The Estimation of Geometry and Motion of a Surface from Image Sequences by Means of Linearization of a Parametric Model

M. J. KORSTEN and Z. HOUKES

Control Systems and Computer Engineering Laboratory,
Faculty of Electrical Engineering, University of Twente,
P.O. Box 217, 7500AE Enschede, The Netherlands

Received August 13, 1987; accepted May 5, 1989

A method is given to estimate the geometry and motion of a moving body surface from image sequences. To this aim a parametric model of the surface is used, in order to reformulate the problem to one of parameter estimation. After linearization of the model standard linear estimation methods can be used to estimate the parameters. The main contribution of this paper is that a method is provided to perform the linearization without specifying the model. Therefore structure from motion estimation and nonrigid body motion estimation can be performed regardless of the model.

1. INTRODUCTION

This paper is concerned with the extraction of 3D information about a scene from images. The process of information extraction can be divided into two parts: recognizing the objects and determining their properties, expressed in parameters. To prevent the description from being too complicated a set of rather simple primitives is defined each having a set of parameters. Because of these parameters a primitive represents a class of specimen bodies, each described with the same parameters but with different parameter values. The primitives are defined according to a physical model of the bodies in the scene. Next the primitives are analyzed separately and combined at a higher level of analysis. From this discussion there arise two problems treated in this paper with respect to the primitives: their definition and the estimation of their parameters. Our approach is to address these problems in a systematic way in order to obtain a number of flexible algorithms. The foundation is formed by parameter estimation theory. Here an optimal estimation of a set of parameters is computed from a series of measurements and a theoretical relation between measurements and parameters. In our case the measurements are a set of grey values in a digital image. Especially the idea of linearizing a nonlinear parameter estimation problem around a previous guess is utilized. We therefore obtain a considerable freedom in our modeling. Modeling the shape and the motion of a body is performed in a body coordinate system defined suitably to keep the models simple. Next the models are transformed by means of a coordinate transformation to get a description of the scene from the camera point of view. In this paper no physical model for the imaging process is used. Instead a previously recorded image is transformed geometrically according to models for the shape and the motion of a primitive to get a prediction of the set of grey values used as measurements. Thus we arrive at a class of algorithms for motion and structure from motion estimation. However, we want to emphasize that the method is in no
way limited to this class of algorithms. If a model describing the physics of the imaging process is added an explicit theoretical prediction is obtained for the grey values, thus allowing the analysis of single images. At the moment a paper is prepared about that topic. In this paper we will restrict ourselves to motion and structure from motion estimation. In Section 2 we introduce the basic concepts of parameter estimation theory and image analysis as an application. In Section 3, modeling and the transformation of models is treated. In Section 4 the linearization is worked out. In Section 5, we introduce the moving plane as an example, while in Section 6 some results from experiments are shown. In Section 7, conclusions are given.

2. PARAMETER ESTIMATION

In this section we consider the problem of estimating a set of parameters from a series of measurements. In 2.1, linearization is presented as a general method of solving nonlinear parameter estimation problems. In 2.2, an optimal solution is given while, in 2.3, image sequence analysis is considered as an application. In 2.4, a remark is made about the problem of the starting estimate.

2.1. Linearization of the Nonlinear Parameter Estimation Problem

Consider a relationship between measurements \( \theta_i \) and certain parameters \( \alpha_j \). Collecting the measurements in a vector \( \theta \) and the parameters in a vector \( \alpha \) we have

\[
\theta = \theta(\alpha).
\]  

(2.1)

The problem is to estimate the parameter vector \( \alpha \) from the measurement vector \( \theta \). No general applicable solution exists for this problem unless relationship (2.1) is linear. For the linear case linear estimation theory furnishes solutions based on the least squares criterion. A well-known approach is therefore to define an intermediate set of parameters, being linearly related to the measurements. See, for instance, the "essential parameters" in [17, 18, 21, 22]. The conversion of these intermediate parameters to the desired parameters \( \alpha_j \) is then a second problem that is not completely trivial. A drawback of this class of methods is that they depend heavily on the underlying model being used. Therefore the analyses must be repeated each time a new model is introduced. Besides, there exists a possibility that certain models cannot be put into such a framework. The method used in this paper is to linearize the problem around a previous guess, thus creating a linear problem from a nonlinear one. Every model can be put into this framework. A requirement is that we have some idea of the parameter values so that a starting estimate \( \hat{\alpha} \) for the parameter vector can be posed. On the basis of \( \hat{\alpha} \) we can give a prediction \( \hat{\theta} = \theta(\hat{\alpha}) \) of the measurement vector. Expanding \( \theta(\alpha) \) in a Taylor series around \( \hat{\alpha} \) we have

\[
\theta(\alpha) - \theta(\hat{\alpha}) = \frac{\partial \theta(\alpha)}{\partial \alpha} \bigg|_{\alpha=\hat{\alpha}} (\alpha - \hat{\alpha}) + \text{(higher order terms)}. \]  

(2.2)

The measurement \( \theta \) is \( \theta(\alpha) \) and the prediction \( \hat{\theta} \) is \( \theta(\hat{\alpha}) \). The Jacobian matrix \( \frac{\partial \theta(\alpha)}{\partial \alpha} \) represents the derivative of the measurement vector \( \theta \) with respect to the parameter vector \( \alpha \). After definition of the quantities,
FIG. 1. Scheme for parameter estimation by means of linearization around a previous guess.

\[
\begin{align*}
\delta \theta &= \theta(\alpha) - \theta(\hat{\alpha}) \\
\delta \alpha &= \alpha - \hat{\alpha} \\
B &= \frac{\partial \theta(\alpha)}{\partial \alpha} \bigg|_{\alpha=\hat{\alpha}}.
\end{align*}
\]

(2.2) can be converted to

\[
\delta \theta = B \delta \alpha + \mathbf{n}.
\]

The additive noise term \(\mathbf{n}\) takes into account the measurement noise and the fact that the higher order terms in the Taylor series expansion are neglected. (2.3) is the classical form for the linear parameter estimation problem as presented, for instance, by Liebelt [13]. The linearized form is used there to solve the nonlinear orbit determination problem. Note that the "measurement" \(\delta \theta\) is the difference between the real measurement and a prediction. An implementation can be performed according to the scheme of Fig. 1.

2.2. Optimal Estimation of the Parameter Vector

If the noise process \(\mathbf{n}\) has zero mean and a diagonal second moment matrix \(C_n\) and nothing is known a priori about \(\delta \alpha\) an unbiased least squares estimator for \(\delta \alpha\) is

\[
\delta \alpha = (B^TB)^{-1}B^T\delta \theta.
\]

Problems with respect to a bias in the estimation are discussed in [13]. The singular value decomposition (SVD) [12] detects the existence of small or zero eigenvalues of the matrix \((B^TB)\). Thus parameters, that cannot be estimated are excluded from the inversion process and a pseudo inverse of \(B\) is obtained. In an iterative estimation process some parameter starting values might be outside the convergence range because of the values of other parameters. The SVD can now also be applied to prevent these parameters from running away. The use of the SVD requires, however, knowledge of the properties of the model parameters in order to be able to interpret the estimation results. To eliminate the necessity of a matrix inversion the Kalman form [13] can be used in which the differences \(\delta \theta_i\) are processed recursively. To conclude, we remark that specific knowledge about the noise can be utilized leading
to more complicated versions of (2.4) [13]. Questions with respect to convergence are considered in [1, 2, 5, 10, 11]. In this paper we will restrict ourselves to the use of the classical least squares form as the objective here being the design of models within the framework of estimation theory rather than the analysis of estimation methods themselves.

2.3. Estimating Parameters from Image Sequences

Next, parameter estimation theory is applied to the analysis of images. The scene being imaged consists of bodies of which the surface is visible. The measurements are the image grey values \( I(V_i) \) at several positions \( V_i \) in the image. If an image sequence is available a prediction of the image grey values can be determined from an image recorded previously if something is known about the temporal behavior of the image points. Here an image point is the projection of a point of the body surface. The position of this projection is \( V' \) at time \( t \) and \( V \) at time \( t + \Delta t \). Assume then that there is a relation between these positions:

\[
V' = V'(V; \alpha),
\]

where \( V' \) is given as a function of \( V \) and of a set of parameters collected in \( \alpha \). To find the function \( V'(V; \alpha) \) in (2.5) a model for the shape and the motion of the surfaces and of the imaging geometry is required. This problem will be worked out in Section 3. The “constant brightness” assumption states that the grey value of an image point is a constant of the time. Changes in grey values thus only occur because of displacements of the image points. This results in a prediction of grey values in the second image from grey values in the first image according to the relation:

\[
I_2(V) = I_1(V'),
\]

where \( I_1 \) is a grey value at time \( t \) and \( I_2 \) a grey value at time \( t + \Delta t \). This leads to the estimation algorithm of Fig. 2 in which the parameters from (2.5) are estimated. We still need an expression for the Jacobian matrix \( B \) appearing in (2.3), to be used

![Fig. 2. Scheme for parameter estimation applied to motion. \( \theta_i \) is a vector formed by the image grey values at several positions \( V_i' \): \( \theta_i = [I_1(V_i'), I_2(V_i'), \ldots]^T \). Analogous remarks for \( \theta_2 \) and \( \delta \theta \).](image-url)
by the estimator, where

\[ B_{ij} = \frac{\partial \theta_i}{\partial \alpha_j} = \frac{\partial I_2(V_i; \alpha)}{\partial \alpha_j} \bigg|_{\alpha = \hat{\alpha}} \]  \hspace{1cm} (2.7)

Because we do not use an analytical prediction for \( I_2(V) \) we do not have an analytical expression for the derivative (2.7). However, with the help of (2.6) and of the gradient to be estimated from the images, (2.7) can be written as

\[ B_{ij} = \frac{\partial I_1(\hat{V}_i')}{\partial \alpha_j} = \nabla I_1(\hat{V}_i') \frac{\partial V'(V_i; \alpha)}{\partial \alpha_j} \bigg|_{\alpha = \hat{\alpha}} \]  \hspace{1cm} (2.8)

with

\[ \hat{V}_i' = V'(V_i; \hat{\alpha}), \]

a position in the first image, determined with (2.5) and an estimated \( \hat{\alpha} \).

Substituting (2.8) in (2.2) results in:

\[ \delta \theta_i = \delta I(V_i) = I_2(V_i) - I_1(\hat{V}_i') = \nabla I_1(\hat{V}_i') \frac{\partial V'(V_i; \alpha)}{\partial \alpha} \bigg|_{\alpha = \hat{\alpha}} \delta \alpha. \]  \hspace{1cm} (2.9)

The derivative of \( V'(V_i; \alpha) \) to \( \alpha \) is computed in Section 4.

The situation is not completely ideal. Because of the noise term \( n_1 \) appearing in Fig. 2 the prediction \( \hat{\theta}_2 \) is corrupted with noise. Both noise terms \( n_1 \) and \( n_2 \) can be combined to a noise term \( n \) having a double variance. However, also the gradient image is corrupted with noise, being in contradiction with the requirement, that the matrix \( B \) is known exactly. See, however, Golub [7] for discussions about a transformation matrix that is not perfectly known.

2.4. Closed Form and Iterative Solutions

To conclude this section we pay some attention to the connection between closed form and iterative solutions. Closed form solutions are obtained by expanding the grey value function in a Taylor series with respect to position and time [9]:

\[ I(V + \delta V, t + \delta t) = I(V, t) + \frac{\partial I}{\partial V} \delta V + \frac{\partial I}{\partial t} \delta t + \text{(higher order terms)}. \]  \hspace{1cm} (2.10)

Together with

\[ I(V + \delta V, t + \delta t) = I(V, t) \]  \hspace{1cm} (2.11)
(constant brightness assumption), we obtain in the limit if $\delta t \to 0$,

$$\frac{\partial I}{\partial V} \frac{\partial V(t; \alpha)}{\partial t} = -\frac{\partial I}{\partial t},$$

(2.12)

where now $\partial V(t, \alpha)/\partial t$ represents the flow velocity, depending on the time $t$, but also on several parameters, describing the shape of the registered body and the imaging geometry, collected in a vector $\alpha$. Analytical expressions for the parameters from the flow velocity field yield together with (2.12) a closed form solution for the parameters from the image gradients. If they are not available an interactive scheme is necessary, expanding $\partial V(t; \alpha)/\partial t$ in a second Taylor series around a guess for $\alpha$. (2.2) performs both expansions at once. If the higher order terms in (2.10) cannot be neglected, the solution breaks down. The damage can be repaired by incorporating these higher order terms explicitly in (2.10). However, then higher order derivatives of the image grey value function must be estimated. This requires the Taylor series expansion to apply in a larger environment and the problem repeats itself. An alternative is to modify (2.11) to

$$I(V + \Delta V, t + \delta t) = I(V, t),$$

(2.13)

where $\Delta V(V; \hat{\alpha})$ now may be considered as an improvement of an estimate $\Delta V(V; \hat{\alpha})$ of the displacement. If $\Delta V(\cdot)$ is small enough to neglect the higher order terms, the left hand side of (2.13) can be expanded in a Taylor series around $V + \Delta V$ to obtain

$$I(V, t) - I(V, t + \delta V(V; \hat{\alpha}), t + \delta t) = \frac{\partial I(V')}{\partial V'} \bigg|_{V' = V + \Delta V(V; \hat{\alpha})} \delta V(V; \hat{\alpha})$$

$$= \frac{\partial I(V')}{\partial V'} \bigg|_{V' = V + \Delta V(V; \hat{\alpha})} \frac{\partial \Delta V(V; \hat{\alpha})}{\partial \alpha} \delta \alpha. \tag{2.14}$$

Comparing (2.14) with (2.10) it is observed, that no expansion is performed with respect to the time. Thus a displacement field is obtained instead of an optical flow field and $\delta t$ does not necessarily have to be small. Furthermore, the expansion in the spatial domain is around $V + \Delta V(\cdot)$, where $\Delta V(\cdot)$ may be considered as an initial estimate for the displacement, to be improved with $\delta V(\cdot)$. Finally, $\Delta V(\cdot)$ must be known beforehand. We have arrived at an iterative solution; this is so even if $\Delta V(\cdot)$ is available, and $\delta V(\cdot)$ are known as a function of the parameters. Closed form solutions are only possible if the differential form (2.12) applies. Alternatively it can be stated that (2.10) gives rise to iterative algorithms with the initial displacement estimate $\Delta V(\cdot)$ being zero and converging in one iteration. All this is a consequence of the fact that (2.12) is itself a linearization, neglecting the higher order terms in the Taylor series expansion (2.10). A last remark is, that in our own treatment, positions are computed instead of displacements according to

$$V' = V'(V; \alpha) = V + \Delta V(V; \alpha) \tag{2.15}$$

and (2.14) can be easily transformed to (2.9) with (2.15). The reason for computing
positions is just that they fit better into our framework of transformations between coordinate systems.

3. MODELING THE SCENE AND THE IMAGING PROCESS

In this section a framework is presented in which modeling of the scene and the imaging process can be performed. The aim is to obtain an expression (2.5), giving an image position as a function of another position in a later registered image. With this expression and with (2.6), predictions for image grey values $I'(V)$ in this later registered image can be obtained. The following topics arise:

—coordinate systems
—models for the shape and the motion of a surface
—the displacement of the image points.

3.1. Coordinate Systems

It is assumed that the scene consists of one body. It will be considered as a surface, because only the outside is visible. This leads to a so called 2½D description [16]. The estimation of its shape and motion can be computed from a series of at least two images using a parametric model. The problem has been addressed already by a number of authors [1–4, 10, 14, 17–23]. A variety of models is furnished. Most authors restrict themselves to the motion of a rigid plane. However, Tsai and Huang [22] treat the case of a curved surface while Chen and Penna [4] even relax the assumption of rigidity. Weng, Huang, and Ahuja [23] give an analysis from the mechanical point of view resulting in the local constant angular momentum (LCAM) model. The idea in this paper is to separate the phases in which modeling takes place so that if one part of the model is changed the rest can be maintained. To this end we use an estimation algorithm that can handle all linear and nonlinear parameter estimation problems, see Section 2 of this paper. Furthermore, several coordinate systems are introduced to facilitate modeling. They are sketched in Fig. 3:

—3D body coordinates $(X_1, X_2, X_3)$ in which the shape and the motion of the surface can be modeled and that can be chosen suitably to facilitate modeling of the body,

—3D camera coordinates $(U_1, U_2, U_3)$ defining the position of the camera with respect to the body. In this paper the concept of a pinhole camera is used. The origin of the camera coordinate system is the perpendicular projection of the pinhole into the image plane while the ground plane $(U_1, U_2)$ parallels the image plane,

—2D image coordinates $(V_1, V_2)$ in which the image can be described as an intensity function $I(V) = I(V_1, V_2)$. The origin of the image coordinates equals the origin of the camera coordinates.

In this paper the coordinate systems are taken Cartesian although this is not an essential requirement of the estimation method to work.

Homogeneous coordinates. The use of several coordinate systems requires an easy method to transform between coordinate systems (Table 1). To this end we use homogeneous coordinates as introduced in [6]. In homogeneous coordinates a
The transformation between Cartesian coordinate systems is represented as a multiplication of the four-dimensional coordinate vector with a transformation matrix $M$. In this paper Cartesian coordinates are written with upper case characters and their homogeneous counterparts, with lower case ones. Thus to the Cartesian body coordinates $(X_1, X_2, X_3)$ the homogeneous system $(x_1, x_2, x_3, x_4)$ is joined. Analogously, the homogeneous coordinates $(u_1, u_2, u_3, u_4)$ are joined to the Cartesian camera coordinates $(U_1, U_2, U_3)$. The unity vectors in homogeneous coordinates are depicted by $e_i$, $i = 1$ to 4, while the homogeneous identity matrix is $I$.

**Transformations between coordinate systems.** The perspective projection of a surface into an image plane can be performed by the two consecutive transformations from body to camera coordinates and from camera to image coordinates. The transformation between the body and camera coordinate systems is invertible and can be written as a cascade of the following transformations:

- a translation moving the origin of the coordinate system to the pinhole,
- a rotation such that the optical axis of the camera coincides with the $U_3$ axis of the camera coordinates,
- a translation along the optical axis over the camera focal length $f$, so that the origin of the coordinate system moves to the image plane,
- a deformation that combined with an orthographic projection yields the correct perspective projection.

The transformation between camera and image coordinates depicts the orthographic projection into the image plane and is not invertible. The perspective projection geometry is according to Fig. 4. The origin of the camera coordinates lies in the image plane at the projection of the pinhole.

**TABLE 1**

The Coordinate Systems Used in This Paper

<table>
<thead>
<tr>
<th>Coordinates</th>
<th>Cartesian</th>
<th>Homogeneous</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surface</td>
<td>$(X_1, X_2, X_3)$</td>
<td>$(x_1, x_2, x_3, x_4)$</td>
</tr>
<tr>
<td>Camera</td>
<td>$(U_1, U_2, U_3)$</td>
<td>$(u_1, u_2, u_3, u_4)$</td>
</tr>
<tr>
<td>Image</td>
<td>$(V_1, V_2)$</td>
<td></td>
</tr>
</tbody>
</table>
According to the geometry of the perspective projection the projection formulas in Cartesian coordinates are

\[ V_1 = \frac{fX_1}{f - X_3}, \quad V_2 = \frac{fX_2}{f - X_3}, \]  

(3.1)

just writing the projection part of the transformation and neglecting the rotation and the translations. In general, the transformation between homogeneous body and camera coordinates is given by

\[ u = Mx, \]  

(3.2)

where the transformation matrix \( M \) can be written as

\[
M = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & f \\
0 & 0 & -f^{-1} & 0
\end{bmatrix} \begin{bmatrix}
\cos \gamma & -\sin \gamma & 0 & 0 \\
\sin \gamma & \cos \gamma & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1
\end{bmatrix}
\times \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & \cos \psi & -\sin \psi & 0 \\
0 & \sin \psi & \cos \psi & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
\cos \varphi & 0 & \sin \varphi & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & l_1 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]  

(3.3)

which can be abbreviated to

\[ M = P \begin{bmatrix}
\phi & \phi l \\
\mathbf{e}_4^T
\end{bmatrix} \]  

(3.4)

with

\( \phi \) a three-dimensional rotation matrix, describing the orientation of the body coordinates with respect to the camera

\( \mathbf{l} \) a translation vector \((l_1, l_2, l_3)^T\)

\[
P = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & f \\
0 & 0 & -f^{-1} & 1
\end{bmatrix} \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & f \\
0 & 0 & -f^{-1} & 0
\end{bmatrix}.
\]
Thus $P$ is the product of the projection matrix [6] with the matrix representing the last transformation along the optical axis of the camera system.

3.2. Models for the Shape and the Motion of the Surface

Next we discuss the properties of the surface expressed by its shape and motion. Doing this we use homogeneous coordinates. The surface shape can, in general, be described by giving the points (in body coordinates) being part of it. An analytical description is obtained by requiring the coordinates of the surface points to obey the functional relation

$$f(x; \alpha) = 0 \quad \text{and} \quad x_4 = 1,$$

where the second condition arises from the definition of the homogeneous coordinates [6] and where the parameter vector $\alpha$ may contain shape parameters of the surface like the curvature. Equation (3.5) must be made explicit when a specific model has been chosen. This will be done in Section 5 for the case of the moving plane.

To incorporate motion we refer back to (2.5) and (2.6). From there it appears, that positions $V'$ are required at time $t$, given positions $V$ at time $t + \Delta t$. Let, therefore, the surface shape description (3.5) be given at the time $t + \Delta t$. The position of the body points at time $t$ can be obtained using the motion of the body. The motion model maps each point $x$ at time $t + \Delta t$ in space into another point $x'$ at time $t$ according to

$$x' = g(x; \alpha),$$

where the points $x$ are chosen such that they satisfy (3.5).

3.3. The Displacement of the Image Points

With the shape and motion models and the transformation from body to camera coordinates, expressed by the matrix $M$, the displacement of the image points can be computed. To ease notation the homogeneous vectors $v$ and $v'$ are introduced, with

$$v = (V_1, V_2, -f, 1)^T$$

$$v' = (V'_1, V'_2, -f, 1)^T.$$

Now all computations can be kept in homogeneous coordinates. Thus instead of looking for positions $V'$ obeying (2.5), homogeneous positions $v'$ are sought, obeying

$$v' = v'(v; \alpha),$$

$v'$ is the position in the first image of a point having the position $v$ in the second image. The general route to find $v'(v; \alpha)$ can be divided into the following steps:

1. Find the body points being projected at time $t + \Delta t$ in the image plane, described by the functional relation:

$$x = x(v; \alpha).$$

$x(v; \alpha)$ can be found by combining the imaging geometry, given by the matrix $M$ in (3.3), with the body shape description (3.5) (see Section 5).
II. Find the positions $x'$ of the body points at time $t$, using the motion model (3.6) from their positions $x$ at time $t + \Delta t$.

III. Find the positions $v' = v'(v; \alpha)$ of the projections of the body points at time $t$ from the imaging geometry.

Combining the three steps yields the required relation (3.8). We start with the inverse transformation from camera to body coordinates. In homogeneous coordinates we have

$$x = M^{-1} u = u_4 M^{-1} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ 1 \end{bmatrix}. \quad (3.10)$$

Now from (3.4) $M^{-1}$ can be written as

$$M^{-1} = \begin{bmatrix} \phi^{-1} & -I \\ e^T_4 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & f^-1 \\ 0 & 0 & f^-1 & 1 \end{bmatrix}. \quad (3.11)$$

From (3.11) the last row of $M^{-1}$ is $(0 \ 0 \ f^-1 \ 1)$, from which it can be deducted easily that

$$u_3 = f(x_4 - u_4). \quad (3.12)$$

Together with:

$$U_3 = u_3/u_4 \quad \text{and} \quad x_4 = 1,$$

$$U_1 = V_1 \quad \text{and} \quad U_2 = V_2,$$

it follows that

$$x = M^{-1}(u_4 v + f e_3) \quad (3.13)$$

and

$$u = (u_4 v + f e_3), \quad (3.14)$$

the inverse transformation between image and camera coordinates appearing as a byproduct. (3.13) is the inverse transformation from image to body coordinates. It is valid irrespective of the surface model and therefore valid at time $t$ and at time $t + \Delta t$. In this first case all position variables must be written with a prime ($'$). There is still one unknown, $u_4$, and it is there that we have to use knowledge about the surface shape and motion. That is a function $u_4(v; \alpha)$ must be found such that

$$u_4 = u_4(v; \alpha). \quad (3.15)$$
This knowledge depends on the specific model chosen. The determination of the set of points \( v' \) with the above-mentioned steps II and III is straightforward from here. The result is in homogeneous coordinates:

\[
v' = v'(v; \alpha) = \frac{1}{\alpha_4'(v; \alpha)}(\alpha'(v; \alpha) - f_e_3)
\]  

(3.16)

with

\[
\alpha'(v; \alpha) = Mx'
\]

(3.17)

\[
x' = g(x; \alpha) \quad \text{from (3.6)}
\]

\[
x = x(v; \alpha) \quad \text{from (3.9)}.
\]

The cause, that at this stage no explicit expression for \( v'(v; \alpha) \) can be given is, that the shape model does not give an explicit correspondence between surface and image points. However, it appears possible to put the variation of the prediction with \( \alpha \) in terms of the derivative of the shape function \( f(x; \alpha) \) and we will do so in Section 4.

3.4. Summary of Section 3

In Section 3, a framework has been presented enabling us to model the scene and the geometry of the imaging process. The various models deliver parameters. To

![Diagram](image)

**FIG. 5.** Predicting the second image, registered at time \( t + \Delta t \), from the first image, registered at time \( t \).
simplify notation we collect all these parameters in one parameter vector $\alpha$:

- parameters from the shape model of the body
- parameters from the motion model of the body
- parameters from the transformation model describing the position and orientation of the body with respect to a camera.

With these models the position $v'$ of an image point in a previously recorded image has been determined as a function of its position $v$ at a later moment. The steps to be taken are depicted in Fig. 5.

4. DETERMINING THE VARIATION OF THE ESTIMATED MOTION WITH THE PARAMETERS

In order to estimate the model parameters with the linearized estimator of Section 3, Eqs. (2.7) to (2.9), we need to compute the variation of the image position vector $V'(V; \alpha)$ with the parameters $\alpha_j$. In homogeneous coordinates an expression is required for

$$\frac{\partial}{\partial \alpha_j} v'(v; \alpha)$$

with $v'(v; \alpha)$ from (3.16) and (3.17).

We perform the computation such, that (3.9), being an explicit description of the body shape from the camera point of view, is replaced by the more general (3.5). Not only do we obtain a general formulation of the estimator in which at a later stage a freely chosen body shape description can be inserted. Also no differentiation of (3.9) is required, the differentiation of (3.5) suffices. This means, that a numerical solution of (3.9) from (3.5) and the imaging geometry might suffice during the implementation. The forms of (3.16) and (3.17) invite strongly to use a chain rule. To begin with (3.16) is differentiated with respect to $v'(\cdot)$ with the result,

$$\frac{\partial}{\partial \alpha_j} v'(v; \alpha) = \frac{1}{\omega'_2(v; \alpha)} \left( I - v'(v; \alpha)e_2^T \right) \frac{\partial \omega'(v; \alpha)}{\partial \alpha_j}, \quad (4.1)$$

where use has been made of the dyadic product $v'(\cdot)e_2^T$. We continue with the computation of

$$\frac{\partial \omega'(v; \alpha)}{\partial \alpha_j} = \left. \frac{\partial}{\partial \alpha_j} \{ M g(x; \alpha) \} \right|_{x=x(v; \alpha)}$$

$$= \left. \left\{ \frac{\partial M}{\partial \alpha_j} g(x; \alpha) + M \frac{\partial g(x; \alpha)}{\partial \alpha_j} + M \frac{\partial g(x; \alpha)}{\partial x} \frac{\partial x(v; \alpha)}{\partial \alpha_j} \right\} \right|_{x=x(v; \alpha)} \quad (4.2)$$

The last expression in (4.2) represents three variations:

1. the variation of the transformation between surface and camera coordinates
2. the variation of the motion model
3. the variation of the shape model.
The first two variations have been determined as soon as we have specified our model. With respect to the third one the derivative of \(x(v; \alpha)\) in (4.2) is eliminated in favor of the derivative of \(f(x; \alpha)\). We have from (3.9),

\[
f(x; \alpha) = 0. \tag{4.3}
\]

This provides a constraint for the surface points when a parameter \(\alpha_j\) is varied. Because \(f(x; \alpha)\) should still be zero after a variation of \(\alpha_j\) with an amount \(d\alpha_j\) we have

\[
df = \frac{\partial f(x; \alpha)}{\partial \alpha_j} d\alpha_j + \frac{\partial f(x; \alpha)}{\partial x} dx = 0, \quad x = x(v; \alpha) \tag{4.4}
\]

and we obtain

\[
\frac{\partial f(x; \alpha)}{\partial x} \frac{\partial x(v; \alpha)}{\partial \alpha_j} = -\frac{\partial f(x; \alpha)}{\partial \alpha_j}, \quad x = x(v; \alpha). \tag{4.5}
\]

(4.5) is just one scalar relation for the four components of \(\partial x(\cdot)/\partial \alpha_j\). Therefore, three other ones must be found from additional conditions. The first one is rather trivial and only an artifact from the use of homogeneous coordinates. Because \(x_4 = 1\) (3.5):

\[
\frac{\partial x_4(v; \alpha)}{\partial \alpha_j} = \frac{\partial}{\partial \alpha_j} \{e_4^T x(v; \alpha)\} = 0. \tag{4.6}
\]

The other two conditions stem from the following argument. Suppose a parameter \(\alpha_j\) is varied. Then the positions at the surface at \(t\) and at \(t + \Delta t\) will vary. And accordingly the position of the projection \(v'\) in the first image at time \(t\). However, the position of the projection \(v\) in the second image at time \(t + \Delta t\) should not vary, because it is an independent variable. The positions of the surface points are varied with the constraint that the positions of the projection in the second image stay at the same place. With

\[
V_1 = \frac{u_1}{u_4}, \quad V_2 = \frac{u_2}{u_4}
\]

and

\[
u = Mx = Mx(v; \alpha),
\]

it follows that

\[
\frac{\partial}{\partial \alpha_j} \left\{ e_4^T Mx(v; \alpha) \right\} = \frac{\partial}{\partial \alpha_j} \left\{ e_4^T Mx(v; \alpha) \right\} = 0. \tag{4.7}
\]
We now have four scalar relations for the variation of $x(v; \alpha)$, so that it can be eliminated in favor of $f(x; \alpha)$. Its explicit determination is rather lengthy and results in a rather lengthy formula (Appendix A). With (4.1) and (4.2) we finally obtain

$$
\frac{\partial}{\partial \alpha_j} v'(v; \alpha) = \frac{1}{\omega'_j(v; \alpha)} \left\{ I - v'(v; \alpha) e_i^T \right\}
$$

$* \left\{ \frac{\partial M}{\partial \alpha_j} g(x; \alpha) + M \frac{\partial g(x; \alpha)}{\partial \alpha_j} - M \frac{\partial g(x; \alpha)}{\partial x} M^{-1} \right\}$

$* \left\{ \frac{1}{k} \frac{\partial f(x; \alpha)}{\partial \alpha_j} v + \left\{ I - \frac{1}{k} \frac{\partial f(x; \alpha)}{\partial x} M^{-1} \right\} \frac{\partial M}{\partial \alpha_j} x \right\}$

$$
x = x(v; \alpha)
$$

$$
k = \frac{\partial f(x; \alpha)}{\partial x} \bigg|_{x=x(v; \alpha)} * M^{-1} v.
$$

where $I$ is the four dimensional identity matrix and $v \frac{\partial f(x; \alpha)}{\partial x}$ a dyadic product.

5. THE MOVING PLANE

We illustrate the use of our method with the example of the moving plane. The following topics will be discussed:

— the model for the shape and motion of the moving plane and the identification of the parameters,
— uniqueness,
— the problem which parameters of the model can be determined from a series of images.

5.1. The Model for the Shape and the Motion

The model for the moving plane requires a description of its shape and motion in surface coordinates and of the position of the system of surface coordinates with respect to camera coordinates. The description of the shape of the plane in surface coordinates can be kept rather simple. We choose the plane to coincide with the $X_1, X_2$-plane (Fig. 6). The consequence for the shape function is

$$
f(x) = x_3 = 0 \quad x = x(v; \alpha).
$$

The model for the shape does not contain parameters. Here the advantage of choosing smart body coordinates appears very clear.

As a model for the motion, the first two terms in the Taylor series expansion for the displacement have been chosen. This is a description of general nonrigid motion
in the approximation of small motion [15]. If we write in Cartesian coordinates,

\[ X = X + \Delta X(X), \]  

(5.2)

where \( \Delta X(X) \) is a position dependent displacement vector, we obtain

\[
\Delta X(X) = \Delta X(0) + \frac{\partial \Delta X(X)}{\partial X} \bigg|_{X=0} X + \cdots \\
= \Delta X(0) + (\Omega + S)X + \cdots, 
\]

(5.3)

where \( \Omega \) is the rotation matrix describing the rotation

\[
\Omega = \begin{bmatrix}
0 & -R_3 & R_2 \\
R_3 & 0 & -R_1 \\
-R_2 & R_1 & 0
\end{bmatrix} 
\]

(5.4)

and \( S \) is a scaling matrix, describing the nonrigid part of the motion,

\[
S = \begin{bmatrix}
S_{11} & S_{12} & S_{13} \\
S_{12} & S_{22} & S_{23} \\
S_{13} & S_{23} & S_{33}
\end{bmatrix}.
\]

(5.5)

To complete the motion model it is observed that, because \( x_3 = 0, S_{33} \) has no meaning and \( S_{23} \) and \( S_{13} \) cannot be discriminated from \( R_1 \) and \( R_2 \), respectively. They are consequently dropped from consideration, thus leaving nine motion parameters (\( \Delta X_1, \Delta X_2, \Delta X_3, S_1, S_2, S_{12}, R_1, R_2, R_3 \)). Here, for simplicity, \( S_{11} \) and \( S_{22} \) are indicated by \( S_1 \) and \( S_2 \), respectively. The result is, that only scaling parameters concerning scaling within the plane appear, which means that the plane stays a plane. The scaling parameters, deforming the plane to a nonplane body cannot be discriminated from rotation parameters in this model and will therefore not be considered. The motion model can be represented in homogeneous coordinates with the \( 4 \times 4 \) matrix \( D \) as

\[
g(x, \alpha) = Dx 
\]

(5.6)
SURFACE MOTION ESTIMATION

\[
D = \begin{bmatrix}
1 + S_1 & -R_3 + S_{12} & R_2 & \Delta X_1 \\
R_3 + S_{12} & 1 + S_2 & -R_1 & \Delta X_2 \\
-R_2 & R_1 & 1 & \Delta X_3 \\
0 & 0 & 0 & 1
\end{bmatrix}
\] (5.7)

which can be abbreviated to

\[
D = \begin{bmatrix}
I + \Omega + S & \Delta \mathbf{X} \\
\mathbf{e}_4^T
\end{bmatrix}.
\] (5.8)

with I the identity matrix. It should be noted, that the motion model is of course no general model for nonrigid body motion. It just describes nonrigid body motion in the case of a plane in the small motion approximation.

Position and orientation of the plane. The position and orientation of the plane with respect to the camera are given by the transformation matrix \( M \), where \( M \) is the product of a projection, a translation, and a rotation matrix. In the case of the plane, a rotation in the plane does not change the plane shape, so that only the rotations (around the \( X_1 \) and the \( X_2 \) axes) that tilt the plane with respect to the image plane are considered. Note that here rotations are considered describing the orientation of the plane. The rotations of the motion are given in (5.4) and (5.7) and here the rotation \( R_3 \) around the \( X_3 \) axis is important.

Then \( M \) can be written as

\[
M = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & f \\
0 & 0 & -f^{-1} & 0
\end{bmatrix} \begin{bmatrix}
\cos \varphi & 0 & \sin \varphi & 0 \\
0 & 1 & 0 & 0 \\
-\sin \varphi & 0 & \cos \varphi & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & l \\
0 & 0 & 0 & 1
\end{bmatrix}
\] (5.9)

where in comparison with (3.3) \( \phi \) does not contain the orientation angle \( \gamma \) with respect to the \( X_3 \)-axis anymore. The distance perpendicular to the plane from the plane to the pinhole is depicted by \( l \), while the normal on the plane has in Cartesian camera coordinates the direction:

\[
\mathbf{N} = \begin{bmatrix}
\sin \varphi \\
-\sin \psi \cos \varphi \\
\cos \psi \cos \varphi
\end{bmatrix}.
\] (5.11)
The set of parameters from the transformation thus is \((\psi, \varphi, l)\). The total parameter vector \(\alpha\) is found by including both motion and transformation parameters:

\[
\alpha = (\psi, \varphi, l, \Delta X_1, \Delta X_2, \Delta X_3, R_1, R_2, R_3, S_1, S_2, S_{12})^T. \tag{5.12}
\]

Thus \(\alpha\) in this case contains 12 parameters.

**Position of the image points.** Next we compute for the specific case of the moving plane the position of the image points at \(t, v'\), given their position at time \(t + \Delta t\), which is \(v\). To this end the information from the model is used to calculate \(\alpha_4(\cdot)\) and \(\alpha'_4(\cdot)\) in (3.13) and (3.16). We have

\[
f\{x(v; \alpha)\} = x_3(v; \alpha) = e_{3}^T x(v; \alpha) - e_{3}^T M^{-1}(\alpha_4(v; \alpha)v + f e_3)
\]

\[
= 0. \tag{5.13}
\]

It follows that

\[
\alpha_4(v; \alpha) = -f \frac{(M^{-1})_{33}}{e_3^T M^{-1} v}
\]

\[
\alpha'_4(\cdot) \tag{5.14}
\]

with \((M^{-1})_{33}\) the 3, 3 element of \(M^{-1}\).

\(\alpha'_4(\cdot)\) can be expressed as a function of \(\alpha_4(\cdot)\). Substituting the result in (3.16), together with (5.14), (3.17), and (3.14) we obtain

\[
v'(v; \alpha) = -f \frac{Av}{e_3^T A v} \tag{5.15}
\]

with \(A\) the matrix:

\[
A = (MDM^{-1} - I)e_3 e_3^T M^{-1} - (M^{-1})_{33}(MDM^{-1}). \tag{5.16}
\]

Thus \(A\) is the difference between a dyadic product and a full rank matrix.

To obtain the variation of this prediction with the parameters, the models (5.1) and (5.6) must be inserted into (4.8). First it is observed that, because the shape model does not contain parameters, the term with \(\partial f(x, \alpha)/\partial \alpha_j\) does not occur.

There remain two possibilities:

If \(\alpha_j\) is a parameter arising from the motion model \((\Delta X_1, \Delta X_2, \Delta X_3, R_1, R_2, R_3, S_1, S_2, S_{12})\), we have

\[
\frac{\partial v'(v; \alpha)}{\partial \alpha_j} = \frac{f}{e_3^T A v} \left( I - v'(v; \alpha)e_3^T \right) M \frac{\partial D}{\partial \alpha_j} M^{-1} \frac{(M^{-1})_{33}}{e_3^T M^{-1} v - e_3}. \tag{5.17}
\]
while if $\alpha_j$ is a parameter from the transformation model $(\psi, \varphi, I)$, we have

$$\frac{\partial \psi'(v; \alpha_j)}{\partial \alpha_j} = \frac{f}{e_3^T A v} \left( I - \psi'(v; \alpha_j) e_3^T \right)$$

$$* \left\{ - \frac{1}{e_3^T M^{-1} v} (MDM^{-1}) e_3^T \frac{\partial M^{-1}}{\partial \alpha_j} \right\}$$

$$- \left\{ \left( \frac{\partial M}{\partial \alpha_j} M^{-1} \right)(MDM^{-1}) - (MDM^{-1}) \left( \frac{\partial M}{\partial \alpha_j} M^{-1} \right) \right\}$$

$$* \left( (M^{-1})_{33} - e_3 \right).$$  \hspace{1cm} (5.18)

5.2. Uniqueness

Several authors have pointed out that a solution vector $\alpha$ is in general not unique [3, 14, 17, 18, 20–22]. This means that more than one parameter vector gives rise to precisely the same displacement field. For the case of rigid motion of the plane, an analytical treatment can be given from (5.15) which relates the position in the first image with that in the second image. Thus the problem is given

$$\psi'(v; \alpha) = -f \frac{A v}{e_3^T A v},$$

is there a matrix $A'$ such that

$$\psi''(v; \alpha) = -f \frac{A' v}{e_3^T A' v}$$

is the same as the above-mentioned $\psi'(\cdot)$ for all $v$?

The first observation is that $A$ can be multiplied freely with a constant. Apart from that we have

$$A v = A' v$$

or

$$(A - A') v = 0 \quad \text{for all } v.$$  \hspace{1cm} (5.19)

Thus $v$ lies in the kernel of $(A - A')$ and it is not difficult to prove that $(A - A')$ can be written as the dyadic product

$$A - A' = a(0 \quad 0 \quad 1 \quad f),$$  \hspace{1cm} (5.20)

because the second vector is the only one perpendicular to all $v$. $a$ is a vector that can be chosen freely. These results must be used to find relations between the
parameters. To this end we define the quantities

$$\Omega_c = \phi \Omega \phi^{-1}, \quad S_c = \phi S \phi^{-1}$$  \hspace{1cm} (5.21)

with \( \phi \) from (3.4), and

$$\Delta_c = \phi \left( \Delta X - l(\Omega + S) \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right).$$  \hspace{1cm} (5.22)

In (5.21) and (5.22) a coordinate transformation can be recognized so that \( \Omega_c, S_c, \) and \( \Delta_c \) are the rotation, stretching, and translation as seen from the pinhole position. In Appendix B the uniqueness relation is worked out:

$$\frac{1}{l} \Delta_c N^T - (\Omega_c + S_c + I) = \text{const.}$$  \hspace{1cm} (5.23)

If \( S = S_c \) is zero (rigid motion) this is the same result as obtained by Negahdaripour and Horn [17]. We then have only one solution for (5.23), apart from the depth \( l \), acting as a scaling factor for the translation. Thus apart from the scaling factor there are only two possibilities for \( \alpha \). The second solution for \( \alpha \) is written with primed symbols. If the elements of \( \Omega \) are small it follows that

$$N' = \Delta_c \| \Delta_c \| \quad \text{and} \quad \Delta'_c / \| \Delta'_c \| = N$$  \hspace{1cm} (5.24)

and

$$\Omega'_c = \Omega_c + \left( \Delta_c N^T - N \Delta_c^T \right) / l.$$  \hspace{1cm} (5.26)

Thus apart from the scaling factor we have, in both cases, precisely two solutions for \( \alpha \), well separated from each other in parameter space.

5.3. The Estimation of the Model Parameters

One can wonder if, in the more general case, both rotation and stretching occur there still exist uniqueness properties like (5.24). A maximum number of independent parameters can be found from (5.15) which in general can be written in the bilinear forms,

$$V_1' = \frac{a_1 V_1 + a_2 V_2 + a_3}{c_1 V_1 + c_2 V_2 + 1}$$

$$V_2' = \frac{b_1 V_1 + b_2 V_2 + b_3}{c_1 V_1 + c_2 V_2 + 1}.$$  \hspace{1cm} (5.27)

where \( a_i, b_i, c_i \) are the essential parameters [17, 18, 21, 22] related to the parameters \( \alpha_i \). From (5.27) it appears that no more than eight parameters can be estimated independently. Thus it is not possible to estimate all model parameters from (5.12)
at the same time. On the other hand, it might be possible to use such a model if some parameters are known, so that they do not have to be estimated. From the discussion above it appears that no more than eight parameters that occur in our model can be estimated. The next question is, which combinations of eight parameters can be estimated? For this the eight by eight matrix \((B^TB)\) from (2.4) that has been built up from these parameters must be regular and that depends on:

- the combination of parameters,
- the values of the parameters,
- the choice of the positions for the measurements \((V_i)\),
- the properties of the gradients that occur from signal properties.

Because the issue here is the choice of the parameter combination, we consider a smart choice for the positions and the parameters and favorable properties of the signal. How many measurements are necessary? It is not difficult to see that if there are less than eight measurements, \(B^TB\) will always be singular. Now suppose that no eight measurements can be found that make \(B^TB\) regular. Can more than eight measurements do? The answer is no as can be proved with a simple argument [8]. But in the case of eight measurements \(B\) is square and therefore the following conclusion results:

- a combination of eight parameters can be estimated iff there are eight positions \(V\) and eight gradients that make the square eight by eight matrix \(B\) regular.

Next we determine which combinations of eight parameters can be estimated. We start with rigid body motion and the uniqueness discussion in Section 5.2. It appears that the depth \(l\) is a scaling factor for the translation \(\Delta X\). Therefore every combination of parameters can be chosen as long as not all the translation parameters and the depth are part of it. To investigate the behavior of the nonrigid motion parameters, computer experiments have been carried out that gave clear results:

- If the eight by eight matrix \(B\) has been built up from parameters, among which \(\Delta X_1, \Delta X_2, \Delta X_3,\) and \(l\), \(B\) is always singular. If one of these parameters is missing \(B\) can be made regular.

Thus, indeed, most combinations of eight parameters can be estimated. One should, however, keep in mind that the regularity of \(B\) also depends on the actual value of the parameters used. For instance, if the plane happens to be parallel with the image plane it is not possible anymore to discriminate a shrink of the plane, described by \(S_1\) and \(S_2\), from a translation perpendicular to both planes \((\Delta X_3)\). But this problem just occurs from unfavorable values of some parameters.

To conclude, we remark that the full power of our method does not completely appear in this example. For the case of the plane it is also possible to calculate the \(\partial \nu'(v; \alpha)/\partial \alpha\) from the explicit expression (5.15). But if this expression becomes more complicated, the task of calculating the variation becomes difficult and the use of (4.8) is much easier. This method even works if there is no analytical expression for \(\nu'(v; \alpha)\) available.
6. EXAMPLES OF THE PERFORMANCE OF THE ESTIMATION ALGORITHM

To conclude this paper we give two examples of the performance of the algorithm as implemented on our image processing system. Again the example considered is of the moving plane on which a sine pattern has been applied,

\[ I(V) = 63 \sin \left( \frac{V_1 \pi}{25.6} \right) + 63 \sin \left( \frac{V_2 \pi}{25.6} \right) + 128, \]

where \( V \) is the image coordinate (Fig. 7).

For the estimation experiments in all cases, a window of 64 by 64 pixels was used. The first experiment concerns pure (nonrigid) motion estimation. The shape parameters are assumed known, the plane being parallel with the image plane. Thus \( \psi = \varphi = 0 \). Besides we assume known: \( \Delta X_1 = 0, f = 50p, l = 150p \), where “p” stands for “pixel,” the unit of length in this experiment. The parameters, to be estimated are \( \Delta X_1 = 25p, \Delta X_2 = 15p, R_1 = -0.1 \text{ rad}, R_2 = 0.1 \text{ rad}, R_3 = 0.2 \text{ rad}, S_1 = 0.1, S_2 = -0.2, S_{12} = 0.2 \). The second image was generated from the test image according to the model, so that no model errors could occur. The results from the motion estimation process are shown in Fig. 8. The convergence appears to be very rapid; in this case stable values are obtained after about 4 iterations.

The second example concerns structure from motion and is also an example of rigid body motion. The parameters, assumed known, are: \( S_1 = S_2 = S_{12} = 0, f = 50p, l = 150p \).

Duality. As we have seen in Section 5 there are two solutions for the parameter vector \( \alpha \) as soon as parameters are estimated giving the orientation of the plane, like the transformation angles \( \psi \) and \( \varphi \). The parameter values and their duals are as in Table 2. As can be seen from the table the dual solution for \( \Delta X_3 \) is the same as the original one. This is a result valid in general, as can be proved analytically. Which solution is reached by the estimator depends on the starting estimate. We give two examples. In the first example, the starting estimate is: no motion, \( \varphi = -1.0 \text{ rad} \),

Fig. 7. Test images: (a) the original image; (b) the image after the motion of the surface (first example).
Fig. 8. Estimation of nonrigid surface motion parameters.

Table 2
Structure from Motion: Parameter Values and Their Dual Values

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Real value</th>
<th>Dual solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\psi$</td>
<td>0.40 rad</td>
<td>-0.96 rad</td>
</tr>
<tr>
<td>$\varphi$</td>
<td>-0.30 rad</td>
<td>0.16 rad</td>
</tr>
<tr>
<td>$\Delta X_1$</td>
<td>25.00 p</td>
<td>8.49 p</td>
</tr>
<tr>
<td>$\Delta X_2$</td>
<td>15.00 p</td>
<td>-30.49 p</td>
</tr>
<tr>
<td>$\Delta X_3$</td>
<td>7.00 p</td>
<td>7.00 p</td>
</tr>
<tr>
<td>$R_1$</td>
<td>0.20 rad</td>
<td>-0.09 rad</td>
</tr>
<tr>
<td>$R_2$</td>
<td>0.10 rad</td>
<td>0.16 rad</td>
</tr>
<tr>
<td>$R_3$</td>
<td>-0.10 rad</td>
<td>0.12 rad</td>
</tr>
</tbody>
</table>

Fig. 9. (a) Structure from motion estimation: convergence to the real solution; (b) Convergence to the dual solution.
\( \psi = 1.0 \text{ rad} \), and the estimator converges to the real solution. In the second example the starting estimate is: no motion, \( \varphi = 1.0 \text{ rad} \), \( \psi = -1.0 \text{ rad} \), and the estimator converges to the dual solution. In the case of the real solution, the end result was reached after about eight iterations (Fig. 9a). In the case of the dual solution it was reached after ten iterations (Fig. 9b). The convergence properties are now becoming more complicated. If there is not enough motion, the shape cannot be estimated; this suggests a tendency to instability of the algorithm. Of course, this is a result that could be expected from a structure from the motion algorithm. We will not elaborate on the convergence properties here, because they are still the subject of research in our group.

7. CONCLUSION

Starting from the theory of parameter estimation we presented a method for the analysis of images. To this end primitives were defined and their parameters were estimated from a set of image grey values. The method is flexible in two ways. In the first place several primitives can be exchanged in a simple way without the need for a new estimation formalism. This is because the various phases in the modeling and the estimation are well separated. In the second place, the method allows motion and structure from motion estimation, but also single image analysis is possible if a physical model for the imaging process is added. In this paper a moving plane was used as an example of a primitive on which motion and structure from motion estimation were performed. It was shown that for the case of the rigid plane the well-known duality result can be deduced. Also it was shown that as long as the constant brightness assumption is utilized no real closed form solutions for the parameter estimation problem exist.

APPENDIX A

In this appendix an expression is computed for \( \frac{\partial x(v; \alpha)}{\partial \alpha_j} \), which is required in (4.2). We differentiate the expressions in (4.7) to

\[
(e_1^T - V_1e_4^T)M \frac{\partial x(v; \alpha)}{\partial \alpha_j} = -(e_1^T - V_1e_4^T) \frac{\partial M}{\partial \alpha_j} x(v; \alpha) \quad \text{(A.1)}
\]

and

\[
(e_2^T - V_2e_4^T)M \frac{\partial x(v; \alpha)}{\partial \alpha_j} = -(e_2^T - V_2e_4^T) \frac{\partial M}{\partial \alpha_j} x(v; \alpha). \quad \text{(A.2)}
\]

(A.1) and (A.2) can be combined with (4.5) and (4.6) to a matrix expression

\[
\begin{pmatrix}
\frac{\partial f/\partial x}{(e_1^T - V_1e_4^T)M} \\
\frac{\partial x(v; \alpha)}{\partial \alpha_j} \\
\frac{\partial x(v; \alpha)}{\partial \alpha_j} \\
\end{pmatrix}
= \frac{\partial \alpha_j}{\partial \alpha_j}
\begin{pmatrix}
\frac{\partial f/\partial \alpha_j}{(e_1^T - V_1e_4^T)\partial M/\partial \alpha_j x(v; \alpha)} \\
\frac{\partial f/\partial \alpha_j}{(e_2^T - V_2e_4^T)\partial M/\partial \alpha_j x(v; \alpha)} \\
0
\end{pmatrix}
\quad \text{(A.3)}
\]
with
\[
\frac{\partial f}{\partial \alpha_j} = \left. \frac{\partial f(x; \alpha)}{\partial \alpha_j} \right|_{x=x(v; \alpha)}, \quad \frac{\partial f}{\partial x} = \left. \frac{\partial f(x; \alpha)}{\partial x} \right|_{x=x(v; \alpha)} \quad (A.4)
\]

and where \( M_0 \) has been defined as
\[
M_0 = \begin{pmatrix}
\frac{\partial f}{\partial x} M^{-1} \\
(e_1^T - V_1 e_1^T) \\
(e_2^T - V_2 e_2^T) \\
f \cdot e_3^T + e_4^T
\end{pmatrix}, \quad (A.5)
\]

where the lowest row of \( M_0 \) can be checked easily from (A.3) and (3.11). \( M_0 \) can be inverted with standard methods yielding
\[
M_0^{-1} = \frac{1}{k} \left( e_1^T - \frac{\partial f}{\partial x} M^{-1} \right) \begin{pmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & f \\
0 & 0 & 0 & 0
\end{pmatrix} + \begin{pmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & f \\
0 & 0 & 0 & 0
\end{pmatrix} \quad (A.6)
\]

with
\[
k = (\frac{\partial f}{\partial x}) M^{-1} v.
\]

Now (A.3) can be converted with (A.6) to
\[
\frac{\partial x(v; \alpha)}{\partial \alpha_j} = - \frac{1}{k} M^{-1} v \frac{\partial f}{\partial \alpha_j} - M^{-1} \left( I - \frac{1}{k} v \cdot \frac{\partial f}{\partial x} M^{-1} \right) \begin{pmatrix}
1 & 0 & 0 & -V_1 \\
0 & 1 & 0 & -V_2 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix} \frac{\partial M}{\partial \alpha_j} x(v; \alpha). \quad (A.7)
\]

(A.7) can still be simplified by the identity,
\[
\begin{pmatrix}
1 & 0 & 0 & -V_1 \\
0 & 1 & 0 & -V_2 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix} = I - \begin{pmatrix} 0 & 0 & v \end{pmatrix} - f e_3 \begin{pmatrix} 0 & 0 & f^{-1} \end{pmatrix}, \quad (A.8)
\]
with $\mathbf{0}$ the zero vector and by recognizing that the last two terms at the right side of the sign do not contribute to (A.7). Then we obtain from (A.7):

$$
\frac{\partial x(v; \alpha)}{\partial \alpha_j} = \frac{1}{k} M^{-1} v \frac{\partial f}{\partial \alpha_j} - M^{-1} \left( I - \frac{1}{k} v \frac{\partial f}{\partial x} M^{-1} \right) \frac{\partial M}{\partial \alpha_j} x(v; \alpha). \quad (A.9)
$$

(A.9) is the required expression yielding, together with (4.2) and (4.1), the result (4.8).

**APPENDIX B: DERIVATION OF THE UNIQUENESS RESULT (5.23)**

We write out the matrix $A$ from (5.16) in terms of the parameters. Consider first the matrix $MDM^{-1}$. We have

$$
M = P \begin{bmatrix}
\phi & l \phi(0) \\
0 & 1
\end{bmatrix}
$$

and

$$
D = \begin{bmatrix}
I + \Omega + S_e \\
e_4^T
\end{bmatrix} \Delta X.
$$

(B.1)

From the expression for $M$ we can calculate $M^{-1}$:

$$
M^{-1} = \begin{bmatrix}
\phi^{-1} & -l(0) \\
0 & 1
\end{bmatrix} P^{-1}.
$$

(B.2)

Multiplying out results in

$$
MDM^{-1} = P \begin{bmatrix}
\phi(I + \Omega + S) \phi^{-1} & \phi \left( \Delta X - l(\Omega + S) \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right) \\
e_4^T
\end{bmatrix} P^{-1}
$$

(B.3)

if we define the camera parameters $\Omega_e, S_e,$ and $\Delta_e$ according to (5.21) and (5.22). From (5.16) the matrix $A$ can be written as

$$
A = f^{-1} P \begin{bmatrix}
\Omega_e + S_e & \Delta_e \\
0^T
\end{bmatrix} e_4 e_2^T \begin{bmatrix}
\phi^{-1} & -l(0) \\
e_4^T
\end{bmatrix} \left[ I + \Omega_e + S_e & \Delta_e \right] P^{-1}
$$

(B.4)

where the fact was used that the element $(M^{-1})_{33}$ equals $f^{-1} I$ and where the
projection matrix $P$ was factored out. Now consider a second set of parameter values written with primed symbols. The resulting matrix $A'$ should equal the original one apart from the facts that it may be multiplied with a scalar constant $\lambda$ and that we can add to it the dyadic matrix,

$$a(0,0,1,f),$$  \hspace{1cm} (B.5)

where the vector $a$ can be chosen freely. This means that

$$PA'P^{-1} = \lambda P\left(\mathcal{A} + P^{-1}ae_4^T\right)P^{-1}$$ \hspace{1cm} (B.6)

$$\mathcal{A}' = \lambda\left(\mathcal{A} + P^{-1}ae_4^T\right).$$ \hspace{1cm} (B.7)

The last matrix in brackets has elements unequal zero only at the fourth column. Therefore, the last column of $\mathcal{A}'$ is unimportant. The first three columns of Eq. (B.7) give

$$\Delta_cN^T - I(I + \Omega_c + S_c) = \lambda\left[\Delta_c'N'^T - I'(I + \Omega'_c + S'_c)\right],$$  \hspace{1cm} (B.8)

where use was made of the fact that

$$\phi\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = N \quad \text{and} \quad \phi^T = \phi^{-1}$$ \hspace{1cm} (B.9)

with $N$ the normal of the plane according to (5.11).

**Rigid body motion.** In the case of rigid body motion, $S_c$ equals zero while $\Omega_c$ is skew symmetric and has diagonal elements zero. From (B.8) we have

$$\Delta_cN^T - \lambda\Delta_c'N'^T = (I - \Omega'_c)I - \lambda\Omega'_c + I\Omega_c.$$ \hspace{1cm} (B.10)

Now the left side of (B.8), being the difference of two dyadic products, must be singular, with the consequence that the right side is singular. Straightforward calculations show that this can happen only if

$$\lambda = l/l'.$$ \hspace{1cm} (B.11)

Thus,

$$\frac{1}{l}\Delta_cN^T - \Omega_c = \frac{1}{l'}\Delta_c'N'^T - \Omega'_c.$$ \hspace{1cm} (B.12)

Adding (B.12) to its transpose gives the result of Negahdaripour and Horn [17]:

$$\frac{1}{l}(\Delta_cN^T + N\Delta_c^T) = \frac{1}{l'}(\Delta_c'N'^T + N\Delta_c'^T).$$ \hspace{1cm} (B.13)

The only nontrivial solution is

$$N' = \Delta_c/\|\Delta_c\| \quad \text{and} \quad \Delta_c'/\|\Delta_c'\| = N$$ \hspace{1cm} (B.14)
and
\[
\frac{||\Delta_\nu||}{l'} - \frac{||\Delta_\omega||}{l}, \tag{B.15}
\]
where \(||a||\) is the length of the vector \(a\).

According to (B.14), the normal to the plane and the translation interchange directions and, according to (B.15), the depth \(l\) is a scaling factor for the translation.

**REFERENCES**