ON ESTIMATING ROTATIONS

Wolfgang Förstner
Institut für Photogrammetrie, Universität Bonn
D-53115 Bonn, Nussallee 15, e-mail: wf@ipb.uni-bonn.de

KEY WORDS:

Rotations, representations, direct estimation.

The paper collects tools for estimating rotations. Starting from the classical representations with quaternions concatenation rules for the Rodriguez parameters are given. Direct estimates for mean rotations and for rotations from homologous spatial directions are given. Two robust estimation procedures are given for estimating the rotation matrix of a single camera from observed straight line segments in a legoland scene based on a grouping procedure for line segment and and a clustering procedure on the 3-sphere.

Festschrift für Prof. Dr.-Ing. Heinrich Ebner zum 60. Geburtstag, Herausg.: C. Heipke und H Mayer, Lehrstuhl fr̈ Photogrammetrie und Fernerkundung, TU Mnchen, 1999 .

1 MOTIVATION

Estimating rotations in three dimensions (3D) is a standard task in Photogrammetry. The orientation of a single camera or an image pair contains the three rotation angles as a subset of the parameters. In case of estimating the exterior orientation cameras containing images of the stellar field only rotation parameters appear as unknowns. This special case was the basis for the doctoral thesis of H. Ebner 1968 (Ebner 1968). As rotations only can be represented with nonlinear functions, usually trigonometric but at least quadratic functions, maximum likelihood estimates in general require approximate values. Using a appropriate representation, however, may enable direct solutions which in some cases are optimal estimates.

We want to collect procedures for estimating rotations in 3D which occur within photogrammetric orientation procedures.

Notation: Vectors and matrices are denoted slanted boldface, e. g. $\boldsymbol{r}$ or $\boldsymbol{R}$. The operator $\mathsf{N}(\cdot)$ normalizes a vector to length 1 and a matrix to a rotation matrix. Homogeneous vectors and matrices, which represent the same object when multiplied with an arbitrary scalar $\neq 0$ are denoted with upright boldface letters, e. g. $\mathbf{q}$. Estimates are denoted with a hat, e. g. $\widehat{\boldsymbol{r}}$. We often use the skew matrix

$$\boldsymbol{S}_r \doteq \boldsymbol{S}(\boldsymbol{r}) = \begin{pmatrix} 0 & -r_3 & r_2 \\ r_3 & 0 & -r_1 \\ -r_2 & r_1 & 0 \end{pmatrix}$$

depending on a vector which allows to write the crossproduct as a matrix vector product $\boldsymbol{a} \times \boldsymbol{b} = \boldsymbol{S}_a \boldsymbol{b} = -\boldsymbol{S}_b \boldsymbol{a} = -\boldsymbol{b} \times \boldsymbol{a}$.

2 REPRESENTATIONS OF ROTATIONS

2.1 Parameterization

Rotations in 3D can be represented by 3 independent parameters. Redundant representations with more than three parameters exist and show certain advantages. There are many ways for representing rotations (Rinner 1957, Schut 1958/59, Rinner and Burkhardt 1972, Faugeras 1993). Some of them are given together with their mutual relations (cf. table 1)

- Eulerian angles:

  $$(\alpha, \beta, \gamma)$$

  Their advantage is the simple geometric interpretation, specially in connection with physical devices. The disadvantage is the large number of rotation matrices being generate by the same rotation angles, depending on the selection and sequence of axes and whether they are stable or moving. Moreover, the rotation matrix contains trigonometric expressions in the angles. We will not use this representation.

- Axis-angle representation:

  $$(\boldsymbol{r}, \phi)$$

  Its advantage is the simple interpretation. Its disadvantage is the instability of the axis for small rotations and the redundancy in the representation. Also here the rotation matrix contains trigonometric expressions of the rotation angle.

	$(\boldsymbol{r},\phi)$	$\mathbf{q}=(q_0,\,\boldsymbol{q})$	$\boldsymbol{m}$	$\boldsymbol{u}$
$\boldsymbol{R}$	eq. (1)	eq. (4)	eq. (3)	eq. (5)
small (α,β,γ)	$\boldsymbol{r}\phi$	$2\,\lvert\boldsymbol{q}\rvert/q_0$	$\boldsymbol{m}$	$2\,\boldsymbol{u}$
$(\boldsymbol{r},\phi)$	$(\boldsymbol{r},\phi)$	$\boldsymbol{r}=\mathsf{N}(\boldsymbol{q})$, $\phi=2\,\mathrm{atan2}(\lvert\boldsymbol{q}\rvert,\,q_0)$	$\boldsymbol{r}=\mathsf{N}(\boldsymbol{m})$, $\phi=2\,\mathrm{atan}(\lvert\boldsymbol{m}\rvert/2)$	$\boldsymbol{r}=\mathsf{N}(\boldsymbol{u})$, $\phi=2\,\mathrm{atan}(\lvert\boldsymbol{u}\rvert)$
$\mathbf{q}=(q_0,\,\boldsymbol{q})$	$(\cos\phi/2,\,\boldsymbol{r}\,\sin\phi/2)$	$\mathbf{q}$	$(1,\boldsymbol{m}/2)$	$(1,\boldsymbol{u})$
$\boldsymbol{m}$	$2\,\boldsymbol{r}\tan\phi/2$	$2\,\boldsymbol{q}/q_0$	$\boldsymbol{m}$	$2\,\boldsymbol{u}$
$\boldsymbol{u}$	$\boldsymbol{r}\tan\phi/2$	$\boldsymbol{q}/q_0$	$\boldsymbol{m}/2$	$\boldsymbol{u}$

Table 1: *Relations between representation of rotation parameters. For small rotations the rotation vector* (α,β,γ) *is identical with the Rodrigues vector* $\boldsymbol{m}$*.*

$$\boldsymbol{R}_{r\phi}(\boldsymbol{r},\phi)=\cos\phi\boldsymbol{I}+(1-\cos\phi)\boldsymbol{r}\boldsymbol{r}^{\mathsf{T}}+\sin\phi\boldsymbol{S}_r \quad (1)$$

- Quaternions:

$$\mathbf{q}=(q_0,q_1,q_2,q_3)^{\mathsf{T}}=\begin{pmatrix} q_0 \\ \boldsymbol{q} \end{pmatrix}$$

With the vector part of the quaternion $\boldsymbol{q}=(q_1,q_2,q_3)^{\mathsf{T}}$ They give an algebraic representation of the rotation matrix

$$\boldsymbol{R}_Q(\mathbf{q})=\frac{1}{|\mathbf{q}|^2}\left((q_0^2-\boldsymbol{q}^{\mathsf{T}}\boldsymbol{q})\boldsymbol{I}+2\boldsymbol{q}\boldsymbol{q}^{\mathsf{T}}+2q_0\boldsymbol{S}_q\right) \quad (4)$$

It is non singular thus is uniquely defined (up to a sign). Both being advantages at the cost of a redundant representation with four values. In case the 4-vector $\mathbf{q}$ is normalized to 1, the rotation matrix only contains quadratic terms in the parameters (cf. eq. 2).

- Rodrigues representation (Rodrigues 1840):

$$\boldsymbol{m}=(a,b,c)^{\mathsf{T}}$$

with an algebraic expression of the rotation matrix (cf. eq. 3) closely linked to the quaternion representation.

It is a minimal representation. It has been used in the PAT-programs for the aerial triangulation (Ackermann *et al.* 1970) due to its simplicity. However, it cannot represent rotations of 180^o as can be seen in table 1.

- Skew-matrix representation:

$$\boldsymbol{u}=(u,v,w)^{\mathsf{T}}$$

also leading to an algebraic expression for the rotation matrix:

$$\boldsymbol{R}_s(\boldsymbol{u})=(\boldsymbol{I}-\boldsymbol{S}_u)^{-1}(\boldsymbol{I}+\boldsymbol{S}_u)=(\boldsymbol{I}+\boldsymbol{S}_u)(\boldsymbol{I}-\boldsymbol{S}_u)^{-1} \quad (5)$$

As $\boldsymbol{u}=1/2\boldsymbol{m}$ (cf. table 1) the Rodrigues matrix (3) also can be written as

$$\boldsymbol{R}_r(\boldsymbol{m})=(\boldsymbol{I}-\boldsymbol{S}_m/2)^{-1}(\boldsymbol{I}+\boldsymbol{S}_m/2)$$

Observe, all representations, except the one with Eulerian angles, are linked through the rotation axis and the rotation angle. Given a rotation matrix, these can be derived from the relations

$$\boldsymbol{a}=\begin{pmatrix} r_{32}-r_{23} \\ r_{13}-r_{31} \\ r_{21}-r_{12} \end{pmatrix}=2\sin\phi\,\boldsymbol{r} \qquad tr\boldsymbol{R}=1+2\,\cos\phi$$

derivable from (1) and therefore

$$\boldsymbol{r}=\mathsf{N}(\boldsymbol{a}) \qquad \phi=\mathrm{arctan2}(|\boldsymbol{a}|,\,tr\boldsymbol{R}-1) \quad (6)$$

Also observe, the quaternion representation is the only one which, except for the sign, is unique and shows no singularities.

Other representations such as $\boldsymbol{R}(\boldsymbol{u})=\mathrm{e}^{\boldsymbol{S}_u}$ also rotating around $\boldsymbol{u}$ (Rodrigues 1840, Faugeras 1993) may be useful in real time applications.

2.2 Concatenation

Rotations form the special orthogonal group of matrices, SO_3. thus the concatenation of two matrices $\boldsymbol{R}'$ and $\boldsymbol{R}''$ leads again to a rotation matrix $\boldsymbol{R}=\boldsymbol{R}'\boldsymbol{R}''$ and an inverse exists, known to be the transpose. Now, epresenting the rotation with three angles does not allow a simple concatenation. But, all other representations give rise to a simple concatenation rule, being based on the concatenation of quaternions.

Classical concatenation of two quaternions $\mathbf{q}=(q_0,\boldsymbol{q})$ and $\mathbf{r}=(r_0,\boldsymbol{r})$ results from Hamilton's quaternion algebra, where

$$\mathbf{p}=\mathbf{q}\mathbf{r}$$

is defined by

$$p_0=q_0r_0-\boldsymbol{q}\cdot\boldsymbol{r} \qquad \boldsymbol{p}=r\boldsymbol{q}+q\boldsymbol{r}+\boldsymbol{q}\times\boldsymbol{r}$$

$$\boldsymbol{R}_Q(\mathbf{q}\,|\,|\mathbf{q}|=1) = \begin{pmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2\,(q_1q_2-q_0q_3) & 2\,(q_1q_3+q_0q_2) \\ 2\,(q_2q_1+q_0q_3) & q_0^2-q_1^2+q_2^2-q_3^2 & 2\,(q_2q_3-q_0q_1) \\ 2\,(q_3q_1-q_0q_2) & 2\,(q_3q_2+q_0q_1) & q_0^2-q_1^2-q_2^2+q_3^2 \end{pmatrix} \qquad (2)$$

$$\boldsymbol{R}_r(a,b,c) = \frac{1}{4+a^2+b^2+c^2} \begin{pmatrix} 4+a^2-b^2-c^2 & 2ab-4c & 2ac+4b \\ 2ab+4c & 4-a^2+b^2-c^2 & 2bc-4a \\ 2ac-4b & 2bc+4a & 4-a^2-b^2+c^2 \end{pmatrix} \qquad (3)$$

As the Rodrigues and the skew-matrix representation are special cases of the quaternion representation they also can be concatenated easily. For the Rodrigues representation we obtain for two rotations (a', b', c') and (a'', b'', c'') the combined rotation (a, b, c)

$$a = \frac{2}{D}(2\,(a' + a'') + b'c'' - b''c') \qquad (7)$$

$$b = \frac{2}{D}(2\,(b' + b'') + c'a'' - c''a') \qquad (8)$$

$$c = \frac{2}{D}(2\,(c' + c'') + a'b'' - a''b') \qquad (9)$$

with

$$D = 4 - a'a'' - b'b'' - c'c''. \qquad (10)$$

For the skew-matrix representation we obtain for two rotations (u', v', w') and (u'', v'', w'') the combined rotation (a, b, c)

$$u = \frac{1}{D}((u' + u'') + v'w'' - v''w') \qquad (11)$$

$$v = \frac{1}{D}((v' + v'') + w'u'' - w''u') \qquad (12)$$

$$w = \frac{1}{D}((w' + w'') + u'v'' - u''v') \qquad (13)$$

with

$$D = 1 - u'u'' - v'v'' - v'v''. \qquad (14)$$

Concatenation in the axis-angle represenations can go via the Rodrigues-representation.

2.3 The 3-Sphere and finite Rotation Groups

As rotations in 3D can be represented as unit quaternions, thus 4-vectors of length 1, they cover the complete three dimensional space of the 3D unit sphere S_3 in $\mathbb{R}^4$, short the 3-sphere.

This interpretation of the 3D-rotations has an analogy when representing rotations in 2D and directions in 3D. In 2D all *rotations in the plane* cover the unit circle, the S_1 in $\mathbb{R}^2$, the 1-sphere, and all *directions in 3D* to cover the unit sphere S_2 in $\mathbb{R}^3$, the 2-sphere.

If one needs approximate values for 3D-rotations one could select a finite number of rotations which are equally spaced in the 3-sphere. Their voronoi tessellation would then evenly fill the complete 3D-space of the 3-sphere. On the 2-sphere S_2 this would lead to the 5 Platonic polyhedra, the tetrahedron, the hexahedron, the octahedron, the pentagon dodecahedron and the icosahedron. E. g. the octahedron would suggest the 6 directions up, down, right, left, front and back to be crude approximate values for directions on the sphere. The voronoi cells of these 6 points then are 6 quadrangles with great circles as sides and 120^o angles in the corners. They form a regular tessellation of the 2-sphere S_2.

Ebner (Ebner 1968) used the hexahedron and the dodecahedron tessellation, among others, for simulating equally spaced stars for the accuracy analysis in his thesis.

Similar tessellations exist on the 3-sphere S_3.

They form finite rotation groups (Coxeter 1963), i. e. two centers with quaternions $\mathbf{q}_i$ and $\mathbf{q}_j$, representing rotations, after multiplication, thus concatenation of the rotations, lead to an element of the same group. Examples are the rotation group of the tetrahedron with 12 elements, the rotation group of the octahedron with 24 elements and the rotation group of the icosahedron with 60 elements. Horn (Horn 1990) suggested to use the elements of the octahedron group as approximate values for the rotation matrix in an iterative scheme for relative orientation.

We will need the octahedron group in the procedure for estimation the rotation matrix from straight lines in a legoland image. The problem there is, that the numbering and the direction of the three axis x, y and z is not known. Thus one obtains the rotation not uniquely. It is uncertain up to all rotations a rectangular tripod may go through to fit into a rectangular coordinate system: these consist of the 8 octants and 3 orientation in each octant, thus 24 possible rotations. As $\mathbf{q}$ and $-\mathbf{q}$ represent the same rotation we come up with the 48 elements being the centers of this tessellation of the 3-sphere.

2.4 Choice of Representation

It often is difficult to choose the appropriate representation. There appear to be a few rules which might guide this choice:

- In case rotations need to be linked to a physical device or need to be interpreted by a human, choose a representation with angles, e. g. with Eulerian angles or the axis-angle representation.

- In case rotations are only used computer internal, choose algebraic representations, such as quaternions or the Rodrigues representation, or just the rotation matrix itself. This seemed to be the reason to choose the Rodrigues representation for estimation and was the motivation to use as *exchange format for orientation parameters the rotation matrix not the rotation angles*in the PAT-programs (Ackermann *et al.* 1970). This simplifies the documentation of the orientation parameters.

- In case the estimated rotations are small or you have good approximate values, use any suitable representation with the minimal set of three parameters, e. g. Eulerian angles or the Rodrigues' representation. This avoids redundant representation and the necessity to use constraints in estimation.

- In case the rotations may be arbitrary especially close to 180^o use a redundant representation, e. g. quaternions or the complete rotation matrix.

3 ESTIMATING ROTATIONS

3.1 Orthonormalization of a Matrix

Assume we have an approximation $\boldsymbol{A}$ for a rotation matrix. The rotation matrix which is closest to $\boldsymbol{A}$ may be defined to minimize

$$||\boldsymbol{R} - \boldsymbol{A}|| \rightarrow \min$$

As $||\boldsymbol{R} - \boldsymbol{A}||^2 = tr(\boldsymbol{R} - \boldsymbol{A})(\boldsymbol{R}^{\mathsf{T}} - \boldsymbol{A}^{\mathsf{T}}) = tr\boldsymbol{R}\boldsymbol{R}^{\mathsf{T}} - tr(\boldsymbol{A}\boldsymbol{R}^{\mathsf{T}} + \boldsymbol{R}\boldsymbol{A}^{\mathsf{T}}) + tr(\boldsymbol{A}\boldsymbol{A}^{\mathsf{T}})$ minimizing $||\boldsymbol{R} - \boldsymbol{A}||$ is equivalent to require

$$tr\boldsymbol{R}\boldsymbol{A}^{\mathsf{T}} \rightarrow \max$$

The optimal rotation matrix approximating a given matrix $\boldsymbol{A}$ can be determined from the singular value decomposition

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^{\mathsf{T}}$$

and is given by

$$\widehat{\boldsymbol{R}} = \mathsf{N}(\boldsymbol{A}) \doteq \boldsymbol{U}\boldsymbol{V}^{\mathsf{T}} \tag{15}$$

Proof: (Arun *et al.* 1987)

3.2 Averaging Rotations

Assume we have given several rotation matrices $\boldsymbol{R}_i, i = 1, ..., I$ and need to determine an average rotation matrix $\widehat{\boldsymbol{R}}$. This can be achieved in two ways.

3.2.1 Averaging Rotation Matrices The average rotation matrix can be determined by simple averaging and normalization:

$$\widehat{\boldsymbol{R}} = \mathsf{N}\left(\frac{1}{I}\sum_{i=1}^{I}\boldsymbol{R}_i\right)$$

This is statistically meaningful, as in case the rotation matrices are close to each other the variances of the corresponding elements are identical, i. e. $\sigma^2_{r_{i'jk}} = \sigma^2_{r_{i''jk}}$ which motivates simple averaging.

3.2.2 Averaging Quaternions As second method first determines the unit quaternions $\mathbf{q}_i$, with $|\mathbf{q}_i| = 1$ using the axis-angle representation and (6), averages those

$$\overline{\mathbf{q}} = \frac{1}{I}\sum_{i=1}^{I}\mathbf{q}_i$$

to obtain the unit quaternion

$$\widehat{\mathbf{q}} = \mathsf{N}(\overline{\mathbf{q}})$$

and determines the corresponding rotation matrix from (4).

This estimate of the mean is identical to the *least squares estimate* for equally accurate orientation angles under the constraint $\mathsf{N}(\widehat{\mathbf{q}}) = 1$.

In this case one can estimate the variance of the rotation angles using the deviation of the average $\overline{\mathbf{q}}$ of the quaternions from 1. We obtain for small angles

$$\widehat{\sigma}^2_{\phi} = \frac{8\,I}{(I-3)}\,(1 - |\overline{\mathbf{q}}|)$$

Proof: Let the mean normalized rotation be $\widehat{\mathbf{q}}$ with $|\widehat{\mathbf{q}}| = 1$. Then the individual rotations are $\mathbf{q}_i = \widehat{\mathbf{q}}\,d\mathbf{q}_i$ with small rotations $d\mathbf{q}_i = (\cos(d\phi/2), d\boldsymbol{q}_i) \approx (1 - d\phi_i^2/8, d\boldsymbol{q}_i)$. Then the mean of the $\mathbf{q}_i$ is

$$\overline{\mathbf{q}} = \widehat{\mathbf{q}}\;(1 - \frac{1}{8I}\sum_{i=1}^{I} d\phi_i^2, \mathbf{0})$$

Now $|\overline{\mathbf{q}}| = (1 - \frac{1}{8I}\sum_{i=1}^{I} d\phi_i^2)$. With $\widehat{\sigma}^2_{\phi} = \sum_{i=1}^{I} d\phi_i^2/(I-3)$ the proof is complete.

3.3 Rotation from Homologous Points on the 2-Sphere

Assume I homologous points, i. e. pairs of normalized direction vectors $(\boldsymbol{x}_i, \boldsymbol{y}_i), i = 1, ..., I$ and on the 2-sphere be given, which are assumed to be related by

$$\boldsymbol{y}_i = \boldsymbol{R}\boldsymbol{x}_i \qquad \forall i \tag{16}$$

The directions are assumed to circular error ellipses but may have different weights w_i.

This problem also occurs in absolute orientation where the estimation of the translation, the scale difference and the rotation can be separated (Sanso 1973, Arun *et al.* 1987): The translation can be determined from the centers of gravity of the two point sets. The squared scale difference can be determined from the ratio of the sum of squares of the distances of the points to the centers of gravity. Then only the determination of the rotation matrix is left and eq. (16) can be interpreted as model for the remaining rotation between the centered point sets, thus $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ having not length 1. However, this only has influence on the weighting.

Minimizing

$$\boldsymbol{\Phi}' = \sum_{i=1}^{I} |\boldsymbol{y}_i - \boldsymbol{R}\boldsymbol{x}_i|^2 w_i$$

is identical to maximizing

$$\boldsymbol{\Phi} = \sum_{i=1}^{I} \boldsymbol{y}_i{}^{\mathsf{T}} \boldsymbol{R}\boldsymbol{x}_i w_i \tag{17}$$

with respect to the rotation matrix, thus under the condition $\boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$.

There are two classical methods, one with a singular value decomposition, one with quaternions.

3.3.1 Solution with Singular Value Decomposition The function (17) to be maximized can also be written as

$$\boldsymbol{\Phi} = tr(\boldsymbol{R}\boldsymbol{H}^{\mathsf{T}}) \rightarrow \max \tag{18}$$

with the matrix

$$\boldsymbol{H} = \sum_{i=1}^{I} \boldsymbol{y}_i \boldsymbol{x}_i w_i{}^{\mathsf{T}}$$

Thus with (15) we obtain

$$\widehat{\boldsymbol{R}} = \mathsf{N}(\boldsymbol{H})$$

(cf. (Arun *et al.* 1987)).

3.3.2 Solution with Quaternions In case the rotation matrix is represented with a unit quaternion $\mathbf{q}$, thus $\mathbf{q}^{\mathsf{T}}\mathbf{q} = 1$, the rotation matrix contains purely quadratic expressions in the 4 elements of $\mathbf{q}$ (cf. eq. (4) and (18)). We therefore can write (17) as a quadratic form

$$c = \mathbf{q}^{\mathsf{T}} \boldsymbol{M} \mathbf{q}$$

which has to be maximized with respect to the condition $\mathbf{q}^{\mathsf{T}}\mathbf{q} = 1$. Thus the optimal rotation can be derived from the unit eigen vector corresponding to the largest eigenvalue of $\boldsymbol{M}$ as unit quaternion (Sanso 1973, Faugeras and Hebert 1983, Horn 1987).

3.3.3 Solution with Skew Matrices The third solution uses the skew-matrix representation. With the rotation vector $\boldsymbol{u}$ the original problem $\boldsymbol{y}_i = \boldsymbol{R}\boldsymbol{x}_i$ can be written as

$$(\boldsymbol{I} - \boldsymbol{S}_u)\boldsymbol{y}_i = (\boldsymbol{I} + \boldsymbol{S}_u)\boldsymbol{x}_i \qquad \forall i$$

which after using $\boldsymbol{S}_a\boldsymbol{b} = -\boldsymbol{S}_b\boldsymbol{a}$ yields

$$(\boldsymbol{S}_{x_i} + \boldsymbol{S}_{y_i})\boldsymbol{u} = \boldsymbol{x}_i - \boldsymbol{y}_i \qquad \forall i \tag{19}$$

Collecting all sums of skew matrices in the $3I \times 3$ matrix $\boldsymbol{A} = \{\boldsymbol{S}_{x_i} + \boldsymbol{S}_{y_i}\}$ and all differences of directions in the $3\boldsymbol{I}$-vector $\boldsymbol{b} = \{\boldsymbol{x}_i - \boldsymbol{y}_i\}$ we obtain the least squares solution

$$\widehat{\boldsymbol{u}} = (\boldsymbol{A}^{\mathsf{T}}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathsf{T}}\boldsymbol{b}$$

which obviously is not optimal, due to the random errors in $\boldsymbol{A}$. But it may be used as a good approximation.

The individual equations in eq. (19) cannot be solved for $\boldsymbol{u}$ as the sum of two skew matrices is skew and therefore singular, thus contains linear dependent equations. In contrast to the procedure in (Brandstätter 1991) for deriving the rotation matrix in relative orientation, also based on the skew-matrix representation, the procedure above does not select certain equations for obtaining a solution.

4 ROTATION MATRIX FROM STRAIGHT LINES OF A LEGOLAND SCENE

Assume one image of a scene is given which dominantly contains three sets of parallel lines which are mutually orthogonal, as e. g. images of buildings. Such scenes are called legoland scenes. The three sets of parallel lines of a legoland scene in general map into three bundles of lines intersecting at vanishing points. The directions from the projection center to the three vanishing points are parallel to the lines in 3D, thus are mutually orthogonal. Therefore one can determine the position of the projection center, i. e. the main parameters of the interior orientation and the rotation matrix from one image using a spatial resection. However, clustering the lines into the three groups is a task of its own, which may be linked to the determination of the vanishing points of the rotation matrix.

We present three algorithms:

1. Determine the interior orientation and the rotation matrix from three vanishing points.
2. Determine the rotation matrix from noisy line segments using the geometry of the vanishing points, provided the interior orientation is given.
3. Determine the rotation matrix from noisy line segments using the quaternion representation, also provided the interior orientation is given.

4.1 Rotation Matrix and Interior Orientation from Vanishing Points

Assume the three vanishing points $V_i, i = 1, 2, 3$ of three mutually orthogonal sets of straight lines is given. Their coordinates are $\boldsymbol{p}_i = (x_i, y_i)^{\mathsf{T}}$.

We first want to determine the rotation matrix provided the interior orientation is given and then give a procedure for determining the three main parameters of the interior orientation.

4.1.1 Rotations Matrix from Vanishing Points

In case the interior orientation is known, especially the principle point is (x_o, y_o) and the principle distance is c [1], we just need to determine the normalized directions from the projection center to the vanishing points

$$\boldsymbol{m}_i = \mathsf{N}\begin{pmatrix} x_i - x_o \\ y_i - y_o \\ c \end{pmatrix}$$

and collect them in the matrix

$$\boldsymbol{D} = (\boldsymbol{m}_1, \boldsymbol{m}_2, \boldsymbol{m}_3)$$

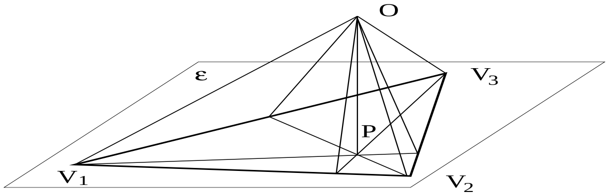
which due to measuring errors is close to the rotation matrix. Normalization

$$\widehat{\boldsymbol{R}} = \mathsf{N}(\boldsymbol{D})$$

yields a rotation matrix.

Instead of the normalization one could also impose the orthogonality constraints in an adjustment procedure taking the precision of the vanishing points into account.

Figure 1: *shows the geometry of the vanishing points V_i in an image with principle point P and projection center O. The three lines OV_i and the three planes OV_iV_j are mutually orthogonal. Thus also the lines V_iP are orthogonal to $V_{i-1}V_{i+1}$, with cyclical indices.*



4.1.2 Interior Orientation from Vanishing Points

In case the interior orientation, namely the principle point $\boldsymbol{p} = (x_o, y_o)$ and the principle distance c are not known, we first determine them from the vanishing points V_i. This can be interpreted as determining a spatial resection, given the three mutually orthogonal directions to the three given vanishing points. The special geometry allows to derive simple expressions.

The principle point is the intersection of the lines V_iP perpendicular to the sides $V_{i-1}V_{i+1}$ of the triangle of the vanishing points and passing through the vanishing points (fig. (1)).

The coordinates (x_o, y_o) of the principle point P can be derived from the intersection of two lines V_iP and are given by (Förstner 1999)

$$x_o = -\frac{\boldsymbol{c}\cdot d\boldsymbol{y}}{\boldsymbol{x}\cdot d\boldsymbol{y}} \qquad y_o = \frac{\boldsymbol{c}\cdot d\boldsymbol{x}}{\boldsymbol{y}\cdot d\boldsymbol{x}}$$

where $\boldsymbol{x} = (x_1, x_2, x_3)^{\mathsf{T}}$ and $\boldsymbol{y} = (y_1, y_2, y_3)^{\mathsf{T}}$ contain the coordinates,

$$d\boldsymbol{x} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \times \boldsymbol{x} = \begin{pmatrix} x_3 - x_2 \\ x_1 - x_3 \\ x_2 - x_1 \end{pmatrix}$$

and

$$d\boldsymbol{y} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \times \boldsymbol{y} = \begin{pmatrix} y_3 - y_2 \\ y_1 - y_3 \\ y_2 - y_1 \end{pmatrix}$$

contain the coordinate differences and

$$\boldsymbol{c} = \begin{pmatrix} \boldsymbol{p}_2\cdot\boldsymbol{p}_3 \\ \boldsymbol{p}_3\cdot\boldsymbol{p}_1 \\ \boldsymbol{p}_1\cdot\boldsymbol{p}_2 \end{pmatrix} = \begin{pmatrix} x_2\,x_3 + y_2\,y_3 \\ x_3\,x_1 + y_3\,y_1 \\ x_1\,x_2 + y_1\,y_2 \end{pmatrix}$$

contains the three dot products of the 2D-point vectors. More explicitly one obtains

$$x_o = \frac{-(y_3 - y_2)\boldsymbol{p}_2\cdot\boldsymbol{p}_3 - (y_1 - y_3)\boldsymbol{p}_1\cdot\boldsymbol{p}_3 - (y_2 - y_1)\boldsymbol{p}_2\cdot\boldsymbol{p}_1}{(y_3 - y_2)x_1 + (y_1 - y_3)x_2 + (y_2 - y_1)x_3}$$

$$y_o = \frac{+(x_3 - x_2)\boldsymbol{p}_2\cdot\boldsymbol{p}_3 + (x_1 - x_3)\boldsymbol{p}_1\cdot\boldsymbol{p}_3 + (x_2 - x_1)\boldsymbol{p}_2\cdot\boldsymbol{p}_1}{(x_3 - x_2)y_1 + (x_1 - x_3)y_2 + (x_2 - x_1)y_3}$$

[1] We choose the sign of c such that the equation $z = c$ represents the image plane in the coordinate system of the camera.

the denominators being equal to double the area of the triangle of the vanishing points.

The principle distance c is given by

$$c = \sqrt{-(\boldsymbol{p}_i - \boldsymbol{p})\cdot(\boldsymbol{p}_j - \boldsymbol{p})}$$

for all $i \neq j$ resulting from the orthogonality of the three rays OV_i.

4.2 Rotation Matrix by Grouping Parallel Straight Lines

Assume only a set of straight line segments is given and the interior orientation to be known. The task is to determine the rotation matrix.

The idea of the following procedure is to group line segments belonging to the same group of parallel lines and exploit their mutual orthogonality. It has been realized and empirically investigated (Wydera 1995).

Formally, let I straight line segments $l_i, i = 1, ..., I$ be given. They are assumed to belong to 2, 3 or 4 classes, namely, $\mathcal{C}_0$, $\mathcal{C}_1$ and possibly $\mathcal{C}_2$ and $\mathcal{C}_3$, where $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_3$ represent sets of lines which are parallel in object space and mutually orthogonal, and $\mathcal{C}_0$ is a set of noisy background edges. The principle point (x_o, y_o) and the principle distance c of the camera, modeled as pinhole camera, are given. The task is to classify the given line segments, determine the rotation matrix and thereby check the validity of the interior orientation.

The line segments l_i are given by their start and end points, P_{s_i} (x_{s_i}, y_{s_i}) and $P_{e_i}(x_{e_i}, y_{e_i})$

For the geometric analysis we represent image points[2] $P_j(\boldsymbol{p}_j)$ and image lines $l(\boldsymbol{n})$ as elements of the projective plane $\mathbb{P}^2$, visualized by the unit sphere S^2 embedded in $\mathbb{R}^3$.

We have for points $P_j(x_j, y_j, c)$ and line segments with starting and end point $p_s^T = \mathsf{N}(x_s, y_s, c)^T$ and $p_e^T = \mathsf{N}(x_e, y_e, c)^T$

$$P_j(\boldsymbol{p}_j) : \boldsymbol{p}_j = \mathsf{N}\begin{pmatrix} x_j \\ y_j \\ c \end{pmatrix}, \quad l(\boldsymbol{n}) : \boldsymbol{n} = \mathsf{N}(\boldsymbol{p}_s \times \boldsymbol{p}_e)\,. \tag{20}$$

Thus points are represented by their direction from the origin O, the projection center, to the image point P, and lines are represented by the normal of the projecting plane, passing through the line and the projection center.

Figure 2: *shows left edge segments with a vanishing point in the middle of the image plane and right the distribution of the normals on the 2-sphere, from (Wydera 1995).*

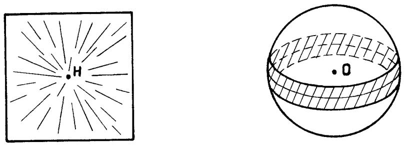
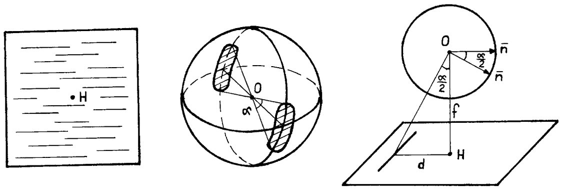


Figure 3: *shows left edge segments with a vanishing at infinity, middle the distribution of the normals on the 2-sphere and right the relation between the image size* $2d$*, the principle distance, here* f*, and the angular length* α *of the cluster, from (Wydera 1995).*



In the following we assume the I normals $\boldsymbol{n}_i^T = (n_{x_i}, n_{y_i}, n_{z_i})$ to be the only available data.

Observe the principle point and the principle distance to be necessary in order the vectors $\boldsymbol{n}_i$ actually to represent the normals of the projecting planes in a Euclidean space.

4.2.1 The Distribution of the Given Data on S^2

We now discuss the distribution of the given normals $\boldsymbol{n}_i$ on the 2-sphere, being the basis for the grouping process:

1. The normals of the sets $\mathcal{C}_1$, $\mathcal{C}_2$,and $\mathcal{C}_3$ lie on 3 great circles ζ_1, ζ_2 and ζ_3 which are mutually orthogonal. The poles of theses circles are identical with their intersection points and in expectation represent the 3 vanishing points $V_1(\boldsymbol{p}_1)$, $V_2(\boldsymbol{p}_2)$, and $V_3(\boldsymbol{p}_3)$. The normalized directions $\boldsymbol{p}_i$ can be collected in a 3×3-matrix being the sought rotation matrix (cf. above)

   $$\boldsymbol{R} = (\boldsymbol{p}_1\ \boldsymbol{p}_2\ \boldsymbol{p}_3) \tag{21}$$

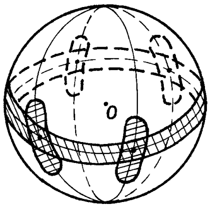
   linking the object and the camera frame.

   The deviations from the circles ζ_c represent the uncertainty of the directions of the image segments towards the vanishing point in concern.

   Because the image is of limited size only parts of the great circles are populated in general: a vanishing point in the center of the image may lead

[2] not to be confused with the principle point P.

Figure 4: *shows the combined distribution of all edge normals, assuming no noise, from (Wydera 1995).*



to a fully populated circle (cf. fig 2), whereas a vanishing point at infinity leads to a great circle which is populated only partly (cf. fig. 3, precisely $\alpha = 2\arctan d/c$ where $|d|$ is is the maximum coordinate.

The combined distribution is shown in fig. 2).

2. The normals of the noisy background edges $\mathcal{C}_0$ are more or less evenly spread over S^2.

The actual distribution will be rotated arbitrarily.

4.2.2 The procedure The task now is equivalent to identifying the three great circles.

This is done in four steps:

1. *1st vanishing point*: Find the first of the three vanishing points by selecting a line segment and confirming it to lie on one, say ζ_1 of the three great circles. The orientation of this great circle fixes 2 of the 3 unknown parameters. Its pole is the first vanishing point V_1. Image edges close to that great circle are collected in $\mathcal{C}_1$.

2. Find one of the remaining great circles, say ζ_2, fixing the second vanishing point V_2. Infer the third vanishing point V_3 using the orthogonality constraint. Image edges close to these great circles are collected in $\mathcal{C}_2$, and $\mathcal{C}_2$ resp. Not yet classified edges form the set $\mathcal{C}_0$.

3. Determine the optimal position of the vanishing points by using all edges in the corresponding class and by taking the uncertainty of the edges into account, which mainly depends on the length of the edges. This also will yield covariance matrices for these points, representing their uncertainty

4. Determine an optimal estimate for the rotation matrix by exploiting the orthogonality constraint. Depending on the number of detected vanishing points also the interior orientation may be updated in a Bayesian manner.

The next sections discuss the first two steps in more detail.

4.2.3 Finding the First Vanishing Point Finding the first vanishing point is the crucial part and performed in the following manner:

1. Select a line segment which is likely to point to a vanishing point with normal, say $\boldsymbol{r}$.

2. Project the normals of the other segments onto the plane π_r orthogonal to $\boldsymbol{r}$:

$$\boldsymbol{n}_i' = \boldsymbol{R}_r \boldsymbol{P}_r \boldsymbol{n}_i \tag{22}$$

where the projection matrix

$$\boldsymbol{P}_r = \boldsymbol{I} - \boldsymbol{r}\boldsymbol{r}^T \tag{23}$$

performs the projection and, using $\boldsymbol{z} = (0,0,1)^\mathsf{T}$ the rotation matrix

$$\boldsymbol{R}_r = \begin{pmatrix} \mathsf{N}(\boldsymbol{z} \times \boldsymbol{r})^T \\ \mathsf{N}[(\boldsymbol{z} \times \boldsymbol{r}) \times \boldsymbol{r}]^T \\ \boldsymbol{r}^T \end{pmatrix} \tag{24}$$

rotates the plane π_r in such a way that the z-values of the projected vectors $\boldsymbol{n}_i'$ are zero. This leads to the combined projection, replacing (22)

$$\boldsymbol{n}_i' = \begin{pmatrix} \mathsf{N}(\boldsymbol{z} \times \boldsymbol{r})^T \\ \mathsf{N}[(\boldsymbol{z} \times \boldsymbol{r}) \times \boldsymbol{r}]^T \\ \boldsymbol{0}^T \end{pmatrix} \boldsymbol{n}_i \tag{25}$$

Practically only the first two elements of the $\boldsymbol{n}_i'$ s are used.

3. Select all those $\boldsymbol{n}_i'$ which lie within a distance t_1 from the origin $(0,0)$ and normalize

$$\overline{\boldsymbol{n}_i'} = \mathsf{N}(\boldsymbol{n}_i') \tag{26}$$

4. Find the maximum of the empirical circular distribution of the angles ϕ, taking into account the π-periodicity of the distribution. This can be achieved in linear time using the empirical circular histogram of the directions $\phi \bmod \pi$.

   Alternatively one could search for the shortest interval on the unit circle, containing a specified percentage of the directions $\phi \bmod \pi$. This requires $O(L \log L)$ operations. The empirical investigations, however, did not confirm the higher complexity empirically.

5. If step 4 succeeded, the direction $\boldsymbol{n}\text{max}(\phi\text{max})$ represents an estimate for the first vanishing point in the rotated plane π_r. Rotating back with

$$\widehat{\boldsymbol{p}}_1^{(0)} = \boldsymbol{n}\text{max} = \boldsymbol{R}_r^T \cdot (\cos\phi\text{max} \;\; \sin\phi\text{max} \;\; 0)^T \tag{27}$$

yields an approximate direction to V_1. If step 4 did not succeed, go back to step 1 and select another line segment.

The rational behind this procedure is

- It is possible to select a line segment which actually belongs to one of the three sets of 3D-lines with high probability, e. g. when selecting the longest line segment in the image.
- It is unlikely that the selected line segment passes through *two* vanishing points, e. g. forming the horizon. This would cause the circular distribution to have two peaks namely those of the two vanishing points in concern.

Therefore it is likely that the circular distribution has only one peak.

Violation of rational 1, i. e. in case the system selects a background edge, no pronounced peak would occur, which can be detected by analyzing the histogram. Violation of rationale 2 would do no harm, as both peaks can be used as result of this first step.

4.2.4 Finding the Second and the Third Vanishing Point Based on the first vanishing point $V_1(\boldsymbol{p}_1)$ the second and the third vanishing points again are found by cluster analysis.

1. Project the normals of the remaining segments onto the plane $\pi_1 \perp \widehat{\boldsymbol{p}}_1$ and normalize
$$\overline{\boldsymbol{n}_i''} = \mathsf{N}(\boldsymbol{R}_1 \boldsymbol{P}_1 \boldsymbol{n}_i) \qquad (28)$$
again with the projection matrix
$$\boldsymbol{P}_1 = \boldsymbol{I} - \hat{\boldsymbol{p}}_1 \hat{\boldsymbol{p}}_1^T \qquad (29)$$
and the rotation matrix
$$\boldsymbol{R}_1 = \begin{pmatrix} \mathsf{N}(\boldsymbol{z} \times \hat{\boldsymbol{p}}_1)^T \\ \mathsf{N}[(\boldsymbol{z} \times \hat{\boldsymbol{p}}_1) \times \hat{\boldsymbol{p}}_1]^T \\ \hat{\boldsymbol{p}}_1^T \end{pmatrix} \qquad (30)$$
thus the complete projection reads as
$$\overline{\boldsymbol{n}_i''} = \mathsf{N}\left[\begin{pmatrix} \mathsf{N}(\boldsymbol{z} \times \hat{\boldsymbol{p}}_1)^T \\ \mathsf{N}[(\boldsymbol{z} \times \hat{\boldsymbol{p}}_1) \times \hat{\boldsymbol{p}}_1]^T \\ \mathbf{0}^T \end{pmatrix} \boldsymbol{n}_i\right] \qquad (31)$$
2. Find the peak in the empirical distribution of the $\overline{\boldsymbol{n}_i''} = (\cos\psi_i \ \sin\psi_i)$. Here the periodicity of $\pi/2$ is taken into account by replacing the angles ψ_i by $\psi_i \bmod \pi/2$
3. If step 2 succeds it yields the best normal $\overline{\boldsymbol{n}_{\text{max}}}''(\psi_{\text{max}})$ which when rotated back yields the second vanishing point $V_2(\hat{\boldsymbol{p}}_2^{(0)})$ with
$$\hat{\boldsymbol{p}}_2^{(0)} = \boldsymbol{n}_{\text{max}}'' = \boldsymbol{R}_1^T (\cos\psi_{\text{max}} \ \cos\psi_{\text{max}} \ 0)^T \qquad (32)$$
Otherwise go back to finding the first vanishing point.
4. An approximation for the third vanishing point $V_3(\boldsymbol{p}_3)$ results from
$$\hat{\boldsymbol{p}}_3^{(0)} = \mathsf{N}(\hat{\boldsymbol{p}}_1^{(0)} \times \hat{\boldsymbol{p}}_2^{(0)}) \qquad (33)$$
by using the legoland assumption.
5. Classify the line segments into $\mathcal{C}_2$, $\mathcal{C}_3$, and $\mathcal{C}_0$.

The rational behind this step is the following: If the first vanishing point actually belongs to one of the three main directions and a second vanishing point can be found, it is likely that a reduced histogram in step 2 actually shows one pronounced peak. If no peak in step 2 is found either the first vanishing point is the only one in the scene or it is not belonging to the three main directions of the legoland scene. Observe that the procedure up to this point does not check the existence of evidence of a third vanishing point before the classification step.

Discussion:

- The system requires three *control parameters*, namely the expected percentage of edges to the vanishing points (two values) and a tolerance for finding the peaks in the circular histograms.
- The system can also handle the case of *only two groups of parallel lines* (cf. figures 5 to 7).
- The system is limited to images with enough edges per vanishing point. A limiting case is shown in figures 8 to 10, where the determination is very unstable.
- The *computing time* for the clustering itself was in the range of 0.3 seconds.
- The procedure can be generalized to the case of *unknown interior orientation*, as the the search for the first vanishing point may be repeated without taking the orthogonality constraint into account.

4.3 Rotation Matrix by Clustering on the 3-Sphere

Instead of grouping the line elements into three groups, pointing towards the vanishing points the following procedure directly determines the parameters of the rotation.

The idea is to represent the $\boldsymbol{R}(\mathbf{q})$ by unit quaternions $\mathbf{q}$ on the S^3, determine for each line the subspace in S^3 and find the optimal $\mathbf{q}$ by clustering in S^3. The procedure has been realized (Heinen 1998) and empirically investigated.

For doing this we observe the following (cf. fig. 11):

Figure 5: *shows the image of a high building.*

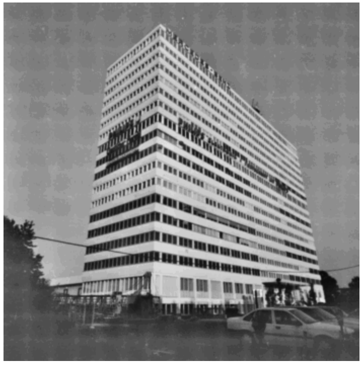
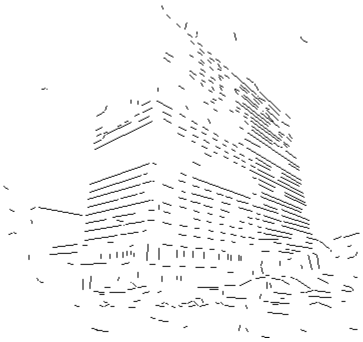


Figure 6: *shows the automatically extracted edges of the high building.*



1. The unknown rotation is determinable up to the elements of the rotation group of the hexagon or the octahedron. Due to the sign ambiguity of the quaternions this group therefore has 48 elements. Its voronoi tessellation consists of 48 spherical cubes with the corners cut off (cf. fig. 11, left). Thus the cells of the tessellation are polyhedra with spherical faces, 6 of them being regular octagons, 8 of them being regular triangles, all sides being great circles on S^3.

2. Given a vanishing point V in the image the rotation matrix $\boldsymbol{R}$ is defined up to a rotation around the axis OV from the projection center.

Figure 7: *shows the found vanishing points of the high building.*

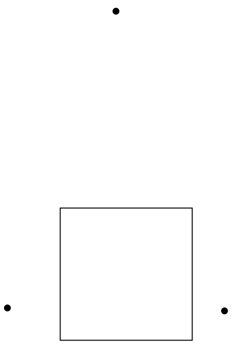


Figure 8: *shows the image of a nice building.*

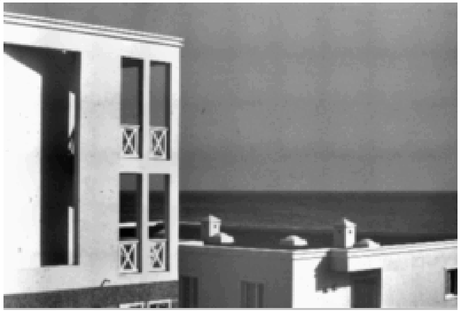
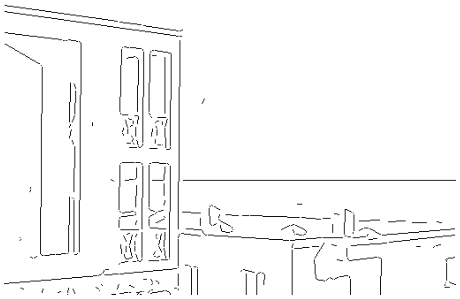


Figure 9: *shows the automatically extracted edges of the nice building. Observe the few edges for the left vanishing point.*



3. The set of all rotations around a given axis corresponds to a great circle in S^3.

4. For each line segment in the image there exist ∞ possible positions for the vanishing point, in case this line segment actually points towards a vanishing point. Thus the set of all rotations induced by a single line segment l corresponds to a set of great circles in S^3, thus a surface $\phi(l)$ in S^3.

   The set of line segments l'_{ij} pointing towards a common vanishing point V_j therefore leads to a set of surfaces ϕ_{ij} intersecting in one great circle c_j in S^3, in case no noise is present.

Figure 10: *shows the found vanishing points for the nice building.*

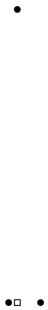
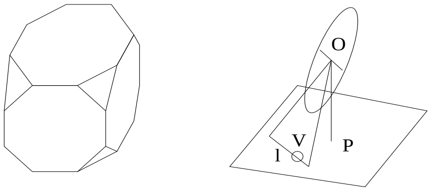


Figure 11: a) Central cell of voronoi tessellation of the octahedron group in S^3. b) Each line segment l' induces a set of vanishing points V_i, which can be sampled as directions from the projection center O towards the image plane and leads to torus on the 3-sphere.



Cell of O-Group Line and the representation of its vanishing points

5. The intersection of at least three of such surfaces $\phi_{l_i}, i = 1, 2, 3$, or at least two of such great circles c_{j_1} and c_{j_2} is the sought point $P(\widehat{\mathbf{q}})$ in S^3, represented by a unit quaternion and giving the sought rotation matrix.

The practical procedure is a clustering procedure. It uses the gnomonic projection of S^3 into $\mathbb{R}^3$. This projection maps points P of S^3 to a tangent plane τ by $P' = OP \cap \tau$, thus great circles to straight lines. This is an advantage when filling the cluster space.

The central voronoi cell can be tessellated parallel to the three axes of $\mathbb{R}^3$. The accumulator thus is restricted to a cube surrounding the gnomonic projection of the central cell of the voronoi tessellation of the permutation group. The restriction to the interior of the central cell around $(0, 0, 0)$ leads to an estimate for $\boldsymbol{R}$ which is closest to the unit matrix.

This procedure however requires to generate *three* great circles per vanishing point candidate on a line according to items 2 and 3, thus to three surfaces $\phi(l)$ for each line l, as the numbering of the vanishing points is not known.

The sampling of the possible vanishing points for one line segment can be performed on the unit circle around the projection center O in the projection plane $\varepsilon(O, l)$. Only half of this circle, preferably showing to the image plane, needs to be considered (cf. fig. 11b).

Due to the small size of the central cell no severe irregularities are to be expected in the density of the sampling of S^3.

Several problems needed to be considered:

1. Define the accumulator size, i. e. the size of the cube covering the central cell of the tessellation after the gnomonic projection. Also determine the number of cells.

   This has shown to be the range $[-0.41, 0.41]$ in all coordinates in the 3D tangent space (for details cf. (Heinen 1998)). The number of cells need to be larger than 60^3, as otherwise the peak in the accumulator could not be detected.

2. Determine the form of $\phi(l)$ for a line.

   Topologically it is a torus as the set of points on a line is cyclic. In the gnomonic projection it is a ruled surface, which is easy to use in the clustering procedure.

3. Derive an efficient procedure for filling the accumulator with a straight line.

   The Bresenham algorithm for drawing straight lines on a raster screen can be used to advantage

4. Analyze the uncertainty of the points on the line segments and the effects onto the corresponding circles and deriving an efficient procedure for filling the accumulator with an uncertain straight line.

   For efficiency this has been realized by smoothing the content of the accumulator after filling.

5. Empirically testing the system with real data and determining its performance.

   As expected, the accumulator has been found to be quite populated as for each line segment three surfaces need to be added. Also the peak in the accumulator has shown to be not well pronounced. However, the second largest peak showed to be low enough to result in a reliable solution in all cases which could be expected to be solvable. The procedure appeared to be quite robust. E. g. with only 5 lines in each of the three directions the rotation could be correctly determined. Though three lines would be sufficient in the ideal case, the multiple filling of the accumulator requires a larger number of lines per group. In case of 5 lines per group and less than 30 noisy edges the probability of finding the correct rotation was above 98 %.

Discussion:

- The procedure requires only one *control parameter* namely for evaluating the peak.
- The system can also handle the case of only *two groups of parallel lines*.
- The system requires enough lines per vanishing point.
- The procedure *cannot* directly be generalized to the case where the interior orientation is not given.

- The *computation time* was in the range of a minute, the main part being the smoothing of the accumulator. This appears to be the price payed for having a one step procedure.

5 OUTLOOK

The examples for estimating rotations show the advantages of the different representations. Eulerian angles were not used, due to the complexity of the resulting rotation matrix. Only a few applications are given. All orientation procedures in Photogrammetry requiring 3D rotations may take advantage of the algebraic representations. The statistical properties of the estimates have only touched. Explicit expressions for describing the accuracy, i. e. covariance matrices for the estimated rotation parameters need to be investigated.

REFERENCES

Ackermann, F.; Ebner, H.; Klein, H. (1970): Das Programm PAT-M für die Aerotriangulation mit unabhängigen Modellen. *BuL*, (38):206–217, 1970.

Arun, K. S.; Huang, T. S.; Blostein, S. D. (1987): Least-Squares Fitting of Two 3-D Point Sets. *IEEE T-PAMI*, 9(5):698–700, 1987.

Brandstätter G. (1991): Notizen zur vorraussetzungslosen gegenseitigen Orientierung von Meßbildern. *ÖZfVuPh*, 79(4):273–280, 1991.

Coxeter H. S. M. (1963): Unvergängliche Geometrie. Birkhäuser, 1963.

Ebner, H. (1968): *Genauigkeitsuntersuchung zur photogrammetrischen Sternkoordinatenbestimmung durch geschlossene Bündelausgleichung*. PhD thesis, Universität Stuttgart, DGK C 141, München, 1968.

Faugeras, O. D.; Hebert, M. (1983): A 3-D Recognition and Positioning Algorithm using Geometrical Matching between Primitive Surfaces. In: *Proc. Int. Joint. Conf. on Art. Intell.*, pages 996–1002, 1983.

Faugeras, Olivier (1993): *Three-Dimensional Computer Vision*. The MIT Press, 1993.

Förstner, W. (1999): Interior Orientation from Three Vanishing Points. Technical report, Institut für Photogrammetrie, Universität Bonn, 1999.

Heinen, H. H. (1998): Automatische Bestimmung der Kameralage von Legoland-Bildern. Master's thesis, Institut für Photogrammetrie, Universität Bonn, 1998.

Horn, B. K. P. (1990): Relative Orientation. *IJCV*, 4(1):59–78, 1990.

Horn, K. P. (1987): Closed-Form Solution of Absolute Orientation using Unit Quaternions. *J. of Optical Society of America*, 4:629–642, 1987.

Rinner, K.; Burkhardt, R. (Eds.) (1972): *Jordan/Eggert/Kneissl: Handbuch der Vermessungskunde*, Band IIIa/1, Photogrammetrie. Metzlersche Verlags-Buchhandlung, Stuttgart, 1972.

Rinner, K. (1957): *Über räumliche Drehungen*. DGK A 25, München, 1957.

Rodrigues, M. O. (1840): Des lois géométrique qui régisseut les déplacements d'un système solide dans l'espace, et de la variation des coordinées provent de ces déplacements considérés indépendamment des causes qui peupent les produire. *J. de Mathématique pures et appliquées, Bachelier Paris*, 5:380–440, 1840.

Sanso, F. (1973): An Exact Solution to the Roto-Translation Problem. *Photogrammetria*, 29:203–216, 1973.

Schut, H. (1958/59): Construction of Orthogonal Matrices and Their Applications in Analytical Photogrammetry. *Photogrammetria*, 15(4), 1958/59.

Wydera, M. (1995): Automatische Bestimmung von Fluchtpunkten. Master's thesis, Institut für Photogrammetrie, Universität Bonn, 1995.