A Step Towards Self-calibration in SLAM: Weakly Calibrated On-line Structure and Motion Estimation

Sebastian Haner and Anders Heyden
Centre for Mathematical Sciences
Lund University, Sweden
haner@maths.lth.se, heyden@maths.lth.se

Abstract

We propose a structure and motion estimation scheme based on a dynamic systems approach, where states and parameters in a perspective system are estimated. An on-line method for structure and motion estimation in densely sampled image sequences is presented. The proposed method is based on an extended Kalman filter and a novel parametrization. We derive a dynamic system describing the motion of the camera and the image formation. By a change of coordinates, we represent this system by normalized image coordinates and the inverse depths. Then we apply an extended Kalman filter for estimation of both structure and motion. Furthermore, we assume only weakly calibrated cameras, i.e. cameras with unknown and possibly varying focal length, unknown and constant principal point and known aspect ratio and skew. The performance of the proposed method is demonstrated in both simulated and real experiments. We also compare our method to the one proposed by Civera et al. and show that we get superior results.

1. Introduction

Estimation of 3D structure and motion from 2D images is a central problem in computer vision. There exist essentially two different approaches to solve this problem; (i) batch approaches and (ii) iterative (recursive) approaches. Batch approaches aim at providing an accurate result by using all the images at the same time. These approaches are typically based on multi-view tensors, bundle adjustment or convex optimization, see [9] for the former and [13] for the latter. These methods are not suitable for mobile applications, both due to their complexity and due to the off-line nature, requiring all images to be gathered before any computations can be made. Iterative (or recursive) approaches aim at real-time performance, by updating a current estimate as soon as a new image becomes available. These approaches are either based on variations of methods used for batch approaches, e.g. iteratively estimating the camera pose and the structure, [3], or by fast estimation of relative motion [18]. The first ones are not suitable for mobile applications either, due to their high computational complexity, while the second ones have a higher potential.

Yet another approach is to formulate the camera motion and the imaging process as a dynamic system and apply non-linear observers to estimate the structure and the translational and rotational velocities of the motion. The standard approach is to apply an extended Kalman filter to a dynamic system, with a perspective transformation in the output equations. One of the pioneering approaches is [2], where an extended Kalman filter is applied directly to the dynamic system, without any re-parametrization. Another approach, based on tracking the essential matrix can be found in [19].

For structure estimation only, i.e. known motion, a number of non-linear observers based on methods for automatic control theory have been developed, e.g. [17, 12, 4, 8, 1, 15, 14, 6]. Similar approaches, based on adaptive non-linear observers, for full structure and motion estimation can be found in [20, 21, 10].

Lately, [7, 5] developed a variation of the extended Kalman filter, by using the inverse depth as one of the parameters, adjusting the uncertainties to the imaging situation and fixing the imaging rays from the first camera in order to gain stability. The method is highly over-parameterized but performs well in most situations, both in terms of accuracy and robustness. Another approach based on inverse scaling can be found in [16]. These methods are suitable for mobile applications, due to their recursive nature and relatively low computational complexity.

This paper describes how a re-parametrization of the underlying perspective dynamic system can be used to formulate the structure and motion estimation problem as an observer problem of a non-linear dynamic system, with a linear output function. We will show that this novel parametrization will result in a more accurate extended

Kalman filter. Moreover, we will allow a weakly calibrated camera, i.e. a camera with unknown and possibly varying focal length, unknown and constant principal point and known aspect ratio and skew.

2. Problem formulation

Consider a calibrated perspective camera that is observing a moving rigid object. Observe that it is just a philosophical difference between assuming a fixed camera and a moving object or a moving camera and a fixed object, since it is only the relative motion that can be estimated, but for modelling purposes one or the other might be preferable. Assume a camera system where the camera is situated at the origin and the optical axis is aligned with the z-axis. Let y_i denote the image coordinates and x_i denote the (time-varying) object coordinates. Introducing

$$\xi = \begin{pmatrix} \frac{x_1}{x_3} & \frac{x_2}{x_3} \end{pmatrix}^T \ , \tag{1}$$

we can write down a dynamic system

$$\begin{aligned} \dot{x} &= Ax + b \, , \\ y &= C_f \xi + \delta \ , \end{aligned} \tag{2}$$

where

$$A = S(\omega) = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix} \tag{3}$$

is the skew symmetric matrix obtained from the (possibly time varying) angular velocity vector

$$\omega = \begin{pmatrix} \omega_1 & \omega_2 & \omega_3 \end{pmatrix}^T , \tag{4}$$

$$b = \begin{pmatrix} b_1 & b_2 & b_3 \end{pmatrix}^T \tag{5}$$

denotes the (possibly time varying) translational velocity, and C_f and δ are intrinsic camera parameters. In our case we have

$$C_f = \begin{pmatrix} af_c & sf_c \\ 0 & f_c \end{pmatrix} \ , \tag{6}$$

where s denotes the (known) skew, a the (known) aspect ratio and f_c the (unknown and possibly varying) focal length and

$$\delta = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} \ , \tag{7}$$

denotes the (unknown) principal point. After a suitable change of coordinates, we may assume that

$$C_f = \begin{pmatrix} f_c & 0 \\ 0 & f_c \end{pmatrix} \ , \tag{8}$$

since s and γ are assumed to be known.

We can now state the problem as follows:

Problem 1 (On-line structure and motion estimation) *Given the image coordinates y from (2), estimate recursively the object coordinates x, the (time varying) motion parameters ω and b and the (time varying) focal length f.*

3. The parametrization

Considering (2) and introduce the scalar parameter γ and the vector z by

$$\gamma = \frac{1}{\sqrt{x^T x}}, \quad z = \gamma x \ , \tag{9}$$

which can be interpreted as the inverse distance to the object. Observe that ξ, according to (1) and by the definition of z in (9), also can be expressed as

$$\xi = \begin{pmatrix} \frac{z_1}{z_3} & \frac{z_2}{z_3} \end{pmatrix}^T \ . \tag{10}$$

This means, using (9) and the definition of ξ in (1), that the vector z which then can be expressed as

$$z = \frac{1}{\sqrt{\xi_1^2 + \xi_2^2 + 1}} \begin{pmatrix} \xi_1 & \xi_2 & 1 \end{pmatrix}^T \tag{11}$$

can also be assumed known. This vector can be interpreted as the image coordinates on a spherical image plane.

In the case of calibrated cameras, z is a measurable signal, and can therefore be considered an output of the system (2). The parametrization exploits this fact, and aims at rewriting the system (2) so that z appears explicitly in the equations. In the self-calibration case, i.e. where the focal length f_c and the principal point (x_0, y_0) are unknown, z is measurable up to a transformation involving the intrinsic parameters of the camera.

Using (2) and the fact that $x^T A x = 0$ since A is skew-symmetric, gives, introducing

$$g_0(z) = I - zz^T \tag{12}$$

a rewritten dynamic system, corresponding to (2), on the form

$$\begin{aligned} \dot{z} &= Az + g_0(z) b \gamma \\ \dot{\gamma} &= -\gamma^2 z^T b \\ y &= C_f \xi + \delta \, . \end{aligned} \tag{13}$$

For the motion of more than one point a dynamic system corresponding to (13) is obtained as

$$\begin{aligned} \dot{z}^i &= Az^i + g_0(z^i) b \gamma^i \\ \dot{\gamma}^i &= -(\gamma^i)^2 (z^i)^T b \quad , \quad i \in \{1, 2 \dots N\} \ , \\ y^i &= C_f \xi^i + \delta \end{aligned} \tag{14}$$

where N denotes the number of feature points. Equation (9) together with (13) and its multipoint version (14), constitute the desired dynamic vision parametrization, from which we shall proceed. Observe that the dynamic system contains 4 state variables; 3 for z and 1 for γ and that z has to fulfill the constraint $|z| = 1$.

4. The extended Kalman filter

The extended Kalman filter estimates the system state s_k given a previous estimate $\hat{s}_{k-1}$, a new measurement μ and state transition and observation models $s_k = f(s_{k-1})$ and $\mu_k = h(s_k)$. At every time-step the new state and the state covariance P are predicted,

$$\begin{aligned} \hat{s}_{k|k-1} &= f(\hat{s}_{k-1|k-1}) \\ P_{k|k-1} &= F_{k-1} P_{k-1|k-1} F_{k-1}^T + Q_{k-1} \end{aligned} \tag{15}$$

and, given a new measurement μ_k, corrected to

$$\begin{aligned} \hat{s}_{k|k} &= \hat{s}_{k|k-1} + K_k\big(\mu_k - h(\hat{s}_{k|k-1})\big) \\ P_{k|k} &= P_{k|k-1} - K_k H_k P_{k|k-1} \end{aligned} \tag{16}$$

where

$$\begin{aligned} F_{k-1} = \left.\frac{\partial f}{\partial s}\right|_{\hat{s}_{k-1|k-1}}, \quad H_k = \left.\frac{\partial h}{\partial s}\right|_{\hat{s}_{k|k-1}} \\ K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1} \end{aligned} \tag{17}$$

and Q and R the assumed process and measurement noise covariances, respectively.

Adapting the dynamic system (14) to the EKF setting, the state vector is taken to be

$$s = \left(b^T, \omega^T, f_c, x_0, y_0, (z^1)^T, \gamma^1, \ldots, (z^N)^T, \gamma^N\right)^T, \tag{18}$$

while the measurement vector is given by

$$\mu = h(s) = (y_1^1, y_2^1, \ldots, y_1^N, y_2^N)^T \tag{19}$$

with components

$$y^i = C_f \xi^i + \delta = \begin{pmatrix} f_c & 0 \\ 0 & f_c \end{pmatrix} \left(\frac{z_1^i}{z_3^i}, \frac{z_2^i}{z_3^i}\right)^T + \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}. \tag{20}$$

The update equation is a discretized version of (14):

$$\begin{aligned} \tilde{z}^i &= e^{S(\omega_k)} z_k^i + g_0(z_k^i) b_k \gamma_k^i \\ \tilde{\gamma}^i &= \gamma_k^i - (\gamma_k^i)^2 (z_k^i)^T b_k \\ z_{k+1}^i &= \tilde{z}^i |\tilde{z}^i|^{-1} \\ \gamma_{k+1}^i &= \tilde{\gamma}^i |\tilde{z}^i| \end{aligned} \quad , \quad i \in \{1, 2 \ldots N\} \, , \tag{21}$$

where $|z^i| = 1$ is enforced.

Note that we assume a camera-centric coordinate system and estimate only linear and angular velocities, which must be integrated over time to recover the absolute motion.

Adding features

A main advantage of the camera-centric coordinate system is the ease with which new features can be inserted into the filter; the uncertainty of new features is independent of any extrinsic camera parameters, unlike in the unified inverse depth method. Removing features simply means deleting the corresponding entries, rows and columns in the state vector and covariance matrices, but if a feature is only temporarily occluded or otherwise not detected, it can still be kept in the filter if it is assigned an infinite measurement uncertainty.

Complexity

The computational complexity of the filter can be made lower than that of e.g. the unified inverse depth parametrization. First note that since the output function h is linear if the measured image coordinates are first transformed using equation (11), it is fully represented by the Jacobian H, and very sparse. In fact, all non-zero elements equal one, and multiplying a matrix by H amounts to removing rows or columns of the matrix. Thus four matrix multiplications can be avoided in the filter update step. The update equation, however, is not linear, due to the camera-centric representation, and the Jacobian F will no longer be (nearly) diagonal, as in the world-centric case. But it will still be rather sparse, with the first 6 columns full and the rest block diagonal with 4-by-4 blocks. The a-priori covariance update can thus still be performed quite efficiently. The fact that only 3 or 4 parameters are required per feature is a considerable advantage compared to the unified inverse depth scheme, where features must eventually be converted to a Cartesian parametrization to maintain frame rate.

5. Experiments

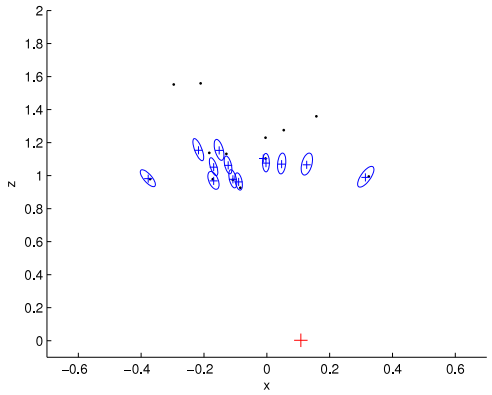
In the following experiments, no priors on the structure or motion are given. Features are initialized at an arbitrary depth and with large uncertainty in the γ coordinate. The linear and angular velocities are assumed constant, and acceleration is modelled as zero-mean Gaussian process noise. When assuming varying focal length, the variation is also modelled as process noise, while an unknown principal point is assumed fixed but initialized with some uncertainty at the center of the image.

As has been reported in [5], the EKF can converge under these circumstances; however, it is found that fixing the depth of one point, thus determining the overall scale, greatly aids convergence. Further, the normalization step of the update equation (21) has been found not to be strictly necessary (when using the full parametrization) and in fact does not significantly impact the results.

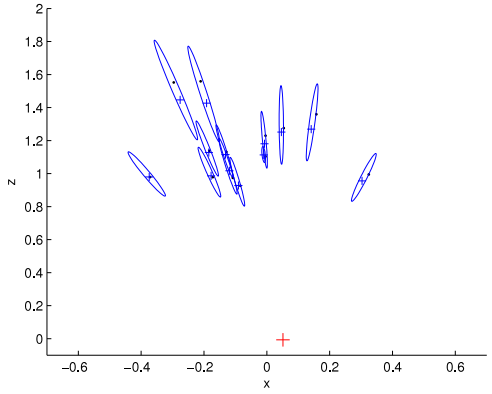
Constant and known intrinsic parameters

We repeat an experiment in [16] and show that the proposed parametrization does not suffer from the underestimation of uncertainty associated with the inverse depth parametrization of [5] and typically converges faster as a

result (Fig. 1 and Fig. 2). This issue of inconsistency is common to many SLAM algorithms and is analyzed in e.g. [11].



(a) Unified inverse depth



(b) Proposed

Figure 1. Position and covariance estimates after observing 30 frames of simulated data (black: ground truth, blue: estimate $\pm\sigma$). The inverse depth parametrization underestimates the errors, here leading to slower convergence, while the proposed parametrization more accurately captures the depth uncertainty.

Varying focal length and unknown principal point

In experiments on noisy simulated data the filter is able to track varying focal length and determine an offset in the principal point (Fig. 4 and Fig. 5). Observe that even if the convergence rate for the principal point is relatively low, the motion estimation and structure estimation is converging much faster.

Real data

We also apply the proposed and unified inverse depth methods to a real video sequence. The camera motion and 3D coordinates of 7 feature points tracked (using the KLT

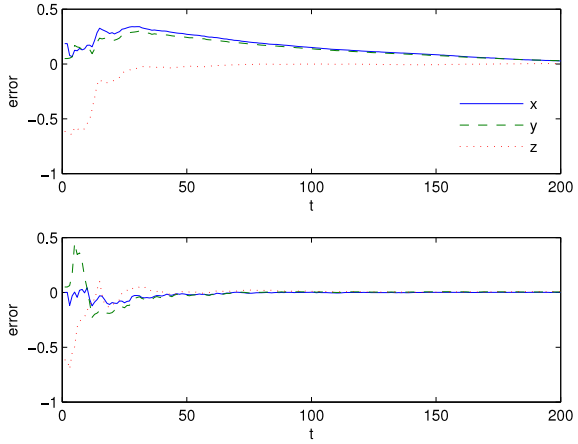


Figure 2. Convergence plot of the Cartesian coordinates of a point in a simulated reconstruction problem. Top: inverse depth, bottom: proposed parametrization.

algorithm) over 70 frames of a desktop sequence are reconstructed. Some geometry is overlaid to verify the results (Fig. 3). A (subjective) assessment indicates that the proposed method gives a more consistent reconstruction than unified inverse depth.

In a more general setting we use the following structure and motion framework:

- From the first frame, extract SURF features and add them to the state vector. The observed features are initialized to lie in a plane at unit depth. To set the overall scale, the depth of one feature is fixed by assigning zero uncertainty in the γ coordinate.
- For subsequent frames:
  1. Extract and match features to those active in the filter. Remove outliers by fitting an affine transformation between the observed feature locations and their predicted locations in a RANSAC scheme.
  2. Assign active features not detected in the current frame infinite measurement uncertainty. Features not detected for a set number of frames, e.g. 100, are removed from the filter by deleting the appropriate entries.
  3. Update the filter state.
  4. If the number of active features is too low, select new ones from the unmatched features in the current frame and initialize their depth as the mean depth of the currently visible active features.

While SURF features are computationally expensive (compared e.g. to the approach in [7]), they facilitate the rede-

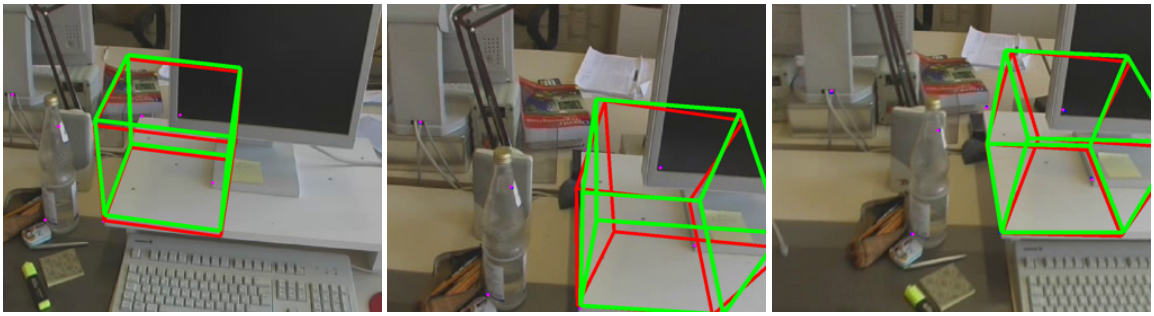


Figure 3. Visual result of integrating geometry into a tracked video sequence (from left to right, frames 1, 50 and 70 are shown). The green box shows the solution using the proposed method, while the red was computed using the inverse depth parametrization. Although the re-projection errors are similar, the proposed method produces a more accurate motion estimate.

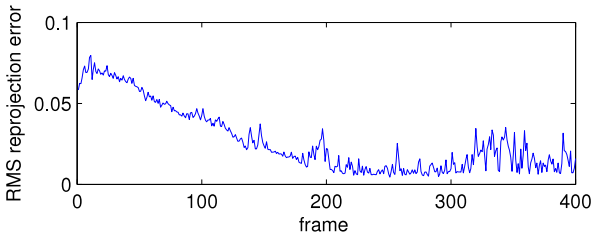


Figure 5. Re-projection error in the above experiment. The error increases towards the end of the sequence as the camera moves further from the point cloud.

tection of previously observed landmarks. In Fig. 6 the algorithm is applied to a typical augmented reality scenario.

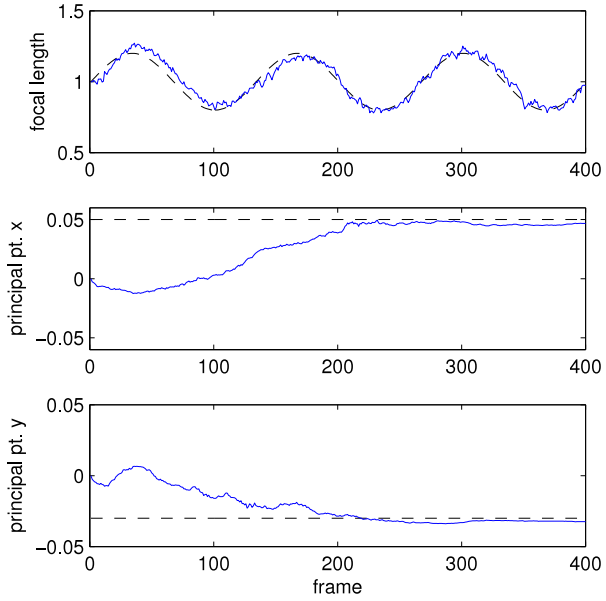


Figure 4. Result of experiment on simulated data. 15 intermittently visible features were observed by a circling camera at a noise level of about 1 pixel. The focal length varied while the principal point remained fixed. Dashed lines: ground truth.

6. Conclusions

We have used a novel parametrization in order to develop an extended Kalman filter for full structure and motion estimation. The filter is shown to perform well on both simulated and real data and has been compared to other state-of-the-art approaches. We have furthermore successfully dealt with unknown and varying focal length and unknown principal point. The relatively low computational complexity of the proposed filter should make it feasible for use on on mobile devices.

References

[1] R. Abdursul, H. Inaba, and B. K. Ghosh. Nonlinear observers for perspective time-invariant linear systems. *Automatica*, 40:481–490, 2004.

[2] A. Azarbayejani and A. P. Pentland. Recursive estimation of motion, structure, and focal length. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(6):562–575, 1995.

[3] P. Beardsley, A. Zisserman, and D. Murray. Sequential updating of projective and affine structure from motion. *International Journal of Computer Vision*, 23(3):235–259, 1997.

[4] X. Chen and H. Kano. A new state observer for perspective systems. *IEEE Transactions on Automatic Control*, 47(4):658–663, April 2002.

[5] J. Civera, A. Davison, and J. Montiel. Inverse depth parametrization for monocular slam. *IEEE Transactions on Robotics*, 24(5):932–945, 2008.

[6] O. Dahl, F. Nyberg, and A. Heyden. Nonlinear and adaptive observers for perspective dynamic systems. In *American Control Conference*, July 2007.

[7] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, 2007.

[8] W. E. Dixon, Y. Fang, D. M. Dawson, and T. J. Flynn. Range identification for perspective vision systems. *IEEE Transac-*

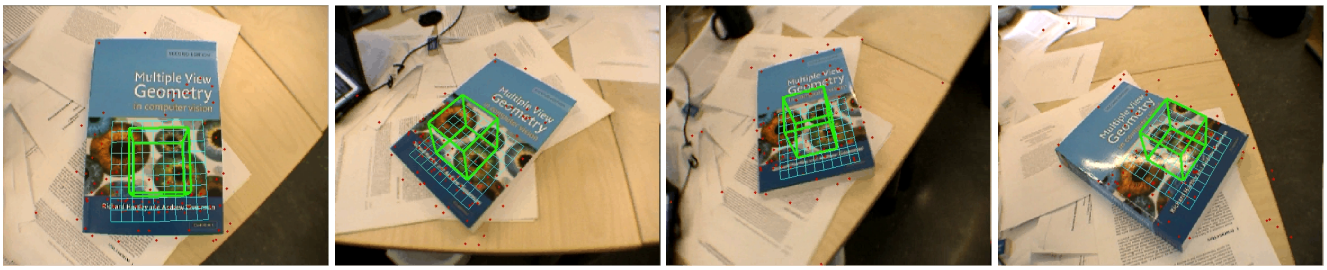


Figure 6. Visual result of integrating geometry into a video sequence using the proposed parametrization and a fully calibrated camera. SURF feature matching and RANSAC provide robust tracking. The cube is automatically aligned after detecting the dominant plane.

tions on Automatic Control, 48(12):2232–2238, December 2003.

[9] R. Hartley and A. Zisserman. *Multiple View Geometry*. Cambridge University Press, 2003.

[10] A. Heyden and O. Dahl. Provably convergent structure and motion estimation for perspective systems. In *Control Decision Conference*, 2009.

[11] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis. Analysis and improvement of the consistency of extended Kalman filter-based SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 473–479, Pasadena, CA, 2008.

[12] M. Jankovic and B. K. Ghosh. Visually guided ranging from observations of points, lines and curves via an identifier based nonlinear observer. *Systems & Control Letters*, 25:63–73, 1995.

[13] F. Kahl. Multiple view geometry and the L_∞-norm. In *International Conference on Computer Vision*, pages 1002–1009. IEEE Computer Society Press, 2005.

[14] D. Karagiannis and A. Astolfi. A new solution to the problem of range identification in perspective vision systems. *IEEE Transactions on Automatic Control*, 50(12):2074–2077, December 2005.

[15] L. Ma, Y. Chen, and K. L. Moore. Range identification for perspective dynamic systems with 3d imaging surfaces. In *American Control Conference*, June 2005.

[16] D. Marzorati, M. Matteucci, D. Migliore, and D. Sorrenti. Monocular slam with inverse scaling parametrization. In *BMVC08*, 2008.

[17] L. Matthies, T. Kanade, and R. Szeliski. Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*, 3:209–236, 1989.

[18] D. Nister. An efficient solution to the five-point relative pose problem. *Int. Conf. Computer Vision and Pattern Recognition*, 2:195–202, 2003.

[19] S. Soatto. 3-d structure from visual motion: Modeling, representation and observability. *Automatica*, 33(7):1287–1312, 1997.

[20] S. Soatto and P. Perona. Reducing structure from motion: A general framework for dynamic vision, part 1: Modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(9), 1998.

[21] S. Soatto and P. Perona. Reducing structure from motion: A general framework for dynamic vision, part 2: Implementation and experimental assessment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(9), 1998.