“Dual Representations” for Vision-Based 3D Reconstruction

Etienne Grossmann and Jos Santos-Victor
Instituto de Sistemas e Robtica
Instituto Superior Tcnico
Av. Rovisco Pais, 1
1049-001 Lisboa
Portugal
{etienne,jasv}@isr.ist.utl.pt

Abstract

We consider the problem of representing sets of 3D points in the context of 3D reconstruction from point matches. We present a new representation for sets of 3D points, which is general, compact and expressive : any set of points can be represented; geometric relations that are often present in man-made scenes, such as coplanarity, alignment and orthogonality, are explicitly expressed. In essence, we propose to define each 3D point by three independent linear constraints that it verifies, and exploit the fact that coplanar points verify a common constraint. We show how to use the dual representation in Maximum Likelihood estimation, and that it substantially improves the precision of 3D reconstruction.

1 Introduction

We present a new representation for sets of 3D points and show that this representation is relevant to the problem of 3D reconstruction from point matches. In this problem[4], the input data is a collection of pixel coordinates taken from a sequence of three or more images. These are obtained by tracking along the sequence the projections in the images of 3D points which we wish to recover. The desired output is

1. The coordinates of the 3D points,
2. The positions of the cameras
3. Some characteristics (intrinsic parameters) of the camera.

Figure 1 illustrates the problem of 3D reconstruction as considered in the present work. Most methods published to day represent the reconstructed points by their coordinates in a given basis. No particular geometric relation is recognized between the reconstructed points. In man-made scenes, which are a common and important special case for reconstruction, there exist geometric properties such as coplanarities, parallelism, orthogonality. The representation of 3D points that we propose explicitly describes these

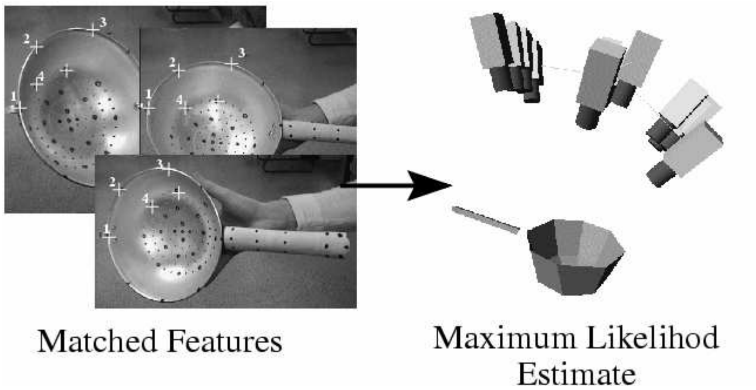

Figure 1: The problem of 3D reconstruction.

relations. Other work [1, 11, 10] has shown that these relations can be exploited to improve the reconstruction. In [1], geometric properties, known beforehand, are expressed by using polynomial constraints. In contrast, we use only linear constraints to represent 3D points, and bilinear restrictions to constrain the "directions" (see Section 2.1) used in our representation.

Szeliski and Torr [11] show how various geometric constraints can influence the precision of 3D reconstruction. Like us, they use parallelism and orthogonality constraints. In their work, a collection of techniques are presented, and their output is compared on a series of data set. In comparison, we limit ourselves to Maximum Likelihood estimators. The reason for our choice is multiple : first, we can argue that these estimators give "the best" estimates, in a precise probabilistic meaning [5]. Second, the error committed can be characterized [8, 7] : we know both the estimate and the covariance of the estimator that produced it.

A fundamental difference between [1, 11, 10] and this paper lies in the use of a scheme, the "dual representation", for representing a set of 3D points together with its properties. In [1], points are represented by their coordinates, and constraints are imposed in the estimation process. In [11] and [10], computation steps are inserted to the process of estimation, in accordance to the known geometric properties.

In the present work, points are represented together with their properties, or rather, *by* their properties. Because we explicitly represent only properties, and some geometric properties may be shared by many points, the number of parameters needed in the representation is often smaller than in the equivalent coordinate-based representation. Sets of 3D points are obtained from a vector of real parameters. Because this mapping is differentiable (Section 2.1), it can be used within an optimization scheme such as ML estimation.

1.1 Representation

In the context of shape-from-X, many representations of shape have been proposed : 2-and-1/2-D, voxels, generalized cylinders [3], CAD-like parametric models[9] and certainly many others. In the spirit of Grenander [6], we build our model from building blocks, which in our case are linear constraints. Contrarily to what is usually the case, each building block does not represent one or more points, but rather *a property* of some points. Each point is defined by three building blocks.

Using a single orthogonal basis provides some mathematical as well as algorithmic simplifications, but by doing so, some interesting geometric properties, such as coplanarity, parallelism and alignment, are occluded.

It is well known that a 3D point $\mathbf{x}$ is defined by any three independent linear constraints of the form $v_i = \mathbf{f}_i^*(\mathbf{x})$ where the $\mathbf{f}_i^*$ are independent linear function from R^3 to R; that is, the $\mathbf{f}^*$ are elements in the dual space of R^3. The dual space [2] of R^3 is the vectorial space of linear functions from R^3 to R. Its elements can be identified with points of R^3 because any function $\mathbf{f}^*$ in it can be defined by $\mathbf{f}^*(\mathbf{x}) = \mathbf{f}^\top \mathbf{x}$ for some $\mathbf{f} \in R^3$.

If some points, represented in such a way, have a linear constraint in common, they are necessarily coplanar. If they have two constraints in common, they are aligned. If the points $\mathbf{x}$ of a set verify $\mathbf{f}^*(\mathbf{x}) = v$ while points $\mathbf{x}'$ of another set verify $\mathbf{f}^*(\mathbf{x}') = v'$, then these sets belong to parallel planes. If moreover both sets verify a common constraint $\mathbf{g}^*(\mathbf{x}) = \mathbf{g}^*(\mathbf{x}') = w$, then these sets belong to parallel lines. By continuing in this way, it is easy to represent grids of points. Our building blocks, linear constraints, combine in a simple way to express explicitly geometric relations. We call this representation the *"Dual Representation"* because it is based on elements of the dual space of R^3.

To obtain a dual representation, one must first associate a set of 3D points to a number of meaningful constraints (planarities). In this paper, like in [1, 11, 10] , we assume that these geometric relations are known a-priori, and concentrate primarily on the framework of estimation theory.

Automatically determining from the data the "interesting" geometric properties is a separate problem that we *do not* address in this article. This problem is of course of great importance : specifying by hand the coplanarity and orthogonality relations is not very practical. We are currently working on automatic methods, preliminary results being available in the experimental (3) section.

A formal definition is given in Section 2, where relations of orthogonality are taken into account. We also show how how to use this representation in Maximum Likelihood estimation. In Section 3, it is shown that the dual representation can improve the precision of 3D reconstruction obtained from matched points.

2 Dual Representation

In this section, we define the dual representation (DR), give some examples of DR's and show how a DR can be integrated in an optimization problem over sets of points. We use the convention that boldface letters, like "$\mathbf{x}$", represent points in R^3 ("3D points") and normal letters, like "v", will represent scalars or collections of objects. Adopting matrix notation, 3D points will be identified with 3-by-1 matrices.

2.1 Definition

A *dual representation* for the set $x = (\mathbf{x}_1, \ldots, \mathbf{x}_P)$ of points in R^3 is given by a quintuplet $\Delta = (d, v, \delta, \beta, \Omega)$ consisting of :

1. The *directions* $d = (\mathbf{d}_1, \ldots, \mathbf{d}_D)$ of R^3 where $\forall i \; \|\mathbf{d}_i\| = 1$.
2. The *values* $v = (v_1, \ldots, v_V)$ in R.
3. The *direction indices* $\delta = (\delta_{11}, \delta_{21}, \delta_{31}, \delta_{12}, \ldots, \delta_{1P}, \delta_{2P}, \delta_{3P})$ where the δ_{ip} are natural numbers in $1 \ldots D$.
4. The *value indices* $\beta = (\beta_{11}, \beta_{21}, \beta_{31}, \beta_{12}, \ldots, \beta_{1P}, \beta_{2P}, \beta_{3P})$ where the β_{ip} are natural numbers in $1 \ldots V$.
5. The *orthogonality constraints,* $\Omega = (\Omega_1, \ldots, \Omega_D)$. Each direction $\mathbf{d}_i$ is constrained to be orthogonal to all the (at most two) $\mathbf{d}_{\omega_{ij}}$, where $\omega_{ij} \in \Omega_i$.

Each point $\mathbf{x}_p$ is unambiguously defined by :

$$\underbrace{\begin{bmatrix} \mathbf{d}_{\delta_{1,p}}^\top \\ \mathbf{d}_{\delta_{2,p}}^\top \\ \mathbf{d}_{\delta_{3,p}}^\top \end{bmatrix}}_{M_p} \mathbf{x}_p = \underbrace{\begin{bmatrix} v_{\beta_{1,p}} \\ v_{\beta_{2,p}} \\ v_{\beta_{3,p}} \end{bmatrix}}_{\mathbf{b}_p},$$

where it is assumed that M_p is not singular[1]. The indices δ_{ip} and β_{ip} select which directions and values are used in the definition of the point $\mathbf{x}_p$[2]. Thus a dual representation consists of a discrete part, δ and β, which we shall call the *"shape"* of Δ, and a continuous part, d and v, its *"parameters*". We will use $x(\Delta)$ to represent the points in R^3 defined by Δ.

Finding δ and β, the "shape", consists in associating 3D points to planes. In the context of this paper, we assume that the "shape" terms have been determined a-priori, so that we can focus on the "parameter" estimation. We are currently developing automatic methods to extract the shape components (i.e. finding the most adequate planarity constraints from the data points) and preliminary results can be found in Section 3.

When counting the number of parameters used in a dual representation, we count 2 parameters per direction, because of the restriction on the norm of the $\mathbf{d}_i$; one parameter is counted for each value v_i. When speaking of *directions,* we will always mean unit vectors. A *constraint* of the dual representation is any pair $(\mathbf{d}_i, v_j)$, or equivalently (i, j) such that at least one point in $x(\Delta)$ is defined from, amongst others, the relation $\mathbf{d}_i^\top \mathbf{x}_p = v_j$. The set of points using the constraint (i, j) in its definition is called the *support* of that constraint. The vocabulary we just defined is sufficient to express ourselves clearly in the rest of this article. It can be extended and formulated mathematically.

[1] If M_p is singular, then the dual representation is badly formed.

[2] These clumsy indices are necessary for the rigorous definition of dual representations. We shall avoid using them when possible, and introduce some vocabulary to that effect.

Example of a "Trivial" dual representation : A possible DR for x is

1. $d = \left(\mathbf{d}_1 = [1, 0, 0]^\top = \mathbf{e}_1,\, \mathbf{d}_2 = [0, 1, 0]^\top = \mathbf{e}_2,\, \mathbf{d}_3 = [0, 0, 1]^\top = \mathbf{e}_3\right)$. The directions defined by the canonic basis of R^3.
2. $v = (x_{11}, x_{21}, x_{31}, x_{12}, \ldots, x_{3P})$. All the coordinates of the 3D points. Note that $v_{3p+i-3} = x_{ip}$ for all $1 \leq i \leq 3$ and $1 \leq p \leq P$.
3. $\delta_{ip} = i$.
4. $\beta_{ip} = 3p + i - 3$.

One easily verifies that this DR defines the 3D points x. In all, $3P + 6$ parameters are used, $3P$ for the values v and 6 for the three directions (two for each, since they have unit norm).

Representation of planar points : We now assume all points in x lie in the same plane: there exist a direction $\mathbf{f}$ and real number v_1 such that for all p, the relation $\mathbf{f}^\top \mathbf{x}_p = v_1$ holds. We assume without loss of generality that $\mathbf{f} \notin \{\mathbf{e}_2, \mathbf{e}_3\}$. A DR for x is be given by:

1. $d = (\mathbf{d}_1 = \mathbf{f},\, \mathbf{d}_2 = \mathbf{e}_2,\, \mathbf{d}_3 = \mathbf{e}_3)$.
2. $v = (v_1, x_{21}, x_{31}, x_{22}, x_{32} \ldots, x_{3P})$.
3. $\delta_{ip} = i$.
4. $\beta_{1,p} = 1,\ \beta_{2,p} = 2p$ and $\beta_{3,p} = 2p + 1$.

This DR uses $2P + 6 + 1$ parameters, to be compared with the $3P$ that are used when representing the x_p by their coordinates in a given basis.

Points on two parallel planes : We now assume that points in x lie in one of two parallel planes : there exist a direction $\mathbf{f}$ and real numbers v_1 and v_2 such that for all p, the relation $\mathbf{f}^\top \mathbf{x}_p \in \{v_1, v_2\}$ holds. We assume without loss of generality that $\mathbf{f} \notin \{\mathbf{e}_2, \mathbf{e}_3\}$ and that there is an index $P_1 \in \{2, \ldots, P\}$ such that $\mathbf{f}^\top \mathbf{x}_p = v_1$ if $p < P_1$ and $\mathbf{f}^\top \mathbf{x}_p = v_2$ otherwise. A DR for x is be given by :

1. $d = (\mathbf{d}_1 = \mathbf{f},\, \mathbf{d}_2 = \mathbf{e}_2,\, \mathbf{d}_3 = \mathbf{e}_3)$.
2. $v = (v_1, v_2, x_{21}, x_{31}, x_{22}, x_{32} \ldots, x_{3P})$.
3. $\delta_{ip} = i$.
4. $\beta_{1,p} = 1$ if $p < P_1$ and $\beta_{1,p} = 2$ otherwise. $\beta_{2,p} = 2p + 1$ and $\beta_{3,p} = 2p + 2$.

Here, $2P + 6 + 2$ parameters are used, versus the $3P$ that are used when representing the x_p by their coordinates in a given basis.

Points on parallel lines in a plane : We now assume that points in x lie on Q distinct parallel lines within a plane. The coplanarity is expressed by the existence of a direction $\mathbf{f}$ and real number v_0 such that for all p, the relation $\mathbf{f}^\top \mathbf{x}_p = v_0$. Alignment of some points is expressed by the existence of a direction $\mathbf{g} \neq \mathbf{f}$ and values $v'_1, \ldots, v'_Q$ such that for all p, $\mathbf{g}^\top \mathbf{x}_p \in \{v'_1, \ldots, v'_Q\}$ We assume without loss of generality that $\mathbf{f} \neq \mathbf{e}_3$ and $\mathbf{g} \neq \mathbf{e}_3$. A DR for x is given by :

1. $d = (\mathbf{d}_1 = \mathbf{f},\, \mathbf{d}_2 = \mathbf{g},\, \mathbf{d}_3 = \mathbf{e}_3)$.
2. $v = (v_0, v'_1, \ldots, v'_Q, x_{31}, x_{32} \ldots, x_{3P})$.
3. $\delta_{ip} = i$.
4. $\beta_{1,p} = 1$, $\beta_{2,p}$ takes values in $\{v'_1, \ldots, v'_Q\}$ according to the line it lies on. $\beta_{3,p} = Q + 1 + p$.

In this example, the number of parameters is $P + Q + 6 + 1$.

Based on these examples, one can easily find DRs for points on 2D or 3D grids. More generally, sets of points which contain subsets verifying coplanarity, parallelism and alignment relations, as well as their combinations can naturally be represented using constraints. This representation is compact and explicitly shows coplanarity relations (points that belong to the support of a common constraint), collinearity (points being in the support of two common constraints); the fact that points lie on parallel planes is expressed by points having a direction in common, but different constraints defined for that direction.

2.2 Application to optimization problems

We now show how to transform a problem in which a function $x \longrightarrow L(x)$ is minimized, into a problem in which $(d, v) \longrightarrow L(x(\Delta))$ is minimized. Popular algorithms such as conjugate gradient, only need to evaluate the function L and its differential $\frac{\partial}{\partial x}L$. The Levenberg-Marquardt algorithm, which minimizes a function of the form $L(x) = ||u - F(x)||^2$, only requires the computation of F and $\frac{\partial}{\partial x}F$. It is clear that in order to use one of these algorithms, we need to be able to compute $\frac{\partial}{\partial d}L$ and $\frac{\partial}{\partial v}L$[3] or, for Levenberg-Marquardt, $\frac{\partial}{\partial d}F$ and $\frac{\partial}{\partial v}F$. We note that

$$\begin{aligned} \frac{\partial}{\partial d}L &= \frac{\partial}{\partial x}L \cdot \frac{\partial}{\partial d}x \quad \text{and} \\ \frac{\partial}{\partial v}L &= \frac{\partial}{\partial x}L \cdot \frac{\partial}{\partial v}x. \end{aligned}$$

So, if the derivatives with respect to x can be computed, we only need, in order to compute the derivatives with respect to d and v, to be able to compute $\frac{\partial}{\partial v}x$ and $\frac{\partial}{\partial d}x$. The first derivative is trivial. For the second, we must specify a parameterization of a direction $\mathbf{d}_i$.

Ommiting the indice i, we call d_1, d_2, d_3 the coordinates of $\mathbf{d}$. We parameterize each direction $\mathbf{d}$ by three parameters $\alpha = (\alpha_1, \alpha_2, \alpha_3)$: $d_i = \alpha_i / \sqrt{\alpha_1^2 + \alpha_2^2 + \alpha_3^2}$, and write

[3] For convenience, we consider in this section $\mathbf{x}$, d and v as elements in R^{3P}, R^{3D} and R^V. $\frac{\partial}{\partial d}x$ is thus a $3P$-by-$3D$ matrix etc.

$\mathbf{d} = \mathbf{d}(\alpha)$. Note that this representation is redundant and present the advantage of being non-singular. Within the optimization algorithm, we will enforce the relation $\alpha_i = d_i$ at each step, by performing reprojection of the parameters. Because $\mathbf{d}$ must verify $||\mathbf{d}|| = 1$, one easily sees that

$$\frac{\partial}{\partial \alpha}\mathbf{d} = \mathbf{I}_3 - \mathbf{d}\mathbf{d}^\top. \tag{1}$$

As can be expected, this matrix has rank 2. If $\mathbf{d}$ is constrained to be orthogonal to another direction $\mathbf{d}$' of the dual representation, one has

$$\frac{\partial}{\partial \alpha}\mathbf{d} = \mathbf{I}_3 - \mathbf{d}\mathbf{d}^\top - \mathbf{d}'\mathbf{d}'^\top. \tag{2}$$

If $\mathbf{d}$ is subject to two orthogonal constraints, $\mathbf{d}^\top\mathbf{d}' = \mathbf{d}^\top\mathbf{d}'' = 0$, one has,

$$\frac{\partial}{\partial \alpha}\mathbf{d} = 0 \tag{3}$$

Here, $\mathbf{d}$ does not depend at all on α, which can be excluded from the optimization problem.

We also need the derivatives of a direction $\mathbf{d}$ that is constrained to be orthogonal to another direction $\mathbf{d}' = \mathbf{d}(\alpha')$:

$$\frac{\partial}{\partial \alpha'}\mathbf{d} = -\mathbf{d}'\mathbf{d}^\top. \tag{4}$$

Finally, we need the derivative of the point $\mathbf{x}$, defined by $\mathbf{x} = M^{-1}\mathbf{b}$ with respect to the columns of M. It is sufficient to give the derivative with respect to the i^{th} column of M, which we call $\mathbf{d}$:

$$\frac{\partial}{\partial \mathbf{m}}\mathbf{x} = \left(\tilde{\mathbf{d}}^\top\mathbf{b}\right) M^{-1}, \tag{5}$$

where $\tilde{\mathbf{d}}$ is the i^{th} column of M^{-1}.

Using equations (1-5) one can easily convert an optimization problem over x into an optimization problem over (d, v). Clearly, we can convert an optimization problem over parameters (x, θ) into a problem of optimization over (d, v, θ); in the case of 3D reconstruction, θ will represent the camera positions and calibration parameters.

3 Experimentation

In this section, we illustrate the compactness of the dual representation and how it improves the precision of 3D reconstruction. Figure 2 (a) shows one of the input images that were used. Point matches are available, along 12 images, for each of the 48 corners of the calibration grid. ML estimates of the 3D points were computed, using three different estimators. Their respective outputs are labeled **IC**, **DRC** and **DROC**:

IC Coordinates of points are estimated independently.

DRC A dual representation models the coplanarities in the scene.

DROC A dual representation models the coplanarities and orthogonalities in the scene.

All estimators use the perspective camera model, with unknown but constant intrinsic parameters. Five unknown intrinsic parameters are estimated : aspect ratio and skew, the principal point and the focal length.

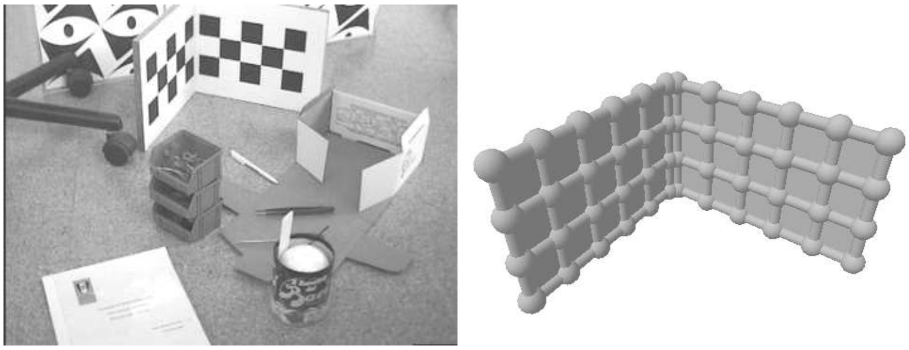

Figure 2: (a) First input image (b) A dual representation

Figure 2 (b) shows the dual representation corresponding to the third estimator. The association of points to planes (constraints) was done automatically by an algorithm that we do not discuss here. Points are represented as spheres, alignment relations are represented as cylinders between points and the two main planarity relations are represented as opaque surfaces. In all, 3 orthogonal directions and 18 values are used : 4 along the vertical direction, and 7 along each of the horizontal directions. In contrast, 144(=48$\times$3) values are needed to represent the same set of points by their coordinates.

We estimate the covariance of the three ML estimators using the techniques described in [8, 7] with the notable difference that we estimate the covariance of the noise of the observations rather than assume that it is known beforehand.

Table 1 shows the estimated standard deviation of the three considered ML estimators.

	Estimator	**IC**	**DRC**	**DROC**
Estimated quantity	Scale	Standard Deviation of Error		
3D Point Coordinates	1	0.0310	0.00715	0.00261
Camera Orientation	degrees	0.516	0.271	0.264
Camera Position	7.05	0.778	0.214	0.175
Logarithm of Focal Length	1.5	0.0491	0.0158	0.0135

Table 1: Standard deviation of the Maximum Likelihood estimators. The "Scale" column contains the mean abs. value of the estimated quantity, except for the "camera orientation" row, where error is given in degrees.

Table 1 shows that the standard deviation of the dual representation estimators is always significantly lower than that of the ML estimator of independent point coordinates. One should also note that the estimation of all parameters, including the orientation of cameras, their positions and focal length, is improved.

We now present empirical error measures, obtained from synthetic data : a virtual calibration grid and camera setup was used to produce virtual point matches. Iid. Gaussian noise was added. The amplitude of the noise was one-hundredth of the mean squared value of the observations, corresponding to a SNR of 40DB. The value of estimators was computed from these synthetic observations. This experience was run 50 times. The error, measured by the difference between the synthetic calibration grid and the output of the estimators, is shown in Table 2.

	Estimator	**IC**	**DRC**	**DROC**
Estimated quantity	Scale	$\sqrt{\text{Mean Squared Error}}$		
3D Point Coordinates	1	0.0316	0.0132	0.00409
Camera Orientation	degrees	0.458	0.332	0.250
Camera Position	7.05	0.727	0.283	0.195
Logarithm of Focal Length	1.5	0.0516	0.0277	0.0159

Table 2: Empirical mean absolute value of error of the Maximum Likelihood estimators. The "Scale" column contains the mean abs. value of the estimated quantity, except for the "camera orientation" row, where error is given in degrees.

Table 2 shows that the error of the dual representation estimators (**DRC** and **DROC**) is significantly lower than that of the **IC** estimator. This result is in accordance with the results of [1, 11]. In particular, this confirms the statement, in [11], that using orthogonality constraints greatly improves the precision.

The theoretical (table 1) and empirical (table 2) error measures clearly agree with each other.

4 Conclusions and future work

We have presented a representation for 3D points that is simple and expressive. We have shown how this representation can be used in the framework of ML estimation. The dual representation intrinsically contains geometric properties that are known to [1, 11] to be beneficial to the precision. We have confirmed this improvement, both by the theoretical study of the covariance of the estimators, and by empirical measure of error.

The dual representation could be further improved : relations between the "values" part of the DR could be identified. Namely, we are interested in recognizing when distances are equal, and thus further reducing the number of continuous parameters used in the representation. Also, it may be possible to encode camera positions and orientation.

Automatically associating points to planes is a challenging task. Because 3D points in the real world are never exactly coplanar, model selection is an important issue. We are currently working on these problems and plan to address them in a future publication.

References

[1] Didier Bondyfalat and Sylvain Bougnoux. Imposing euclidean constraint during self-calibration processes. In *Proc. SMILE Workshop*, pages 224–235, Freiburg, Germany, June 1998.

[2] Ray M. Bowen and C.-C. Wang. *Introduction to vectors and tensors*, volume 1. Plenum Press, 1976,1980.

[3] R. A. Brooks. Symbolic reasoning among 3-d models and 2-d images. *Artificial Intelligence*, (17):285–348, 1981.

[4] O. Faugeras. *Three Dimensional Computer Vision*. MIT Press, 1993.

[5] Arthur Gelb, editor. *Applied Optimal Estimation*. M.I.T Press, 1974.

[6] U. Grenander. *Pattern Synthesis: Lectures in Pattern Theory*, volume 18 of *Applied Mathematical Sciences*. Springer-Verlag, 1976.

[7] Etienne Grossmann and José Santos-Victor. The precision of 3d reconstruction from uncalibrated views. In *proc. BMVC*, volume 1, pages 115–124, 1998.

[8] R.M. Haralick. Propagating covariance in computer vision. In *Proc. Workshop on Performance Characteristics of Vision Algorithms*, pages 1–12, Cambridge, April 1996.

[9] D. Koller, Kostas Daniilidis, and Hans-Hellmut Nagel. Model-based object tracking in monocular image sequences of road traffic scene. *ijcv*, 10(3):257–283, 1993.

[10] Gunnar Sparr. Euclidean and affine structure/motion for uncalibrated cameras from affine shape and subsidiary information. In *Proc. SMILE Workshop*, pages 187–207, Freiburg, Germany, June 1998.

[11] Richard Szeliski and P.H.S Torr. Geometrically constrained structure from motion: points on planes. In *Proc. SMILE Workshop*, pages 171–186, Freiburg, Germany, June 1998.