

COMPACT - A Surface Representation Scheme

Pavel Grossmann*
Long Range Research Laboratory
GEC Hirst Research Centre
East Lane
Wembley
Middlesex HA9 7PP

This paper describes the continuation of our work on extracting simple surfaces from a 3D line-segment representation of stereo image data. The surface parameters form a basis of a scene representation which is suitable for navigation by a mobile robot and also for recognition and manipulation of simple objects by a robot arm.

Following our earlier work on planes we have now developed methods and algorithms for recovering spherical, cylindrical and conical surfaces. The essence of our approach is testing of small groups of 3D line segments for compatibility with a particular surface type. Maximal sets of segments supporting different surfaces are then identified. In this we make a novel use of the Dual Space Representation.

1 Introduction

Choice of the best representation of image data obviously depends on the purpose for which the representation is being designed - e.g. an object recognition and manipulation task may require a more complex and accurate representation of the visible surfaces than a simple indoor navigation task. The choice also depends on the nature of the image data itself - e.g. dense depth maps generated by a laser range finder and the sparse depth data produced by a stereo edge-based system may give rise to very different representations.

In principle we can use all of the usual representation primitives based on points, lines, surfaces or volume elements [1,2]. Many present systems, however, use predominantly only one of the geometrical primitives and try to explore its full potential. In some applications points [3] or volumes offer specific advantages, but in the domains of indoor scenes or manufactured objects, simple lines and surfaces can provide a basis for a more compact representation.

These are the domains of interest in our collaborative ESPRIT project P940 - "Depth and motion analysis". The aim of the project is to develop a vision system capable of guiding a mobile robot in indoor navigation tasks and also a robot arm in object recognition and manipulation tasks.

*This research was supported in part by the ESPRIT project P940.

Many indoor scenes can be adequately described in terms of planes, and also a large category of manufactured objects that are being considered for our recognition and manipulation tasks can be adequately described in terms of four surface types: planes, cylinders, cones and spheres. We have therefore proposed (and are currently implementing at the Long Range Research Laboratory, HRC, Wembley) a compact scene representation scheme (COMPACT) based on those four surface primitives.

The input to the higher level representation is a set of 3D line segments produced by the initial stages of our three-camera stereo vision system [4].

It has been shown that, given a set of 3D line segments, one can construct closed continuous contours (or parts thereof) that directly imply surfaces (usually planar faces of simple polyhedral objects [5], although this approach has also been used to locate cylinders in images [6]). There are two problems associated with this approach: it often requires complete surface contours and also it makes no use of the segments "inside" the surface boundaries that often make up most of the data (e.g. segments corresponding to windows, pictures and other features of walls in interior scenes).

Our approach is based on direct extraction of surface candidates from groups of segments and subsequent merging of the compatible candidates into finite surfaces.

2 The method

2.1 General description

We shall explain the basic method on the example of a planar surface (which has already been described in detail in [7,8]). Here we consider pairs of segments: each pair is tested for coplanarity and each coplanar pair is used either to update an existing surface candidate or to form a new candidate. When all the pairs are exhausted, some surface candidates are removed if the number of segments supporting them is small, and some merged (if compatible) to yield a small number of significant surfaces. The surface parameters (in the planar case the surface normal and the perpendicular distance to the origin) are being continuously updated. We shall refer to this as the "surface growing" method.

The Dual Space Representation (DSR - see Appendix A and [9]) offers an alternative way of finding planes. Here a point, a line and a plane in space are represented

in Dual Space by a plane, a line and a point respectively. Thus a set of coplanar lines in space is mapped onto a set of lines intersecting in a single point in Dual Space. When Dual Space is divided into small cells (similar to the Hough accumulator) such intersections can be found as maxima in the number of lines intersecting each cell.

The main advantage of this (DSR) method (apart from its elegance) is its speed - instead of being proportional to the number of segment pairs, the computing time here is linear in the number of segments (with some additional - fixed - time for the maxima search). On the other hand there is a limit to the number of maxima we can identify in a noisy Dual Space population. Also the treatment of errors is far less straightforward due to the granularity of Dual Space and the nature of the mapping transformations.

2.2 Treatment of errors

The description of uncertainties associated with all the quantities in our calculations is based on the assumption [10] that a covariance matrix always provides a sufficient parametrization of the relevant probability distribution and that the covariant matrices can be propagated using the standard formula [11] :

$$V_{kl}(\bar{y}) = \sum_{i=1}^n \sum_{j=1}^n \frac{\partial y_k}{\partial x_i} \frac{\partial y_l}{\partial x_j} V_{ij}(\bar{x})$$

where \bar{y} is a set of m functions which all depend on the set of n random variables \bar{x} ($y_k = y_k(\bar{x})$). The error on any quantity is then given by the square root of the relevant variance.

Starting with the experimentally determined covariance matrices of the segment end points we first compute the errors for the segment parameters (unit vector and length), then for the intermediate quantities (e.g. vector products) and finally for the resultant surface parameters.

In the surface growing approach the surface parameters are continuously updated by the addition of new estimates and the updated values are determined as the weighted means of all the previous measurements. Our method of computing the (in general correlated) parameters and the relevant variances is sketched out in Appendix B.

In addition, to take into account the departure from ideal planes of many real quasi-planar surfaces, that we perceive (and hence wish to identify) as planar, we have introduced the concept of "perceptual resolution". As an example, consider a set of segments representing a window. For the purposes of recognition we may wish to regard all the segments as belonging to the same planar surface although they may be contained in a rectangular volume of space some 10-20 cms thick. When a segment is tested for consistency with a particular plane, we consider not only the errors on the segment and plane parameters, but also the perceptual resolution.

Hence it is not always easy to test the errors except in some global sense. As an example consider an office scene represented by the 3D segments in Figure 1. There are two main vertical planes there corresponding to the left

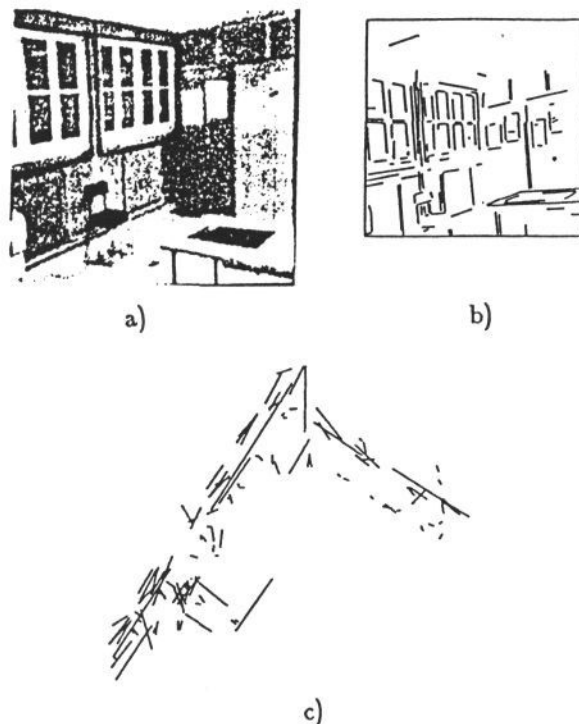


Figure 1: INRIA office scene
a) one of the triplet of images
b) front view of the 3D segments
c) top view of the 3D segments

wall (with windows) and the cabinets on the right (with posters). While the distribution of segments associated with each plane is quite "noisy", we would expect the plane normals \bar{n}_1 and \bar{n}_2 to be well determined and in our case perpendicular (see Figure 1c). In our analysis we get :

$$\cos \theta = \bar{n}_1 \cdot \bar{n}_2 = 0.020 \pm 0.027$$

i.e. the walls are perpendicular within the (small) error.

2.3 Spheres

A spherical surface can be identified by examining triplets of line segments [7,12]. Strictly speaking our method is based on triplets of surface tangents. Here we make an assumption that a line segment, which is an approximation to a section of a surface curve, is also a good approximation to a surface tangent \bar{u}_i at a surface point close to the segment midpoint P_i . Experiments with real data (see our results shown below) suggest that this assumption is reasonable. A triplet of surface tangents defines a triplet of planes ($\bar{r}\bar{u}_1 = p_1$, $\bar{r}\bar{u}_2 = p_2$ and $\bar{r}\bar{u}_3 = p_3$) normal to the tangents, that intersect in a single point \bar{R} (Figure 2) whose position can be computed [13] :

$$\bar{R} = \frac{p_1(\bar{u}_2 \times \bar{u}_3) + p_2(\bar{u}_3 \times \bar{u}_1) + p_3(\bar{u}_1 \times \bar{u}_2)}{\bar{u}_1(\bar{u}_2 \times \bar{u}_3)}$$

unless $\bar{u}_1(\bar{u}_2 \times \bar{u}_3) = 0$. If the distance from \bar{R} to the three surface points P_i is the same ($= D$), the segments are consistent with a spherical surface which is described by its centre \bar{R} and its radius D .

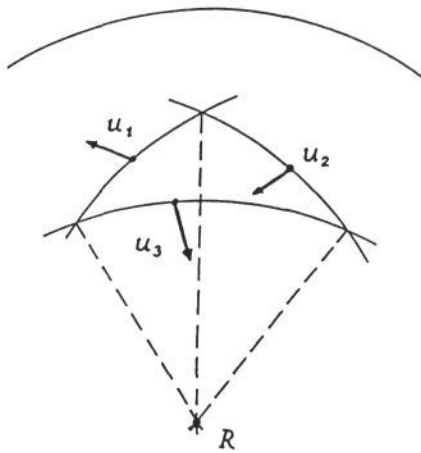


Figure 2: Test of the spherical surface hypothesis

2.4 Cylinders and Cones

The cylinders and cones are treated in the same way. The basic idea is to find the axis of the cylinder or cone as an intersection of several planes that are normal to the surface and that contain the axis. In order to find such planes we make an assumption that our line segments are parts (or approximations thereof) of the surface lines of curvature that are planar and whose associated curvature is constant. Such lines have been found to be important descriptors of surfaces [14]. On cylindrical and conical surfaces these are either straight lines (parallels) or circles (meridians).

For each pair of parallels we can construct their plane of mirror symmetry, which passes through the axis of the surface. Hence the axis can be found as the intersection of several such planes (Figure 3a).

For each segment that approximates a part of a meridian we can find a plane normal to the segment which also passes through the axis. And similarly the axis can be found as the intersection of several such planes (Figure 3b).

The task of searching for such plane intersections is made simpler when we use the Dual Space Representation (DSR). Here a set of planes intersecting in a single line in 3-space is transformed into a set of points lying on a line in Dual Space. This (dual) line can be found by using the Hough transform or other standard methods.

The requirement that a significant fraction of segments is of the "parallel" or "meridian" type may appear somewhat restricting. There are many objects with more or less random distribution of surface markings or, indeed, with no markings at all. For such cases we proposed [12] a way of generating a set of parallels from the extremal (self-occluding) boundaries of the object (we are still talking about cylinders and cones!).

In the mode of operation of the optical system in question, when the cameras move around the object (e.g. a cylinder), accumulating and combining information in order to build a more complete description of the visible surface, different projections of the "real" lines in space will coincide when projected back into space. The extremal boundaries, however, will move along the surface as the cameras move and so they can be identified as

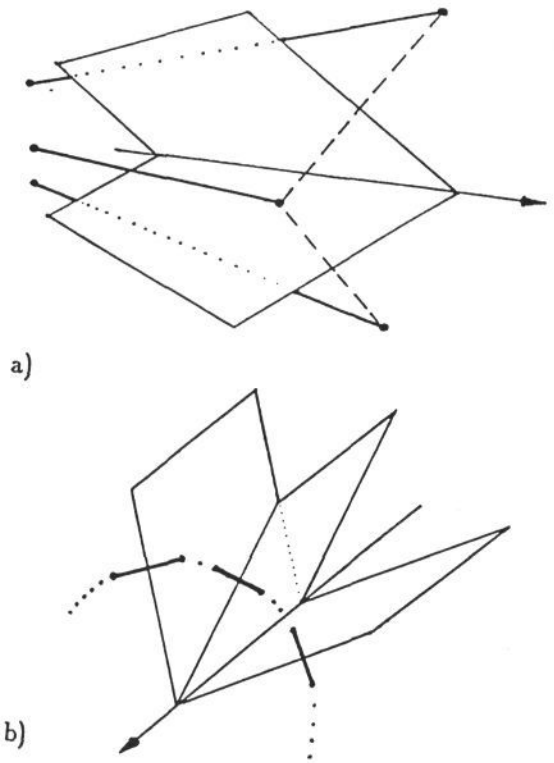


Figure 3: Determining the axis of a cylinder or a cone
a) from a set of parallels
b) from a set of meridians

such. Their projections in 3-space will not coincide; they will form a set of different rulings that can be used in our method. This way of generating useful sets of segments is currently being tested.

3 Implementation and results

The methods described above have been implemented as a suite of programs written in Franz Lisp and running on our VAX 11-750 with the Unix operating system. It is anticipated that some parts will be implemented in 'C'.

The algorithms have been tested with simulated data, both ideal and with superimposed errors, and also with real sets of 3D line segments from the pool of image data provided for the collaboration by our ESPRIT colleagues. The planar algorithms were tested using office scenes (Figure 1) from INRIA (Rocquencourt) and to test the non-planar methods we used triplets of images of a sphere and a cylinder obtained at ELSAG (Genoa) and processed at INRIA where the 3D segments were extracted (Figures 6 and 7).

Time taken to process a typical office scene (~150 segments, Figure 1) is of the order of several minutes. The DSR mapping of segments into the Dual Space is faster, while the spherical algorithm based on segment triplets can take several tens of minutes. At the moment the algorithms are not optimized for speed and we expect considerable improvement in the running times in the future.

The planar case (Figure 1) has already been discussed elsewhere [7,8] and so here we shall concentrate on the quadric surfaces.

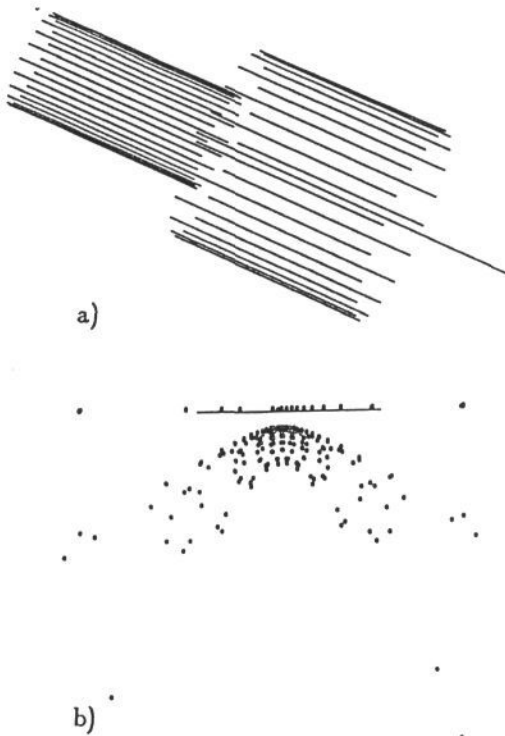


Figure 4: Simulated (ideal) data
 a) two coaxial cylinders - 3D segments
 b) a projection of Dual Space

Figure 4a shows an example of the synthetic (ideal) data. Here we have two coaxial cylinders and all the segments are "parallels" on the surfaces. Because the cylinders are coaxial, both separate sets of segments map onto the same straight line in Dual Space (Figure 4b). There is, however, also a contribution from the "mixed" pairs of segments (each segment from a different cylinder) which constitutes a background. The straight line parameters were determined by the Principal Axis method used in an iterative mode to reject outliers at every step and to gradually "home in" onto the line. This line was then mapped back to the ordinary space to give the correct position of the surface axis (shown in Figure 4a). Once the axis is known, the two radii can easily be computed.

Figure 5 shows the effect of realistic errors being imposed on the ideal segments (5a). There is a scatter of points about the "best line fit" in Dual Space (Figure 5b) that translates into a finite spread of the angle between the segments and the computed axis about the ideal (i.e. zero errors) value.

Finally in Figures 6 and 7 we show the results obtained with real data. For the test of the spherical algorithm we used an image of a ball painted with a simple pattern, resting on a table. For the corresponding set of 3D segments shown in Figure 6a the program found only one significant spherical surface candidate and the segments assigned to it are shown in Figure 6b. The radius of the sphere has been computed to be 46 mm compared with the measured value of 45 mm.

Similarly to test the cylindrical algorithm we used an image of a cylinder decorated with a simple pattern. Each of the corresponding segments in Figure 7a was as-

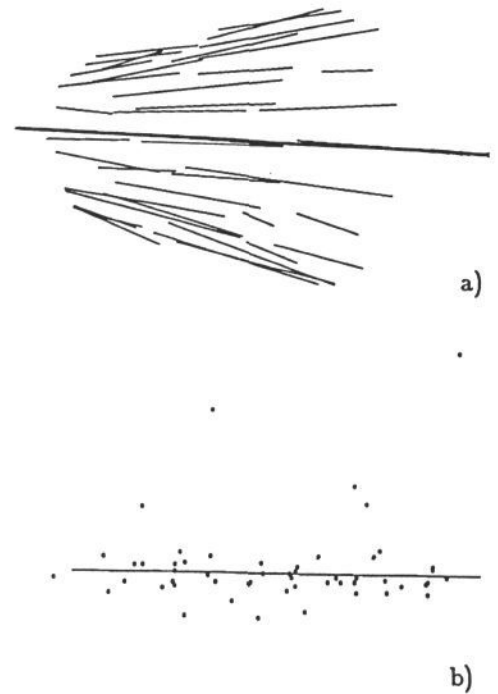


Figure 5: Simulated (noisy) data - a cone
 a) 3D segments and the reconstructed axis
 b) x-y projection of the Dual Space

sumed to be a part of a meridian although many segments have very different orientation. Nevertheless, a straight line fit to the corresponding Dual Space distribution (Figure 7b) gave a reasonable estimate of the cylinder axis as shown superimposed on the data in Figure 7a.

4 Future work

Having demonstrated our ability to determine the surface parameters of each of the four surface types from the line segment representation of image data, we now have to integrate our algorithms into a single system capable of dealing with any mixture of the basic surface types present in complex scenes.

Considering the use of our scheme for navigation and object recognition we have to develop ways of matching the representation of a scene to the representation of the same scene viewed from a different viewpoint or to an appropriate representation of familiar objects likely to be found in the scene. We are also investigating methods for interpretation of image data without referring to geometrical models. This will be done by analyzing the spatial relations between the surfaces and using domain specific knowledge to provide additional constraints.

5 Acknowledgements

Thanks are due to my colleagues in the ESPRIT P940 collaboration (INRIA and ELSAG) for the use of their image data (in Figures 1,6 and 7) and to Mike Brady for many inspiring discussions.

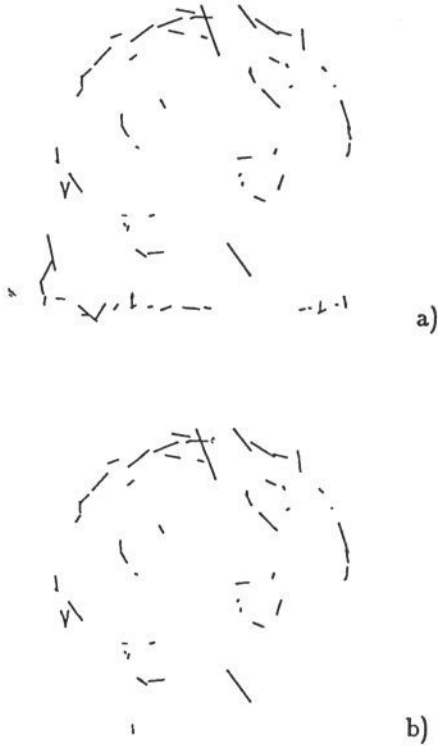


Figure 6: A real sphere
a) all 3D segments
b) segments found to belong to the sphere

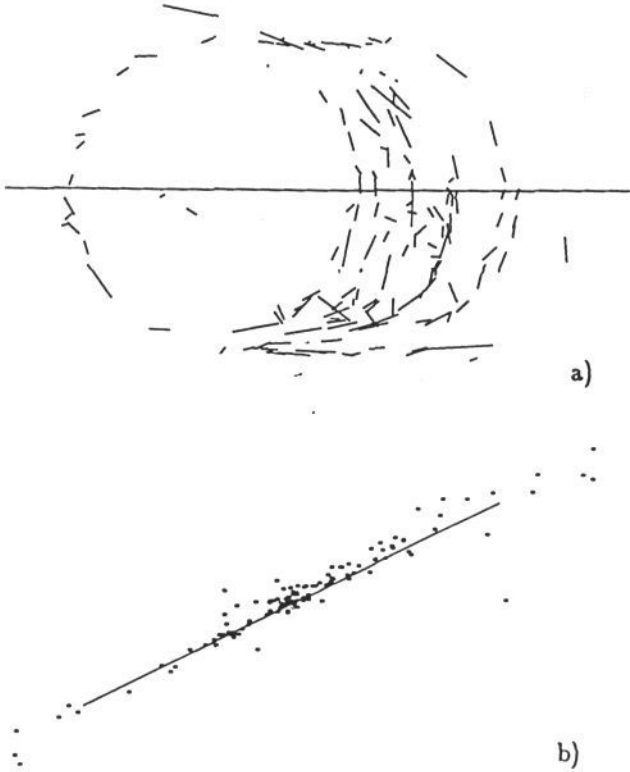


Figure 7: A real cylinder
a) 3D segments and the computed axis
b) a projection of the Dual Space

A Dual Space Representation

A.1 Dual point to a plane

A plane with an unit surface normal $\vec{N} = (N_x, N_y, N_z)$ passing through a point \vec{V} is mapped to a dual point (p, q, r) :

$$(p, q, r) = \left(\frac{N_x}{N_z}, \frac{N_y}{N_z}, -\frac{\vec{N}\vec{V}}{N_z} \right)$$

if $N_z \neq 0$

A.2 Dual plane to a point

A point $\vec{V} = (x, y, z)$ is mapped to a dual plane with an unit surface normal $\vec{N} = \left(\frac{x}{B}, \frac{y}{B}, \frac{1}{B} \right)$ and passing through a dual point $\vec{V} = \vec{N}(-\frac{z}{B})$ where $B = \sqrt{(1+x^2+y^2)}$.

A.3 Dual line to a line

A line with a unit vector $\vec{L} = (L_x, L_y, L_z)$ and passing through a point \vec{V} ($\vec{r} = \vec{V} + \lambda\vec{L}$) is mapped to a dual line joining the following two points S_1 and S_2 :

$$\vec{S}_1 = (p^1, q^1, r^1) = \left(\frac{N_x^1}{N_z^1}, \frac{N_y^1}{N_z^1}, -\frac{\vec{N}\vec{V}}{N_z^1} \right)$$

$$\vec{S}_2 = (p^2, q^2, r^2) = \left(\frac{N_x^2}{N_z^2}, \frac{N_y^2}{N_z^2}, -\frac{\vec{N}\vec{V}}{N_z^2} \right)$$

if $N_z^1, N_z^2 \neq 0$

which themselves are dual to the two planes passing through the point \vec{V} and having the following unit surface normals :

$$\vec{N}^1 = \frac{\vec{L} \times \vec{V}}{|\vec{L} \times \vec{V}|}$$

$$\vec{N}^2 = \frac{\vec{L} \times (\vec{L} \times \vec{V})}{|\vec{L} \times (\vec{L} \times \vec{V})|}$$

As points, lines and planes are interrelated, so are their duals.

B Errors for correlated vector quantities

Let us consider estimation of a vector quantity $\vec{\mu}$ given n measurements $(\vec{x}_1, \vec{x}_2, \dots, \vec{x}_n)$ and also the (different) covariance matrices V_i . We shall use the Maximum Likelihood principle to derive the appropriate estimator. Starting from the likelihood function [11] :

$$L(\vec{x}_i, \vec{V}_i; \vec{\mu}) = \prod_{i=1}^n \frac{1}{(2\pi)^{\frac{n}{2}} |V_i|^{\frac{1}{2}}} e^{-\frac{1}{2}(\vec{x}_i - \vec{\mu})^T V_i^{-1} (\vec{x}_i - \vec{\mu})}$$

we require that :

$$\frac{\partial \ln L}{\partial \mu_k} = 0$$

In the case of 3-dimensional vector quantity ($k = 1, 2, 3$) the above condition leads to the following 3 inhomogeneous equations (for better readability we shall drop the summation index i and denote the elements of the inverse covariance matrices V^{-1} by M_{kl}):

$$\begin{aligned} \mu_x \sum M_{xx} + \mu_y \sum M_{xy} + \mu_z \sum M_{xz} &= \\ &= \sum x M_{xx} + \sum y M_{xy} + \sum z M_{xz} \\ \mu_x \sum M_{yx} + \mu_y \sum M_{yy} + \mu_z \sum M_{yz} &= \\ &= \sum x M_{yx} + \sum y M_{yy} + \sum z M_{yz} \\ \mu_x \sum M_{zx} + \mu_y \sum M_{zy} + \mu_z \sum M_{zz} &= \\ &= \sum x M_{zx} + \sum y M_{zy} + \sum z M_{zz} \end{aligned}$$

The solution can be given as follows:

$$\mu_x = \frac{-D_x}{D} \quad \mu_y = \frac{D_y}{D} \quad \mu_z = \frac{-D_z}{D}$$

where:

$$D = \begin{vmatrix} \sum M_{xx} & \sum M_{xy} & \sum M_{xz} \\ \sum M_{yx} & \sum M_{yy} & \sum M_{yz} \\ \sum M_{zx} & \sum M_{zy} & \sum M_{zz} \end{vmatrix}$$

and the determinants D_x, D_y and D_z are formed by replacing the "x,y or z" column in D by the right hand side of the equations above:

$$D_x = \begin{vmatrix} \sum M_{xy} & \sum M_{xz} & \sum (xM_{xx} + yM_{xy} + zM_{xz}) \\ \sum M_{yy} & \sum M_{yz} & \sum (xM_{yx} + yM_{yy} + zM_{yz}) \\ \sum M_{zy} & \sum M_{zz} & \sum (xM_{zx} + yM_{zy} + zM_{zz}) \end{vmatrix}$$

and similarly for D_y and D_z .

Using a similar approach we can also derive the estimator for the covariance matrix associated with the vector $\bar{\mu}$. The result is simple - to obtain the inverse of the covariance matrix we sum up the inverse covariant matrices of the individual measurements:

$$V^{-1}(\mu) = \sum_{i=1}^n V^{-1}(x_i)$$

References

- [1] J. B. Bowen and J. E. W. Mayhew. *Consistency Maintenance in the REVgraph Environment*. Technical Report 20, AI Vision Research Unit, Sheffield University, 1986.
- [2] J. L. Crawley. Representation and maintenance of a composite surface model. In *IEEE Int. Conf. on Robotics and Automation*, 1986.
- [3] C. G. Harris. Determination of ego-motion from matched points. In *Proceedings of the Third Alvey Vision Conference*, 1987.

- [4] Nicolas Ayache and Francis Lustman. Fast and reliable passive trinocular stereovision. In *Proceedings of the First International Conference on Computer Vision*, 1987.
- [5] Martin Herman and Takeo Kanade. Incremental reconstruction of 3D scenes from multiple, complex images. *Artificial Intelligence*, 30, 1986.
- [6] Kashipati Rao and R. Nevatia. Generalized cone descriptions from sparse 3-d data. In *Proceedings CVPR'86*, 1986.
- [7] Pavel Grossmann. COMPACT - a 3D shape representation scheme for polyhedral scenes. In *Proceedings of the Third Alvey Vision Conference*, 1987.
- [8] Pavel Grossmann. *Building Planar Surfaces from Raw Data*. Technical Report R4.1.2, ESPRIT Project P940, 1987.
- [9] M.C.Err. *Computer Interpretation of Engineering Drawings*. PhD thesis, University of Essex, 1985.
- [10] Hugh F. Durrant-Whyte. Uncertain geometry in robotics. *IEEE Journal of Robotics and Automation*, 4(1), 1988.
- [11] O.Skjeggestad A.G.Frodesen and H.Tøfte. *Probability and Statistics in Particle Physics*. Universitetsforlaget Oslo, 1979.
- [12] Pavel Grossmann. *Planes and Quadrics from 3D Segments*. Technical Report R4.1.6, ESPRIT Project P940, 1988.
- [13] I.D.Faux and M.J.Pratt. *Computational Geometry for Design and Manufacture*. Ellis Horwood Ltd., 1981.
- [14] Michael Brady et al. Describing surfaces. *Computer Vision, Graphics and Image Processing*, 32, 1984.