mathbf{S}}_k \\
&= \sum_k (\mathbf{d}\mathbf{a}^\times)_k^\top \times \bar{\mathbf{S}}_k \\
&= \sum_k \bar{\mathbf{S}}_k \times (\mathbf{d}\mathbf{a} \times \mathbf{l}_k) \\
&= \sum_k (\mathbf{l}_k^\top \bar{\mathbf{S}}_k - \mathbf{l}_k \bar{\mathbf{S}}_k^\top) \mathbf{d}\mathbf{a} \\
&= (\mathbf{l} \text{tr}(\bar{\mathbf{S}}) - \bar{\mathbf{S}}) \mathbf{d}\mathbf{a}.
\end{aligned} \tag{24}$$

For the right hand side of eq.(13), we obtain

$$\begin{aligned}
\text{vec}(\bar{\mathbf{S}} \delta \mathbf{A} - (\delta \mathbf{A})^\top \bar{\mathbf{S}}) &= \sum_k (\delta \mathbf{A})_k^\top \times \bar{\mathbf{S}}_k \\
&= \sum_k (\mathbf{A}^{-\top} \mathbf{d}\mathbf{A}^\top)_k \times \bar{\mathbf{S}}_k \\
&= \sum_k (\mathbf{A}^{-\top} \mathbf{X} \mathbf{d}\mathbf{Y}^\top + \mathbf{A}^{-\top} \mathbf{d}\mathbf{X} \mathbf{Y}^\top)_k \times \bar{\mathbf{S}}_k \\
&\stackrel{1}{=} \sum_k (\bar{\mathbf{S}} \mathbf{R} \mathbf{X} \mathbf{d}\mathbf{Y}^\top + \bar{\mathbf{S}} \mathbf{R} \mathbf{d}\mathbf{X} \mathbf{Y}^\top)_k \times \bar{\mathbf{S}}_k \\
&\stackrel{2}{=} \sum_{i=1}^n (\bar{\mathbf{S}} \mathbf{R} \mathbf{X})_i \times (\bar{\mathbf{S}} \mathbf{d}\mathbf{Y})_i + (\bar{\mathbf{S}} \mathbf{R} \mathbf{d}\mathbf{X})_i \times (\bar{\mathbf{S}} \mathbf{Y})_i \\
&\stackrel{3}{=} \det(\bar{\mathbf{S}}) \bar{\mathbf{S}}^{-1} \left( \sum_{i=1}^n (\mathbf{R}\mathbf{X})_i \times \mathbf{d}\mathbf{Y}_i + (\mathbf{R}\mathbf{d}\mathbf{X})_i \times \mathbf{Y}_i \right)
\end{aligned} \tag{25}$$

In making these simplifications we used in step 1 the relationship  $\bar{\mathbf{S}} = \mathbf{R}\mathbf{S}^{-1}\mathbf{R}^\top = \mathbf{R}\mathbf{A}^{-1} = \mathbf{A}^{-\top}\mathbf{R}^\top$ . In step 2, we rewrite the sum over the 3 dimensions of the basis to a sum over

the  $n$  data items. This is easily demonstrated using the anti-symmetric tensor symbol  $\epsilon$  to write the components of the cross product. Let  $\mathbf{B}$  be a  $3 \times 3$  matrix, then:

$$\begin{aligned}
\left( \sum_k (\mathbf{X}\mathbf{Y}^\top)_k \times \mathbf{B}_k \right)_m &= \sum_{ijk} \epsilon_{ijm} (\mathbf{X}\mathbf{Y}^\top)_{ik} \mathbf{B}_{jk} \\
&= \sum_{ijkn} \epsilon_{ijm} \mathbf{X}_{in} \mathbf{Y}_{kn} \mathbf{B}_{jk} \\
&= \sum_{ijn} \epsilon_{ijm} \mathbf{X}_{in} (\mathbf{B}\mathbf{Y})_{jn} \\
&= \left( \sum_n \mathbf{X}_n \times (\mathbf{B}\mathbf{Y})_n \right)_m
\end{aligned}$$

In step 3 above, we used the transformation of a cross product under a linear mapping, which is:

$$\mathbf{B}(\mathbf{x} \times \mathbf{y}) = (\mathbf{B}^{-\top} \mathbf{x}) \times (\mathbf{B}^{-\top} \mathbf{y}) \det(\mathbf{B}) \quad (26)$$

to pull out a factor of  $\det(\bar{\mathbf{S}})\bar{\mathbf{S}}^{-1}$ . Combining eq.(25) with eq.(24), we then get as the solution of eq.(13), to first order:

$$\mathbf{d}\mathbf{a} = \text{vec}(\delta\mathbf{R}) = (\text{tr}(\bar{\mathbf{S}})I - \bar{\mathbf{S}})^{-1} \bar{\mathbf{S}}^{-1} \det(\bar{\mathbf{S}}) \left( \sum_{i=1}^n (\mathbf{R}\mathbf{X})_i \times \mathbf{d}\mathbf{Y}_i + (\mathbf{R}\mathbf{d}\mathbf{X})_i \times \mathbf{Y}_i \right)$$

We can rewrite the factor involving  $\bar{\mathbf{S}} = \mathbf{R}\mathbf{S}^{-1}\mathbf{R}^\top$ , in a manner that is best seen by rewriting  $\mathbf{S}$  using the SVD of  $\mathbf{A}$  as  $\mathbf{V}\mathbf{\Lambda}\mathbf{V}^\top$ , with  $\mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$ :

$$\begin{aligned}
&(\bar{\mathbf{S}}\text{tr}(\bar{\mathbf{S}}) - \bar{\mathbf{S}}^2)^{-1} \det(\bar{\mathbf{S}}) = \\
&= \mathbf{R}(\mathbf{S}^{-1}\text{tr}(\mathbf{S}^{-1}) - \mathbf{S}^{-1}\mathbf{S}^{-1})^{-1} \det(\mathbf{S}^{-1})\mathbf{R}^\top
\end{aligned}$$

$$\begin{aligned}
&= \mathbf{R} \mathbf{V} \left( (\Lambda^{-1} \text{tr}(\Lambda^{-1}) - \Lambda^{-1} \Lambda^{-1}) \det(\Lambda) \right)^{-1} \mathbf{V}^\top \mathbf{R}^\top \\
&= \mathbf{R} \mathbf{V} \left( \text{diag} \left( \frac{1}{\lambda_1} \left( \frac{1}{\lambda_1} + \frac{1}{\lambda_2} + \frac{1}{\lambda_3} \right) - \frac{1}{\lambda_1^2}, \text{ and cyclic } \lambda_1 \lambda_2 \lambda_3 \right) \right)^{-1} \mathbf{V}^\top \mathbf{R}^\top \\
&= \mathbf{R} \mathbf{V} \text{diag}(\lambda_2 + \lambda_3, \lambda_3 + \lambda_1, \lambda_1 + \lambda_2)^{-1} \mathbf{V}^\top \mathbf{R}^\top \\
&= \mathbf{R} (\text{tr}(\mathbf{S}) \mathbf{I} - \mathbf{S})^{-1} \mathbf{R}^\top.
\end{aligned}$$

This proves eq.(14) in the main body of the paper.

## B Derivation of conversion formula

For the conversion of eq.(15) to eq.(16), you need to consider the covariance matrix as a mapping of a vector, and invoke the properties eq.(26) and eq.(23):

$$\begin{aligned}
\left( \sum_i \mathbf{Z}_i^\times \mathbf{C} \mathbf{Z}_i^\times \right) \mathbf{z} &= \sum_i \mathbf{Z}_i \times \mathbf{C} (\mathbf{Z}_i \times \mathbf{z}) \\
&= \sum_i \mathbf{Z}_i \times \left( (\mathbf{C}^{-1} \mathbf{Z}_i) \times (\mathbf{C}^{-1} \mathbf{z}) \right) \det(\mathbf{C}) \\
&= \det(\mathbf{C}) \sum_i \left( \mathbf{C}^{-1} \mathbf{Z}_i (\mathbf{Z}_i \cdot (\mathbf{C}^{-1} \mathbf{z})) - (\mathbf{Z}_i \cdot (\mathbf{C}^{-1} \mathbf{Z}_i)) \mathbf{C}^{-1} \mathbf{z} \right) \\
&= \det(\mathbf{C}) \sum_i \left( \mathbf{C}^{-1} \mathbf{Z}_i \mathbf{Z}_i^\top \mathbf{C}^{-1} - \mathbf{Z}_i^\top \mathbf{C}^{-1} \mathbf{Z}_i \mathbf{C}^{-1} \right) \mathbf{z} \\
&= \det(\mathbf{C}) \left( \mathbf{C}^{-1} \mathbf{Z} \mathbf{Z}^\top \mathbf{C}^{-1} - \text{tr}[\mathbf{Z}^\top \mathbf{C}^{-1} \mathbf{Z}] \mathbf{C}^{-1} \right) \mathbf{z} \\
&= \det(\mathbf{C}) \left( \mathbf{C}^{-1} \mathbf{Z} \mathbf{Z}^\top \mathbf{C}^{-1} - \text{tr}[\mathbf{C}^{-1} \mathbf{Z} \mathbf{Z}^\top] \mathbf{C}^{-1} \right) \mathbf{z}
\end{aligned}$$

We used the cyclic reordering property of the trace to obtain some symmetry and computational gain in the final expression.