

Recursive Updating of Planar Motion

D W Murray and D M Pickup

Robotics Research Group

Department of Engineering Science

University of Oxford

Parks Road, Oxford, OX1 3PJ, UK.

Abstract

This paper presents a recursive algorithm to recover the 3D structure and motion of a planar facet moving with arbitrary but constant motion relative to a single camera. By integrating discrete measurements of visual motion over time, the algorithm imposes a coupling between the scene structure and rotational motion otherwise absent in instantaneous motion processing. The algorithm disambiguates between the two possible values of the rotational motion which arise from instantaneous processing, and shows considerable robustness to noise and small camera angles.

1 Introduction

The direct determination of the 3D disposition and motion of rigidly moving planar surfaces from a sequence of images was the subject of considerable research effort in the first half of the last decade. More recently, relatively more attention has been paid to the recovery of 3D features of lower dimensionality (points, lines and curves), which are more easily and accurately recovered, with surfaces being recovered by grouping at a higher level, if at all. This avoids the thorny issue of segmentation until 3D information is available. Lately, however, there has been a resurgence of interest in direct surface recovery to provide simple navigational aids for autonomous vehicles, where the issue of segmentation is simplified to be one of distinction between negotiable planar surface and obstacle (eg [3, 5]).

There have been several approaches to direct planar surface recovery, some using the discrete image motion found from identifying and tracking a small number of distinctive image features [20, 13]. Others have assumed that only edge-normal components of visual motion are available, as obtained from some edge or gradient based image processing [2]. Use has been made of the derivatives of visual motion [12, 14, 18] and various commitments to least squares analysis have been made [2, 15, 9]. A radically different approach was taken in [17] where changes in image brightness were used directly to recover the structure and motion of a planar surface. Other methods of recovery have involved direct analysis of the motion of contours [11, 6].

A common difficulty with the techniques is that of high sensitivity to noise. Those methods using a small number of discrete points tend to be particularly fragile (eg [20, 13]), but even those using a large number of points in a least squares approach are not particularly robust (eg [16]). A first and fundamental problem, and this applies generally to structure from motion recovery, is the difficulty of

distinguishing in an instantaneous motion field the image motion induced by rotation from that induced by translation. (By instantaneous processing we mean the exclusive use of visual motion data derived from images separated by some small time difference.) This problem is exacerbated as the angle of view of the camera is reduced [16]. A second problem is that there may be multiple solutions to the instantaneous recovery of structure and motion — these have been analysed for the case of a planar facet in [13, 15].

Both these problems can be mitigated by considering the motion of the surface over extended periods of time. Differences between translation and rotation then become apparent, because rotational motion is re-coupled to the scene structure, a coupling absent in the expressions for visual motion (as a glance at Equation (1) below will confirm). It is possible too to distinguish between the multiple solutions postulated by instantaneous processing. To achieve re-coupling it is necessary to incorporate all measurements over a sequence, and not merely use a previous solution as a good starting point for the next frame of instantaneous processing, as was the case in [9]. In this paper therefore we propose the use of an extended Kalman Filter (EKF) to update the structure and motion estimation. The Kalman filter is now well-used for structure-from-motion problems (eg [19, 10, 7, 1]). Of these, ref. [7] demonstrates recovery of a planar surface, but using a model-based description of an object to be found on the ground plane and ref. [19] examines the situation when the movement was constrained to lie in a plane. This paper looks at the problem of *arbitrary*, though constant, motion with respect to the plane.

The next section describes the geometry relating 2D visual motion and 3D scene motion. In section 2 we show how the scene evolves over time and in section 3 we incorporate our measurement and scene models into an EKF. The final section gives some results and early conclusions.

2 Scene and image geometry

The camera and scene geometries under consideration are shown in Figure 1. The camera's optic axis defines \hat{z} , and the image plane lies in the xy -plane at $z = -1$. The lens is assumed to be optically equivalent to a pinhole lying at the origin of the $Oxyz$ coordinate system. Under the perspective transformation, scene points $\mathbf{R} =$

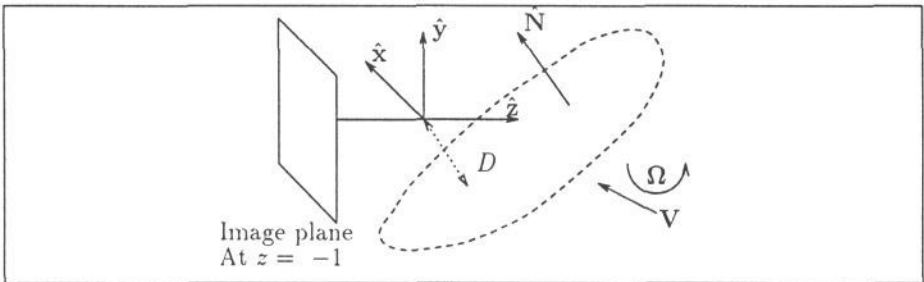


Figure 1: *The geometry describing the scene and the camera.*

$(X, Y, Z)^T$ are imaged at $\mathbf{r} = (x, y, -1)^T$ given by $\mathbf{r} = -\mathbf{R}/Z$. The projected motion in the image is the time differential of the above, that is $\dot{\mathbf{r}} = -\dot{\mathbf{R}}/Z + \mathbf{R}(\dot{\mathbf{R}} \cdot \hat{z})/Z^2$. Now the scene motion $\dot{\mathbf{R}}$ of a point with respect to the camera can be written in

terms of instantaneous rectilinear and angular velocities, \mathbf{V} and $\mathbf{\Omega}$, as $\dot{\mathbf{R}} = \mathbf{V} + \mathbf{\Omega} \wedge \mathbf{R}$, so that the projected motion is given in terms of the scene motion and structure as

$$\dot{\mathbf{r}} = -\frac{\mathbf{V}}{Z} - \frac{(\mathbf{V} \cdot \hat{\mathbf{z}})}{Z} \mathbf{r} + \mathbf{\Omega} \wedge \mathbf{r} + (\mathbf{\Omega} \wedge \mathbf{r} \cdot \hat{\mathbf{z}}) \mathbf{r}. \quad (1)$$

However, the recovery of scene structure and motion involves the determination of Z , \mathbf{V} and $\mathbf{\Omega}$ from *measurable* visual motion, which is unlikely to equate to the projected motion derived above [21]. The method we will use to recover motion from real imagery detects and tracks edge elements from frame to frame [4], combining some of the accuracy of a strong token tracking scheme with the dense coverage of a gradient-based scheme. Experience of recovering structure and motion using data computed by this method [16] suggests that it provides data which broadly agree with Equation (1), though because the visual motion recovery is based on local edge information, it recovers only edge-normal components of visual motion, the *vernier velocities*, \mathbf{v} .

Now any component \mathbf{v} of a vector $\dot{\mathbf{r}}$ obeys $v = (\dot{\mathbf{r}} \cdot \hat{\mathbf{v}})$, where \mathbf{v} is broken into its magnitude v and the unit direction vector $\hat{\mathbf{v}}$. Thus, taking the scalar product of $\hat{\mathbf{v}}$ with Equation (1) we have

$$v = -\frac{(\mathbf{V} \cdot \hat{\mathbf{v}})}{Z} - \frac{(\mathbf{V} \cdot \hat{\mathbf{z}})(\mathbf{r} \cdot \hat{\mathbf{v}})}{Z} + (\mathbf{\Omega} \wedge \mathbf{r} \cdot \hat{\mathbf{v}}) + (\mathbf{\Omega} \wedge \mathbf{r} \cdot \hat{\mathbf{z}})(\mathbf{r} \cdot \hat{\mathbf{v}}), \quad (2)$$

and the recovery of scene structure and motion now involves the recovery of Z , \mathbf{V} and $\mathbf{\Omega}$ from the image observables (\mathbf{v}, \mathbf{r}) . Now, whereas recovery *is* possible using observables $(\dot{\mathbf{r}}, \mathbf{r})$ and Equation (1), it proves *impossible* using (\mathbf{v}, \mathbf{r}) and Equation 2, as may be shown using an information counting argument [2]. To solve for structure and motion from vernier velocities, some extra constraint is required, and here it is that all scene points under consideration lie on a single planar facet in the scene.

The planar constraint

The scene planar facet is defined by $\mathbf{R} \cdot \hat{\mathbf{N}} = -D$, where the unit normal $\hat{\mathbf{N}}$ points out of the solid surface, and D is the perpendicular distance from the camera origin to the planar surface. Using the equation for perspective projection the reciprocal depth of a scene point is found as $Z^{-1} = (\mathbf{r} \cdot \hat{\mathbf{N}})/D$. When this is substituted into Equation (2), the magnitude of the vernier velocity becomes

$$v = -\left(\frac{V}{D}\right) (\mathbf{r} \cdot \hat{\mathbf{N}}) [(\hat{\mathbf{V}} \cdot \hat{\mathbf{v}}) + (\hat{\mathbf{V}} \cdot \hat{\mathbf{z}})(\mathbf{r} \cdot \hat{\mathbf{v}})] + (\mathbf{\Omega} \wedge \mathbf{r} \cdot \hat{\mathbf{v}}) + (\mathbf{\Omega} \wedge \mathbf{r} \cdot \hat{\mathbf{z}})(\mathbf{r} \cdot \hat{\mathbf{v}}), \quad (3)$$

where the rectilinear velocity \mathbf{V} has been separated into its magnitude V and direction $\hat{\mathbf{V}}$.

From eight or more measurements of (\mathbf{v}, \mathbf{r}) , it is possible at an instant to recover the eight independent scene parameters $\hat{\mathbf{V}}$ (two parameters), $\mathbf{\Omega}$ (three), $\hat{\mathbf{N}}$ (two), and the inverse “time-to-contact” $\tau = V/D$ (one) [2]. However, as we noted in the introductory section, instantaneous algorithms prove rather sensitive to noise in the input visual motion, partly because rotation is decoupled from the scene structure.

3 Evolution of scene parameters

Although rotation and structure are decoupled at any instant, a direct coupling can be established by considering the evolution of the scene over time.

First consider the unit surface normal. It varies as

$$\frac{d}{dt}\hat{\mathbf{N}} = \boldsymbol{\Omega} \wedge \hat{\mathbf{N}} \quad .$$

For simplicity, this work considers the evolution of the scene under the assumption that the scene velocities \mathbf{V} and $\boldsymbol{\Omega}$ remain constant over time. Hence we can write the variation of $\hat{\mathbf{N}}$ in integral form as

$$\hat{\mathbf{N}}(t) = \mathbf{Rot}(t)\hat{\mathbf{N}}(0)$$

where $\mathbf{Rot}(t)$ is the 3×3 orthonormal rotation matrix representing a rotation of angle Ωt about the axis $\boldsymbol{\Omega}$. Writing $C = \cos \Omega t$ and $S = \sin \Omega t$ it is straightforward to show that

$$\hat{\mathbf{N}}(t) = (1 - C) \frac{\boldsymbol{\Omega} \cdot \hat{\mathbf{N}}(0)}{\Omega^2} \boldsymbol{\Omega} + C \hat{\mathbf{N}}(0) + S \frac{\boldsymbol{\Omega} \wedge \hat{\mathbf{N}}(0)}{\Omega} \quad . \quad (4)$$

Secondly, we consider the evolution of the inverse time to contact, $\tau(t) = V/D$. As \mathbf{V} is assumed constant we need only consider the variation of $D(t)$. Recalling that $D(t) = -\mathbf{R}(t) \cdot \hat{\mathbf{N}}(t)$ and differentiating, we find

$$\begin{aligned} \dot{D}(t) &= -\dot{\mathbf{R}} \cdot \hat{\mathbf{N}}(t) - \mathbf{R} \cdot \frac{d}{dt} \hat{\mathbf{N}} &= -(\mathbf{V} + \boldsymbol{\Omega} \wedge \mathbf{R}) \cdot \hat{\mathbf{N}}(t) - \mathbf{R} \cdot (\boldsymbol{\Omega} \wedge \hat{\mathbf{N}}(t)) \\ & &= -V \hat{\mathbf{V}} \cdot \hat{\mathbf{N}}(t) \quad . \end{aligned}$$

Substituting the expression for $\hat{\mathbf{N}}(t)$ and integrating we obtain

$$\int_0^t \dot{D}(t) dt = -V \int_0^t \left[(1 - C) \frac{(\hat{\mathbf{V}} \cdot \boldsymbol{\Omega})(\boldsymbol{\Omega} \cdot \hat{\mathbf{N}}(0))}{\Omega^2} + C \hat{\mathbf{V}} \cdot \hat{\mathbf{N}}(0) + S \frac{\hat{\mathbf{V}} \cdot \boldsymbol{\Omega} \wedge \hat{\mathbf{N}}(0)}{\Omega} \right] dt.$$

Thus $D(t) = D(0) - V\lambda(t)$ where

$$\lambda(t) = \left[\left(t - \frac{S}{\Omega} \right) \frac{(\hat{\mathbf{V}} \cdot \boldsymbol{\Omega})(\boldsymbol{\Omega} \cdot \hat{\mathbf{N}}(0))}{\Omega^2} + \frac{S}{\Omega} \hat{\mathbf{V}} \cdot \hat{\mathbf{N}}(0) + (1 - C) \frac{\hat{\mathbf{V}} \cdot \boldsymbol{\Omega} \wedge \hat{\mathbf{N}}(0)}{\Omega^2} \right] ,$$

and so the update equation for the inverse time to contact is

$$\tau(t) = \frac{\tau(0)}{1 - \tau(0)\lambda(t)} \quad . \quad (5)$$

4 The Kalman Filter

Equations (4) and (5) describing the evolution of the scene structure, $\hat{\mathbf{N}}$, and inverse time to contact, τ , are non-linear. We therefore use the formalism of the extended Kalman filter to update the state over time.

The state vector describing the scene at discrete times t_k ($k = 0, 1, \dots$) is $\mathbf{X}_{(k)}$. The image measurements are non-linear functions of the state and are available at the discrete times t_k . In our case the measurements will be the *magnitudes* v_i of the vernier velocities \mathbf{v}_i , $i = 1, \dots, m$, and we assume that the positions \mathbf{r}_i and directions $\hat{\mathbf{v}}_i$ of the vernier velocities are exact. We will write the $m \times 1$ vector of magnitudes as \mathbf{z} . They are related to the state by $\mathbf{z} = \mathbf{h}(\mathbf{X}_{(k)}) + \mathbf{d}_k$, where \mathbf{h} is a non-linear vector-valued observation model and \mathbf{d}_k is an uncorrelated, zero mean, Gaussian noise sequence defined by $E[\mathbf{d}_k] = \mathbf{0}$ and $E[\mathbf{d}_i \mathbf{d}_j^T] = \delta_{ij} \mathbf{R}$. The state

vector at time t_{k+1} is related to that at t_k by $\mathbf{X}_{(k+1)} = \mathbf{f}(\mathbf{X}_{(k)}, \mathbf{u}_k) + \mathbf{e}_k$ where \mathbf{f} is a non-linear, vector valued, state transition function; \mathbf{u}_k is a control input (here zero); and \mathbf{e}_k is an uncorrelated, zero mean Gaussian noise sequence define by $E[\mathbf{e}_k] = \mathbf{0}$ with $E[\mathbf{e}_i \mathbf{e}_j^T] = \delta_{ij} \mathbf{Q}$. The *estimate* of the state vector at time t_k using all the data available up to and including that obtained at t_k is denoted by $\tilde{\mathbf{X}}_{(k|k)}$. If this estimate is used to predict the state at some later time t_{k+1} without using the data collected at t_{k+1} , the prediction is denoted $\tilde{\mathbf{X}}_{(k+1|k)}$.

Ignoring \mathbf{u} , the state prediction is $\tilde{\mathbf{X}}_{(k+1|k)} = \mathbf{f}(\tilde{\mathbf{X}}_{(k|k)})$, and the prediction of covariance is $\mathbf{P}_{(k+1|k)} = \nabla \mathbf{f} \mathbf{P}_{(k|k)} \nabla \mathbf{f}^T + \mathbf{Q}_{(k)}$. Using the data collected at t_{k+1} , the inverse covariance matrix (or information matrix) and state are updated as [8]

$$\begin{aligned} \mathbf{P}_{(k+1|k+1)}^{-1} &= \mathbf{P}_{(k+1|k)}^{-1} + \nabla \mathbf{h}^T \mathbf{R}_{(k+1)}^{-1} \nabla \mathbf{h}_{(k+1)} \\ \tilde{\mathbf{X}}_{(k+1|k+1)} &= \tilde{\mathbf{X}}_{(k+1|k)} + \mathbf{K}_{(k+1)} (\mathbf{z} - \mathbf{h}(\tilde{\mathbf{X}}_{(k+1|k)})) \end{aligned}$$

where the Kalman gain matrix is found as

$$\mathbf{K}_{(k+1)} = \mathbf{P}_{(k+1|k+1)} \nabla \mathbf{h}_{(k+1)}^T \mathbf{R}_{(k+1)}^{-1} .$$

In these expressions $\nabla \mathbf{f}$ is the Jacobian of \mathbf{f} and $\nabla \mathbf{h}$ is the Jacobian of \mathbf{h} , both evaluated at $\mathbf{X}_{(k)} = \tilde{\mathbf{X}}_{(k+1|k)}$.

The form of the state vector

Although only eight independent quantities are recoverable, the state vector \mathbf{X} adopted here has ten members:

$$\mathbf{X} = (\hat{V}_x, \hat{V}_y, \hat{V}_z, \Omega_x, \Omega_y, \Omega_z, \hat{N}_x, \hat{N}_y, \hat{N}_z, \tau)^T .$$

The two superfluous parameters are constrained by demanding that $\hat{\mathbf{V}}$ and $\hat{\mathbf{N}}$ are unit vectors. To achieve this two further *measurements* are appended. If there are m visual motion observations $\mathbf{z} = (v_1 \dots v_m)^T$ we add the two measurements

$$z_{m+1} = \hat{V}_x^2 + \hat{V}_y^2 + \hat{V}_z^2 = 1 , \quad \text{and} \quad z_{m+2} = \hat{N}_x^2 + \hat{N}_y^2 + \hat{N}_z^2 = 1 .$$

Although strictly both measurements are noiseless ($d_{m+1} = d_{m+2} = 0$) because the inverse of \mathbf{R} is required their variances are set very small, but finite.

Deriving the Jacobians

Under the assumption of constant motion, the 10×10 Jacobian matrix $\nabla \mathbf{f}$ can be broken down into components as:

$$\nabla \mathbf{f} = \begin{pmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & 0 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & 0 \\ \mathbf{0}_3 & \mathbf{A}_3 & \mathbf{B}_3 & 0 \\ \mathbf{a}^T & \mathbf{b}^T & \mathbf{c}^T & g \end{pmatrix}$$

Here \mathbf{I}_3 and $\mathbf{0}_3$ are 3×3 identity and zero matrices, respectively. The 3×3 matrices \mathbf{A} , \mathbf{B} , the 3×1 vectors \mathbf{a} , \mathbf{b} , \mathbf{c} , and the scalar g are found by differentiating Equations (4) and (5) with respect to the state variables. In detail these quantities are found as follows.

From Equation (4):

$$A_{il} = \partial \hat{N}_i(t) / \partial \Omega_l(0) = \frac{\Omega \cdot \hat{N}}{\Omega^2} \left[\frac{\Omega_i \Omega_l}{\Omega} \left(St - 2 \frac{(1-C)}{\Omega} \right) + (1-C) \delta_{il} \right] \\ + (1-C) \frac{\Omega_i \hat{N}_l}{\Omega^2} - St \frac{\hat{N}_i \Omega_l}{\Omega} + \frac{\Omega_l (\Omega \wedge \hat{N})_i}{\Omega^2} \left(Ct - \frac{S}{\Omega} \right) + S \epsilon_{ilj} \frac{\hat{N}_j}{\Omega} .$$

Also from Equation (4),

$$B_{il} = \partial \hat{N}_i(t) / \partial \hat{N}_l(0) = (1-C) \frac{\Omega_i \Omega_l}{\Omega^2} + C \delta_{il} - S \epsilon_{ilk} \frac{\Omega_k}{\Omega} .$$

From Equation (5), the components of **a** are:

$$a_i = \frac{\partial \tau(t)}{\partial \hat{V}_i} = \left[\frac{\tau(0)}{1 - \tau(0)\lambda} \right]^2 \left[\left(t - \frac{S}{\Omega} \right) \frac{\Omega \cdot \hat{N} \Omega_i}{\Omega^2} + \frac{S}{\Omega} \hat{N}_i + (1-C) \frac{(\Omega \wedge \hat{N})_i}{\Omega^2} \right] .$$

The components of **b** are:

$$b_i = \frac{\partial \tau(t)}{\partial \Omega_i} = \left[\frac{\tau(0)}{1 - \tau(0)\lambda} \right]^2 \frac{\partial \lambda}{\partial \Omega_i}$$

where

$$\frac{\partial \lambda}{\partial \Omega_i} = \left(\frac{S}{\Omega} - Ct \right) \left[\frac{(\hat{V} \cdot \Omega)(\Omega \cdot \hat{N})}{\Omega^2} - \hat{V} \cdot \hat{N} \right] \frac{\Omega_i}{\Omega^2} \\ + \left(t - \frac{S}{\Omega} \right) \left[\frac{\Omega \cdot \hat{N} \hat{V}_i}{\Omega^2} + \frac{\hat{V} \cdot \Omega \hat{N}_i}{\Omega^2} - 2 \frac{(\hat{V} \cdot \Omega)(\Omega \cdot \hat{N}) \Omega_i}{\Omega^4} \right] \\ + \hat{V} \cdot (\Omega \wedge \hat{N}) \left[\frac{St \Omega_i}{\Omega^3} - 2(1-C) \right] - (1-C) \frac{(\hat{V} \wedge \hat{N})_i}{\Omega^2} .$$

Similarly,

$$c_i = \frac{\partial \tau(t)}{\partial \hat{N}_i} = \left[\frac{\tau(0)}{1 - \tau(0)\lambda} \right]^2 \frac{\partial \lambda}{\partial \hat{N}_i}$$

where

$$\frac{\partial \lambda}{\partial \hat{N}_i} = \left[\left(t - \frac{S}{\Omega} \right) \frac{\hat{V} \cdot \Omega \Omega_i}{\Omega^2} + \frac{S}{\Omega} \hat{V}_i + (1-C) \frac{(\hat{V} \wedge \Omega)_i}{\Omega^2} \right] .$$

The final component, the scalar g , is given by

$$g = \frac{\partial \tau(t)}{\partial \tau(0)} = (1 - \tau(0)\lambda)^{-2} .$$

The second Jacobian, $\nabla \mathbf{h}$, has elements $\nabla h_{ij} = \partial v_i / \partial X_j$. For the measurements of visual motion ($i = 1, \dots, m$) the rows have identical form:

$$\begin{aligned} \nabla h_{i1} &= -\tau(\mathbf{r}_i \cdot \hat{N}) \cos \rho_i & \nabla h_{i2} &= -\tau(\mathbf{r}_i \cdot \hat{N}) \sin \rho_i \\ \nabla h_{i3} &= -\tau(\mathbf{r}_i \cdot \hat{N})(\mathbf{r}_i \cdot \hat{v}_i) & \nabla h_{i4} &= \sin \rho_i + y_i(\mathbf{r}_i \cdot \hat{v}_i) \\ \nabla h_{i5} &= -\cos \rho_i - x_i(\mathbf{r}_i \cdot \hat{v}_i) & \nabla h_{i6} &= -y_i \cos \rho_i + x_i \sin \rho_i \\ \nabla h_{i7} &= -\tau[(\hat{V} \cdot \hat{v}_i) + (\hat{V} \cdot \hat{z})(\mathbf{r}_i \cdot \hat{v}_i)] x_i & \nabla h_{i8} &= -\tau[(\hat{V} \cdot \hat{v}_i) + (\hat{V} \cdot \hat{z})(\mathbf{r}_i \cdot \hat{v}_i)] y_i \\ \nabla h_{i9} &= \tau[(\hat{V} \cdot \hat{v}_i) + (\hat{V} \cdot \hat{z})(\mathbf{r}_i \cdot \hat{v}_i)] & \nabla h_{i10} &= -[(\mathbf{r}_i \cdot \hat{N})(\hat{V} \cdot \hat{v}_i) \\ & & & + (\hat{V} \cdot \hat{z})(\mathbf{r}_i \cdot \hat{N})(\mathbf{r}_i \cdot \hat{v}_i)] . \end{aligned}$$

The remaining two rows $m+1$ and $m+2$ corresponding to the two constraint measurements have zero elements save for:

$$\begin{aligned} \nabla h_{m+1,1} &= 2\hat{V}_x; \quad \nabla h_{m+1,2} = 2\hat{V}_y; \quad \nabla h_{m+1,3} = 2\hat{V}_z \\ \nabla h_{m+2,7} &= 2\hat{N}_x; \quad \nabla h_{m+2,8} = 2\hat{N}_y; \quad \nabla h_{m+2,9} = 2\hat{N}_z . \end{aligned}$$

5 Experiments

The performance of the filter has been explored on simulated motion data. As an example Figure 2 first shows successive frames of motion derived for the following scene

$$\hat{V} = (0.0 \ 0.0 \ -1.0)^T \quad \Omega = (0.0 \ 0.0 \ 0.1)^T \quad \hat{N} = (0.0 \ 0.995 \ -0.0995)^T \quad \tau = 0.1$$

(one can imagine the camera to be diving gently towards the ground, rolling to port). The field of view was 40° . After finding the full motion at points, vernier velocities

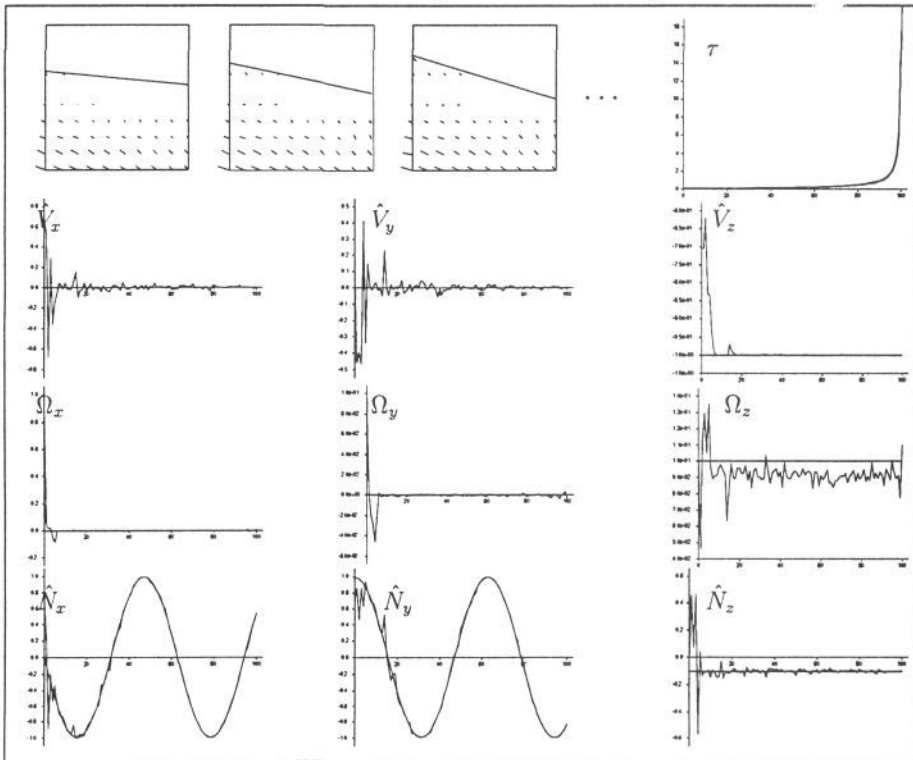


Figure 2: Three successive motion frames and the evolution over time for the ten members of the filter and plant state vectors.

were obtained by projection onto randomly chosen directions. In this experiment, the magnitudes were corrupted with $\sim 20\%$ noise by multiplying by $1 + \epsilon(0, 0.2)$ where $\epsilon(a, \sigma)$ deliver a random number from a Gaussian distribution with mean a and standard deviation σ . The rest of Figure 2 shows in detail the convergence of the ten elements of filter state towards that of the plant. (Note that the rotation about the \hat{z} -axis causes both \hat{N}_x and \hat{N}_y to vary sinusoidally.) After 100 frames, the simulation ends with the steep rise of $\tau = V/D$ towards infinity as the camera impacts the ground.

6 Conclusions

We have described an Extended Kalman Filter algorithm which utilizes vernier image velocities to integrate the ten states describing a moving planar surface over time, under the assumption of constant scene motion relative to the camera.

The experiments to date have shown the filter algorithm to function satisfactorily with $\sim 10 - -20\%$ noise in the input and fields of view of $\sim 20^\circ$, a considerable improvement in robustness over instantaneous measurements. Experiments are underway to use the filter with motion data computed from real imagery captured from a camera moved by a robot arm. The motion of the camera may well not agree with the model and the visual motion data will almost certainly contain errors which lie outside the model of white noise assumed here. Only when these results are available can a proper assessment of the filter's performance be made.

Initializing EKFs with a large number of parameters appears to be a black art, and a considerable amount of testing will be required to gain an empirical feel for the behaviour of the present filter, especially when initialized near an ambiguous solution, or when the plant model is incorrect, for example, if the motion were not quite constant.

Finally we note that the equations used in the updating are surprisingly involved for so simple a scene model, and require a good deal of computation. To what extent this can be reduced, for example, by using look-up tables for trigonometry, remains to be investigated. In broader terms, the high cost raises the question of whether such a filter captures the essence of the problem.

References

- [1] T J Broida, S Chandrashekar, and R Chellappa. Recursive estimation of 3D kinematics and structure from a long image sequence. *Preprint*, 1989.
- [2] B F Buxton, H Buxton, D W Murray, and N S Williams. 3D solutions to the aperture problem. In T O'Shea, editor, *Advances in Artificial Intelligence*, pages 105-114. Elsevier, Amsterdam, 1984.
- [3] S Carlsson. Object detection using model based prediction and motion parallax. In O D Faugeras, editor, *Proceedings of the 1st European Conference on Computer Vision, Antibes, France*, pages 297-306, Springer Verlag, Berlin, 1990.
- [4] D A Castelov, D W Murray, G L Scott, and B F Buxton. Matching Canny edgels to compute the principal axes of optic flow. *Image and Vision Computing*, 6(2):129-136, 1988.
- [5] D A Castelov and A J Rerolle. A monocular ground plane estimation system. In *Proceedings of the 2nd British Machine Vision Conference, Glasgow* (These proceedings) 1991.
- [6] R Cipolla. *Active visual inference of surface shape*. DPhil thesis (in preparation), University of Oxford, 1991.
- [7] R Evans. Kalman filtering of pose estimates in applications of the RAPID video rate tracker. In *Proceedings of the 1st British Machine Vision Conference, Oxford*, pages 79-84, 1990.

- [8] Y Bar-Shalom and T E Fortmann. *Tracking and Data Association*. Academic Press, New York, 1988.
- [9] P Greenaway. Interpreting edge-based optical flow. In A Barrett, editor, *Computer Vision and Image Processing*, pages 158–181. Chapman and Hall, London, 1991.
- [10] C G Harris and J M Pike. 3D positional integration from 3D sequences. *Image and Vision Computing*, 6(2):87–90, 1988.
- [11] K Kanatani. Detecting the motion of a planar surface by line and surface integrals. *Computer Vision, Graphics and Image Processing*, 29:13–22, 1985.
- [12] J J Koenderink and A J van Doorn. Local structure of movement parallax of the plane. *Journal of the Optical Society of America*, 66(7):717–723, 1976.
- [13] H C Longuet-Higgins. The visual ambiguity of a moving plane. *Proc Roy Soc Lond B*, 223:165–175, 1984.
- [14] H C Longuet-Higgins and K Prazdny. The interpretation of a moving retinal image. *Proc Roy Soc Lond B*, 208:385–397, 1980.
- [15] S J Maybank. *A theoretical study of optical flow*. PhD thesis, Birkbeck College, University of London, 1987.
- [16] D W Murray and B F Buxton. *Experiments in the Machine Interpretation of Visual Motion*. MIT Press, Cambridge, MA, 1990.
- [17] S Negadaripour and B K P Horn. Determining 3D motion of planar objects from image brightness patterns. In *Proceedings of the 9th International Joint Conference on Artificial Intelligence, IJCAI-85, Los Angeles CA*, pages 898–901, Los Altos CA, 1985. Morgan Kaufmann.
- [18] M Subbarao and A M Waxman. On the uniqueness of image flow solutions for planar surfaces in motion. Technical Report CAR-TR-114, Center for Automation Research, University of Maryland, 1985.
- [19] G Toscani and O D Faugeras. Structure and motion from two noisy perspective views. 1987
- [20] R Y Tsai and T S Huang. Estimating three dimensional motion parameters of a rigid planar patch. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-29(6):1147–1152, 1981.
- [21] A Verri and T Poggio. Motion fields and optical flow: qualitative properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5:490–498, 1989.