Robust structure from motion estimation using inertial data

Gang Qian, Rama Chellappa, and Qinfen Zheng

Department of Electrical and Computer Engineering, Center for Automation Research, University of Maryland, College Park, Maryland 20742-3275

Received March 5, 2001; accepted April 30, 2001

The utility of using inertial data for the structure-from-motion (SfM) problem is addressed. We show how inertial data can be used for improved noise resistance, reduction of inherent ambiguities, and handling of mixed-domain sequences. We also show that the number of feature points needed for accurate and robust SfM estimation can be significantly reduced when inertial data are employed. Cramér–Rao lower bounds are computed to quantify the improvements in estimating motion parameters. A robust extended-Kalman-filter-based SfM algorithm using inertial data is then developed to fully exploit the inertial information. This algorithm has been tested by using synthetic and real image sequences, and the results show the efficacy of using inertial data for the SfM problem. © 2001 Optical Society of America

OCIS codes: 100.2960, 150.0150, 350.2660.

1. INTRODUCTION

Structure-from-motion (SfM) estimation is a basic and crucial problem in computer vision. (SfM refers not only to scene structure estimation but also to recovery of the relative motion between camera and scene. In this paper, we assume that only the camera is moving.) SfM estimation using sparse feature correspondences has been investigated for nearly 25 years. During the period covering the mid-1970s to the mid-1980s, two-frame-based approaches were pursued.[1] Since the mid-1980s, long-sequence-based approaches[2–4] have been researched, in addition to research based on two and three views. For a critical review of these methods, see Refs. 5 and 6. Applications of long-sequence-based methods to model building, postproduction, and Moving Picture Experts Group applications are lucidly summarized in Ref. 4. Despite limited successes in these specialized applications, it is our view that in general the SfM problem using sparse features remains difficult. Reliably extracting and tracking features in video as well as in infrared images is a challenge; accurate and robust estimation of arbitrary camera ego-motion given noisy sparse correspondences is another challenge. From a computational point of view, real-time SfM algorithms using sparse features are possible for special cases only. Some of these problems could be solved by using multiple cameras[7] and adding inertial information. This paper will address the latter approach.

Integration of inertial and visual systems has been addressed in Refs. 8 and 9 for the calibration of inertial sensors (linear accelerometers and angular rate sensors) and the navigation of a wheel-driven robot coupled with a pair of stereo cameras and inertial sensors, respectively. In Ref. 10, the recovery of object shape and camera motion using a camera and a gyro sensor is discussed. The fusion of inertial data in a general SfM framework with a monocular camera has not been investigated in detail. Intuitively, by the addition of inertial data, the performance of SfM estimation should be improved. However, to obtain a clear understanding of the utility of using inertial data in the SfM problem, the following questions have to be answered: (1) How can the additional inertial information be used effectively? (2) What benefits, if any, can be achieved by using inertial data; i.e., what are the problems that the inertial data can help solve effectively, and what are the problems in which no significant improvements are obtained? In this paper, we discuss these issues. Most of the discussion is based on the instantaneous rotational rate of the camera obtained from a three-axis camera-mounted rate sensor.

For camera ego-motion estimation, we treat the inertial data as additional measurements to sparse feature correspondences. It can be naturally fed into an extended Kalman filter (EKF), together with feature correspondences, to estimate camera motion and scene structure. We show that the inertial data can play an important role in improving resistance to tracking noise and reducing inherent ambiguities. We also show that a smaller number of feature points are sufficient for robust recovery of camera ego-motion when inertial data are available. In addition, another important consequence of using the inertial data is that our EKF-based inertial-data-driven SfM algorithm is capable of robustly recovering camera ego-motion from image sequences belonging to different domains and also of robustly processing mixed-domain sequences (mixed-domain sequences are the sequences containing both small and large camera translation). In Ref. 5, the robust processing of mixed-domain sequences is proposed as a challenging problem in future SfM algorithm development. The Cramér–Rao lower bounds (CRLBs) for the motion and scene structure of parameters from image sequences with arbitrary camera ego-motion are derived and evaluated to quantify the improvement as a result of using inertial data.

The organization of the paper is as follows. In Section 2, we present a method for incorporating the inertial data

0740-3232/2001/122982-16$15.00

into an EKF-based SfM estimator. In Section 3, the CRLB is derived and evaluated to analyze the algorithm's performance and quantify the reduction in the ill effects of errors in correspondences and the number of feature points needed for accurate SfM estimation. The ambiguity reduction problem is discussed in Section 4. In this section, we also show that our algorithm using inertial data is capable of robustly processing mixed-domain image sequences. Experimental results using real image sequences are given in Section 5. Finally, conclusions are in Section 6.

2. FUSION OF INERTIAL DATA IN THE STRUCTURE-FROM-MOTION PROBLEM

In this section, we discuss a strategy for including inertial data in the SfM problem. After a brief review of camera motion and imaging models, we develop an EFK-based SfM algorithm using the inertial data and point correspondences.

A. Inertial Data Acquisition and Fusion Strategy

In our approach, the inertial data that we exploit are the noisy rotational angular rates of the camera (they are called rate data in the rest of this paper). The rate data are sampled and time stamped on the video frame corresponding to the same time instant and embedded at the bottom of that frame.[11] Each component of the rate data is coded by 12 bits. We assume that the calibration of inertial sensors has been done off line, and readers are referred to Ref. 8 for more details about sensor calibration. Given the image sequence with bar codes of rate data, the values of the rate data can be recovered by using a simple decoding algorithm. In practice, the inertial rate data are corrupted by noise. The measurement equations for the inertial rate data can be written as

$$\begin{aligned} \tilde{\omega}_x &= \omega_x + n_x, \\ \tilde{\omega}_y &= \omega_y + n_y, \\ \tilde{\omega}_z &= \omega_z + n_z, \end{aligned} \tag{1}$$

where $\Omega = (\omega_x, \omega_y, \omega_z)^{\mathrm{T}}$ is the camera rotational rate vector and $n_\Omega = (n_x, n_y, n_z)^{\mathrm{T}}$ is the measurement noise. Theoretically, the measurement noise in the rate data is biased, since the rate sensor has a drift, typically 3×10^{-4} rad/s.[12] However, in real applications of using the rate data to help recover camera motion and scene structure from video sequences, since the videos used are usually captured in a very short period of time (several seconds), the drift and the errors in the recored rate data are very small. Hence, in our implementation, an additive white Gaussian noise (AWGN) with zero mean and standard deviation (STD) of 0.01 rad/s is used to model the noise in rate data.

Given the rate data, how can they be effectively used in the SfM problem? Various Kalman-filter-based algorithms have been proposed[2,3,13] to recursively estimate scene structure and camera ego-motion from image sequences. At each step, the previous structure estimates are fused with current feature correspondences to refine the structure estimates, and their contributions to the new estimates are characterized and weighted by the corresponding covariance matrices. As the rate data directly measure the rotation dynamics of the camera, we treat them as another set of measurements or observations for estimating the camera ego-motion. Hence they can be used in the existing Kalman-filter-based algorithms, as illustrated in Fig. 1.

B. Extended-Kalman-Filter-Based Structure-from-Motion Algorithm Using Inertial Data

In our approach, we mainly follow Ref. 13 for camera and structure parameterization. For the sake of completeness, we briefly describe camera motion and imaging models below. Figure 2 illustrates the imaging model of a moving camera. Two three-dimensional (3D) coordinate systems are used. I is an inertial world coordinate system, fixed on the ground, and C is a camera-fixed coordinate system that uses the image plane as its XY plane. These two coordinate systems are coincident at the beginning time, say t_1, and when the camera moves, I remains on the ground and C moves along with the camera. As shown in Fig. 2, the camera motion between two image frames can be uniquely decomposed into a rotation about F, the focus point, and a translation of F. O is the center of the optical lens. In Ref. 13, the focal length of the camera is also estimated simultaneously. In our setup, we assume that the focal length or the field of view (FOV) of the camera is known.

Camera motion model. Because of the unavailability of the knowledge about translational dynamics, only the T_k, the 3D camera positions in the world inertial system, are used as the translational motion parameters. A random-walk model is used to represent the translational dynamics. Hence the translation equation is given by

$$T_{k+1} = T_k + n_T, \tag{2}$$

where n_T represents additive noise.

To reduce the impact of the nonlinearity in the rotation dynamical equation and make the linearization more suitable for the Kalman filter, we use only the angular velocity Ω as the rotational state parameter. The dynamical equation of the angular velocity Ω is

$$\Omega_{k+1} = \Omega_k + n_\Omega, \tag{3}$$

where n_Ω is the random disturbance of rotation velocity. The global rotational angle vector $\mathbf{\Psi} = (\psi_x, \psi_y, \psi_z)^{\mathrm{T}}$ is updated as follows. Let $R(\mathbf{\Psi}) = \mathcal{R}(\mathbf{\Psi}, t)|_{t=1}$ be the rotation matrix generated by $\mathbf{\Psi}$, with $\mathcal{R}(\mathbf{\Psi}, \tau)$ defined by

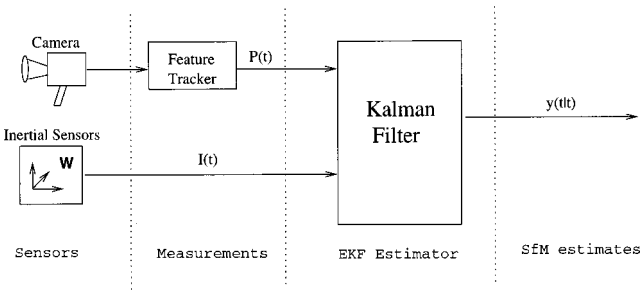


Fig. 1. Kalman-filter-based SfM algorithm for fusing feature correspondences and inertial data.

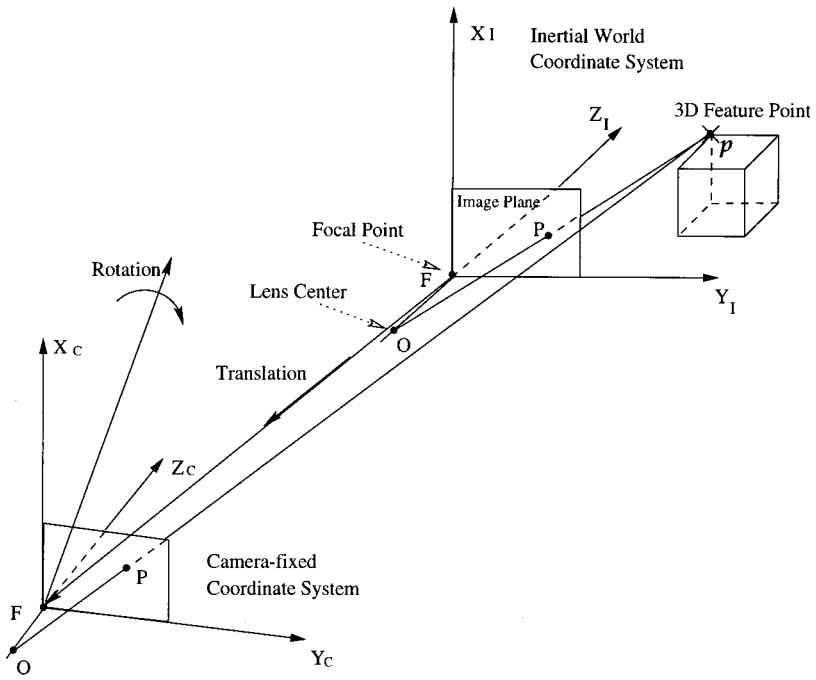


Fig. 2. Imaging model of a moving camera.

$$\mathcal{R}(\boldsymbol{\Psi}, \tau) = \begin{bmatrix} n_1^2 + (1 - n_1^2)\eta & n_1 n_2(1 - \eta) + n_3\zeta & n_1 n_3(1 - \eta) - n_2\zeta \\ n_1 n_2(1 - \eta) - n_3\zeta & n_2^2 + (1 - n_2^2)\eta & n_2 n_3(1 - \eta) + n_1\zeta \\ n_1 n_3(1 - \eta) + n_2\zeta & n_2 n_3(1 - \eta) - n_1\zeta & n_3^2 + (1 - n_3^2)\eta \end{bmatrix}, \quad (4)$$

where $n = (n_1, n_2, n_3)^{\mathrm{T}} = \Psi/|\boldsymbol{\Psi}|$ is the direction cosine vector, $\theta = \tau|\boldsymbol{\Psi}|$, $\zeta = \sin\theta$, and $\eta = \cos\theta$. At first, $R(\boldsymbol{\Psi})$ is updated by

$$R(\boldsymbol{\Psi}_{k+1}) = \mathcal{R}(\Omega_k, t_{k+1} - t_k)R(\boldsymbol{\Psi}_k); \quad (5)$$

then $\boldsymbol{\Psi}_{k+1}$ is computed from $R(\boldsymbol{\Psi}_{k+1})$ by using

$$\boldsymbol{\Psi}_{k+1} = \frac{\phi}{2\sin\phi}(r_{23} - r_{32}, r_{31} - r_{13}, r_{12} - r_{21})^{\mathrm{T}}, \quad (6)$$

where $\phi = \cos^{-1}(\frac{1}{2}\{\mathrm{tr}[R(\boldsymbol{\Psi}_{k+1})] - 1\})$ and r_{jk} is the element in the jth row and the kth column of $R(\boldsymbol{\Psi}_{k+1})$.

Camera imaging model. Assume that L feature points are detected and tracked through an image sequence captured by a moving camera observing a static 3D scene. Denote the 3D coordinates of a feature point p in the world system I by (X, Y, Z). At time t_k, because of camera motion, in the camera system C, p has coordinates (X_k^c, Y_k^c, Z_k^c), given by

$$\begin{pmatrix} X_k^c \\ Y_k^c \\ Z_k^c \end{pmatrix} = R(\boldsymbol{\Psi}_k)\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} - T_k. \quad (7)$$

Denote $(\tilde{u}_k, \tilde{v}_k)$ as the feature tracking result of a feature point in the kth image frame. It is a noisy measurement of the perspective projection of that feature point; i.e.,

$$\tilde{u}_k = \frac{X_k^c}{1 + \beta Z_k^c} + \eta_x, \qquad \tilde{v}_k = \frac{Y_k^c}{1 + \beta Z_k^c} + \eta_y, \quad (8)$$

where β is the inverse of the focal length of the camera. η_x and η_y are noise variables characterizing errors in feature tracking.

To reduce the size of the solution space and make the Kalman filter more stable, we use (l_u, l_v), the two-dimensional (2D) coordinates of the feature points in the first image frame, to approximately describe the direction of location of the feature point. They are called direction of features in Ref. 13. The 3D structure parameters are described as

$$X = (1 + \alpha\beta)(l_u + b_u),$$
$$Y = (1 + \alpha\beta)(l_v + b_v),$$
$$Z = \alpha, \quad (9)$$

where $B = (b_u, b_v)$ is the measurement bias of the direction of location of the feature point. Hence (b_u, b_v, α) can be used to represent the 3D structure of a feature point, instead of (X, Y, Z). Because the dynamic range of the measurement bias is small, a small value can be used to initialize the associated terms in the estimated covariance matrix, which means that the size of the solution space of the 3D parameters is reduced in the X and Y dimensions.

Issue of scaling factor. As we use a perspective projection imaging model, the absolute translation and structure information can be recovered only up to a scale factor if only a monocular camera is used. For this reason, all the translational motion parameters and structure pa-

rameters are normalized with respect to one of these parameters. In our approach, α_L, the z component of the structure parameter of one of the feature points, is used as the unit length for normalization. Without disturbance of the camera calibration, the inverse focal length β is also evaluated when the virtual image film has a width of unit length.

Extended Kalman filter fusing. After the camera motion, inertial, and imaging models have been set up properly, the following parameters make up the state vector:

$$\mathbf{x}_k = (T_k, \Omega_k, b_u^{(1)}, b_v^{(1)}, \alpha_1, \ldots, \alpha_{L-1}, b_u^{(L)}, b_v^{(L)})^{\mathrm{T}}. \tag{10}$$

By using the state equations (2) and (3) and the measurement equations (1) and (8), we can easily implement a standard EKF including rate data as noisy measurements. In the implementation of the EKF, once the Jacobian matrix of the system is obtained, the remaining part is quite straightforward. A full treatment of EKFs can be found in Ref. 14. In the following, we address the computation of the system Jacobian matrix. Let $\mathcal{H}_k$ be the Jacobian matrix at time t_k; i.e.,

$$\mathcal{H}_k = \left(\frac{\partial u_1}{\mathbf{x}_k}, \frac{\partial v_1}{\mathbf{x}_k}, \ldots, \frac{\partial u_L}{\mathbf{x}_k}, \frac{\partial v_L}{\mathbf{x}_k}, \frac{\partial \Omega_k}{\mathbf{x}_k} \right)^{\mathrm{T}}, \tag{11}$$

where $\{(u_l, v_l)\}_{l=1}^{L}$ are the image positions of the features with l as the point index. Ω_k is the camera rotational rate at time t_k. To compute $\mathcal{H}_k$, we need to compute the derivatives of (u_l, v_l) with respect to $\mathbf{x}_k$. Let (l_u, l_v) be the direction of features of the lth feature point. Let R_{k-1} be the rotational matrix generated by $\mathbf{\Psi}_{k-1}$, and let R_Ω be the instantaneous rotational matrix generated by Ω_k during the time period between t_{k-1} and t_k. The rotational matrix at time t_k is $R = R_\Omega R_{k-1}$. Let $R^{(i)}$ and $R_\Omega^{(i)}$ be the ith row of R and R_Ω, respectively. After some straightforward algebra, we can obtain the derivative of (u_l, v_l) with respect to (T_k, Ω_k) as

$$\frac{\partial u_l}{\partial T_k} = \frac{1}{D}(-1, 0, \beta u_l)^{\mathrm{T}}, \qquad \frac{\partial v_l}{\partial T_k} = \frac{1}{D}(0, -1, \beta v_l)^{\mathrm{T}}, \tag{12}$$

$$\frac{\partial u_l}{\partial \Omega_k} = \frac{1}{D} \begin{pmatrix} R_{\Omega_{k,x}}^{(1)} - \beta u_l R_{\Omega_{k,x}}^{(3)} \\ R_{\Omega_{k,y}}^{(1)} - \beta u_l R_{\Omega_{k,y}}^{(3)} \\ R_{\Omega_{k,z}}^{(1)} - \beta u_l R_{\Omega_{k,z}}^{(3)} \end{pmatrix} R_{k-1} \begin{bmatrix} \mu & 0 & 0 \\ 0 & \mu & 0 \\ 0 & 0 & 1 \end{bmatrix} S,$$

$$\frac{\partial v}{\partial \Omega_k} = \frac{1}{D} \begin{pmatrix} R_{\Omega_{k,x}}^{(2)} - \beta v_l R_{\Omega_{k,x}}^{(3)} \\ R_{\Omega_{k,y}}^{(2)} - \beta v_l R_{\Omega_{k,y}}^{(3)} \\ R_{\Omega_{k,z}}^{(2)} - \beta v_l R_{\Omega_{k,z}}^{(3)} \end{pmatrix} R_{k-1} \begin{bmatrix} \mu & 0 & 0 \\ 0 & \mu & 0 \\ 0 & 0 & 1 \end{bmatrix} S, \tag{13}$$

where

$$D = 1 - \beta T_{k,z} + \beta R^{(3)}[(1 + \alpha_l \beta)(b_u^{(l)} + l_u), (1 + \alpha_l \beta)(b_v^{(l)} + l_v), \alpha_l]^{\mathrm{T}},$$

$$S = [b_u^{(l)} + l_u, b_v^{(l)} + l_v, \alpha_l]^{\mathrm{T}},$$

$$\mu = 1 + \alpha_l \beta.$$

$R_{\Omega_{k,\{x,y,z\}}}$ are the derivative matrices of R_Ω with respect to $\Omega_{k,\{x,y,z\}}$ and can be easily computed by using the symbolic toolbox in MATLAB. Also, we can obtain the derivatives of (u_l, v_l) with respect to the structure parameters $(b_u^{(l)}, b_v^{(l)}, \alpha_l)$:

$$\frac{\partial (u_l, v_l)}{\partial [(b_u^{(l)}, b_v^{(l)}, \alpha_l)^{\mathrm{T}}]} = \frac{1}{D} \begin{bmatrix} \mu & 0 & 0 \\ 0 & \mu & 0 \\ \beta(b_u^{(l)} + l_u) & \beta(b_v^{(l)} + l_v) & 1 \end{bmatrix} \times \begin{pmatrix} R^{(1)} - u_l \beta R^{(3)} \\ R^{(2)} - v_l \beta R^{(3)} \end{pmatrix}^{\mathrm{T}}. \tag{14}$$

Based on these derivatives, the system Jacobian matrix can be computed and a standard EKF can then be directly realized.

We have extensively tested the performance of the above EKF-based rate-data-driven algorithm by using synthetic and real image sequences. We have also derived the CRLB to analyze the performance. Both experimental and CRLB analysis results show that using the rate data can improve the SfM algorithm. The resulting SfM algorithm has the following features:

- Is more robust to feature tracking errors.
- Requires fewer feature points to accurately recover camera ego-motion.
- Reduces inherent ambiguities in the recovery of camera ego-motion.
- Robustly processes mixed-domain sequences.

We look at these aspects in detail in the following sections.

3. PERFORMANCE ANALYSIS

In this section, we first derive the CRLB for motion and structure parameters with arbitrary camera ego-motion. Then, by using the CRLB and the real estimation results, we show that the inertial data can play an important role in improving resistance to feature tracking errors and reducing the required number of feature points for accurate and robust SfM estimation. We also compare our results with those obtained by using an algorithm proposed by Azarbayejani and Pentland in Ref. 13. (For convenience, in this paper, these two algorithms will be called the A-P algorithm and the R-D algorithm.) In the comparison, it can be seen that the R-D algorithm outperforms the A-P algorithm. Fundamentally, the A-P and R-D algorithms are similar. They both use the EKF to estimate the camera motion and the scene structure. Therefore the superior performance of the R-D algorithm is mostly due to the integration of the rate data. Also, from the comparisons, we gain an understanding as to where and how the rate data can be used to improve the SfM estimates, which is a primary goal of this paper.

A. Cramér–Rao Lower Bound for the Structure-from-Motion Problem with Arbitrary Camera Ego-Motion

CRLBs are often used to indicate the inherent uncertainty of the estimates in an estimation problem.[15] Although the CRLB has been used in the performance analysis of the SfM algorithm, either the measurements were noisy flow fields[16] or the camera motion was assumed to be uniform.[17] The CRLB for the motion and structure parameters has not been derived in the case of arbitrary camera motion. We first derive the CRLB in this case and then use the results to show the improvement that is due to the use of inertial rate data. Let χ be the parameters to be estimated and $\hat{\chi}$ be their unbiased estimate. The error covariance matrix V is bounded by

$$V = E[(\hat{\chi} - \chi)(\hat{\chi} - \chi)^{\mathrm{T}}] \geqslant J^{-1}, \tag{15}$$

where J is the Fisher information matrix, given by

$$J = E\left[\frac{\partial \ln f(\mathbf{z}|\chi)}{\partial \chi}\left\{\frac{\partial \ln f(\mathbf{z}|\chi)}{\partial \chi}\right\}^{\mathrm{T}}\middle|\chi\right], \tag{16}$$

where $f(\mathbf{z}|\chi)$ is the probability density function of observations $\mathbf{z} = (z_1, z_2, \ldots, z_n)^{\mathrm{T}}$ given parameters χ. The matrix inequality $A \geqslant B$ means that matrix $A - B$ is at least semipositive definite.

Cramér–Rao lower bounds of unbiased estimates from observations corrupted by additive Gaussian noise. Assume that the observations are corrupted by additive Gaussian noise. Let U be the covariance matrix of the observation noise, and let H denote the derivative matrix of the observation with respect to the parameter χ; i.e.,

$$H = \left(\frac{\partial h_1(\chi)}{\partial \chi}, \frac{\partial h_2(\chi)}{\partial \chi}, \ldots, \frac{\partial h_n(\chi)}{\partial \chi}\right), \tag{17}$$

where $z_i = h_i(\chi) + n_i$ is the observation equation. After some simple algebra, we obtain

$$J = HU^{-1}H^{\mathrm{T}}. \tag{18}$$

When the measurement noise is white, Eq. (18) can be simplified as

$$J = \sum_{i=1}^{n} \frac{1}{\sigma_i^2}\frac{\partial h_i(\chi)}{\partial \chi}\left[\frac{\partial h_i(\chi)}{\partial \chi}\right]^{\mathrm{T}}. \tag{19}$$

If we denote

$$J_i = J(z_i) = \frac{1}{\sigma_i^2}\frac{\partial h_i(\chi)}{\partial \chi}\left[\frac{\partial h_i(\chi)}{\partial \chi}\right]^{\mathrm{T}}, \tag{20}$$

Eq. (19) can be rewritten as

$$J = \sum_{i=1}^{n} J_i = \sum_{i=1}^{n} J(z_i), \tag{21}$$

where J_i can be viewed as the Fisher information contained in the ith observation of $\mathbf{z}$ on parameter χ. From Eq. (21), we know that if the observation noise is additive white Gaussian and the estimate is unbiased, the Fisher information matrix of the estimate is just the sum of the individual Fisher information obtained from each observation. In the case of white observation noise, it has been shown that additional observations can decrease the CRLB.[16] It is not difficult to show that similar conclusions can also be made when the observations are contaminated by color noise.

Cramér–Rao lower bounds for motion and structure parameters. As the direction of features is used during structure parameterization, the dominant structure parameters are $\{\alpha_i\}$, the depth values of the feature points. Therefore only the CRLB of $\{\alpha_i\}$ is derived. Also, as we are interested more in the pose of the camera at a certain time instant, say t_k, than in its rotational rate, the CRLB for the global rotation angle $\mathbf{\Psi}_k$ is derived instead of the CRLB for the rotational rate Ω_k. If K image frames are captured and L feature points are tracked in each frame, the following parameters are estimated:

$$\chi = (\{\alpha_i\}_{i=1}^{L-1}, T_2, \mathbf{\Psi}_2, \ldots, T_K, \mathbf{\Psi}_K). \tag{22}$$

Since α_L and $(T_1, \mathbf{\Psi}_1)$ are all known [α_L is unity, as it is used as the normalization parameter, and $(T_1, \mathbf{\Psi}_1)$ are all zero, as coordinate systems I and C are coincident at t_1], they are not to be estimated. The resulting Fisher information matrix J is a $(6K + L - 7) \times (6K + L - 7)$ square matrix. We assume that the observation noise is additive white Gaussian. As shown in Eq. (21), J is the sum of the Fisher information matrices J_i obtained from individual measurements $h_i(\chi)$, which can be either feature correspondence or inertial rate data. We will derive J_i for both kinds of observations.

Let (u, v) denote the image position of the lth feature point in the kth image frame. Taking the derivative of (u, v) with respect to χ, we have

$$\frac{\partial u}{\partial \chi} = \left(\ldots, \frac{\partial u}{\partial \alpha_l}, \ldots, \left(\frac{\partial u}{\partial T_k}\right)^{\mathrm{T}}, \left(\frac{\partial u}{\partial \mathbf{\Psi}_k}\right)^{\mathrm{T}}, \ldots\right)^{\mathrm{T}},$$

$$\frac{\partial v}{\partial \chi} = \left(\ldots, \frac{\partial v}{\partial \alpha_l}, \ldots, \left(\frac{\partial v}{\partial T_k}\right)^{\mathrm{T}}, \left(\frac{\partial v}{\partial \mathbf{\Psi}_k}\right)^{\mathrm{T}}, \ldots\right)^{\mathrm{T}}, \tag{23}$$

and all the terms that do not appear in Eqs. (23) are zero, since (u, v) is determined only by α_l and $(T_k, \mathbf{\Psi}_k)$. The derivatives of (u, v) with respect to the translation and structure parameters can be obtained by using Eqs. (12) and (14). The derivatives with respect to the rotation angles $\mathbf{\Psi}_k$ are obtained as follows:

$$\frac{\partial u}{\partial \mathbf{\Psi}_k} = \frac{1}{D}\begin{pmatrix} R_x^{(1)} - \beta u R_x^{(3)} \\ R_y^{(1)} - \beta u R_y^{(3)} \\ R_z^{(1)} - \beta u R_z^{(3)} \end{pmatrix}\begin{bmatrix} \mu & 0 & 0 \\ 0 & \mu & 0 \\ 0 & 0 & 1 \end{bmatrix} S,$$

$$\frac{\partial v}{\partial \mathbf{\Psi}_k} = \frac{1}{D}\begin{pmatrix} R_x^{(2)} - \beta v R_x^{(3)} \\ R_y^{(2)} - \beta v R_y^{(3)} \\ R_z^{(2)} - \beta v R_z^{(3)} \end{pmatrix}\begin{bmatrix} \mu & 0 & 0 \\ 0 & \mu & 0 \\ 0 & 0 & 1 \end{bmatrix} S, \tag{24}$$

where $R_{\{x,y,z\}}$ are the derivative matrices of R with respect to $\Psi_{k\{x,y,z\}}$ and can also be computed by using the symbolic toolbox in MATLAB. Hence, given a certain noise

level in feature correspondences, the Fisher information contained in one feature correspondence can be computed by using Eq. (20).

As $\mathbf{\Psi}_k$ is used for computing the CRLB, the measurement equation for rate data can be alternatively written as

$$\Omega_k = \frac{1}{\Delta t}(\mathbf{\Psi}_{k+1} - \mathbf{\Psi}_k), \tag{25}$$

where Ω_k is the rate data at time instant k and Δt is the time period between t_k and t_{k+1}. Therefore the Fisher information contained in the inertial rate data can be derived directly. Hence the CRLBs for the parameters in Eq. (22) with and without using rate data can both be evaluated. Intuitively, when more information is used, more accurate estimates should be achieved; i.e., the CRLB with more information data should be lower than the CRLB before the inertial information is added. In the rest of this paper, the CRLBs for motion and structure estimates are evaluated to quantitatively illustrate the benefits of using the inertial rate data in the SfM problem.

B. Measurement Error Effects Analysis

Feature tracking errors are inevitable in real SfM applications. Mainly, there are two types of tracking error sources. One type of error is due to image digitization, poor image quality, or the inadequacy of 2D transformation models to represent the image transformation caused by 3D camera motion. Since this type of error is typically small and the tracking of most of the features is uncorrelated, this error can be approximated by using white Gaussian noise.[5,18] Other types of errors are created by the false matches of the feature points. To illustrate the effects of feature tracking errors on SfM estimates, we generated synthetic image sequences with various tracking error levels and types and then used them to test both the A-P and R-D algorithms. The simulation results show that the R-D algorithm is much less sensitive to feature tracking errors than the A-P algorithm.

The configuration of the simulation is as follows. The virtual camera has a FOV of $53.16° \times 53.16°$ and an image size of 512×512 pixels. The camera translates along the X axis with a constant velocity of -0.3 unit length per second. The 3D structures of the feature points are shown in Table 1. Image sequences are synthesized by computing the feature projections in the image plane using the above 3D feature structures and camera ego-motion according to imaging measurement equations (8). Zero-mean AWGN is added to these feature correspondences. Inertial rate data are generated similarly by using Eqs. (1). Also, inertial rate data are corrupted by AWGN with zero mean and STD of 0.01 rad/s.

Table 1. 3D Structure of Feature Points

Feature	l_u	l_v	α
1	−0.100	0.050	1.41
2	−0.097	0.120	0.95
3	−0.100	−0.135	0.80
4	0.100	0.150	1.10
5	0.200	0.100	0.70
6	0.100	0.200	0.90
7	0.140	−0.100	0.94
9	−0.010	0.020	1.85
9	0.071	−0.135	0.84
10	−0.100	0.150	0.90
11	−0.200	0.100	0.50
12	−0.140	−0.200	0.30
13	0.043	−0.029	1.00

The sensitivity of the two algorithms to the tracking errors is compared by using three noise levels. The STDs of the noise variables are, respectively, 0.5, 2, and 4 pixels. Figure 3 shows the synthetic image sequences and the motion estimation results using these three noise levels. It can be seen that as the noise level increases, the performance of the A-P algorithm deteriorates much faster than that of the R-D algorithm. Figure 4 shows the CRLBs for the motion parameters. It can be seen that along with the increase in feature tracking noise, the CRLB for each parameter increases, while the CRLB with rate data increases much slower than that without rate data. This indicates that although the large feature tracking noise will make the motion estimates worse, the use of rate data can compensate for the effect of large tracking noise and keep the bias in estimation low.

It can be noted that after a certain time, the CRLBs without inertial rate data decrease when more observations are used. However, this does not imply that the A-P algorithm can converge to the correct value eventually. We know that the CRLBs are theoretical lower bounds for the parameters of interest given a set of observations. The CRLBs shown in Fig. 4 should not depend on any specific algorithms. Although we see that the plots of CRLBs go down as more frames are used even without rate data, it is difficult to find a recursive estimator that can converge in spite of large estimation errors at previous time instants. It is possible to find a batch estimator that uses all the currently available observations and get an accurate result. However, it will require many more image frames and much higher computational cost to effectively use all the observations, jeopardizing the possibility of real-time processing of the data. By using the inertial rate data, a simple EKF can provide good results in a recursive fashion with very low computational load.

Since the rate data are corrupted by measurement noise, if this noise is too large, rate data may lose control of the estimates when significant errors in feature tracking are also present. For a set of motion parameters, Δ_S can be used to denote the CRLB in a standard case when the rate data are not used and the feature tracking noise is AWGN with zero mean and unit variance. Assume that the STD of the feature tracking noise increases to σ_t (usually, $\sigma_t > 1$). An interesting task is to find the largest STD σ_{in} of the rate data noise such that the rate data can still possibly help the algorithm obtain estimates with the same accuracy as those in the standard case, such that the CRLBs are equal. For a given σ_t, we will compute σ_{in} such that

$$\text{CRLB}(\sigma_t, \sigma_{\text{in}}) = \Delta_S. \tag{26}$$

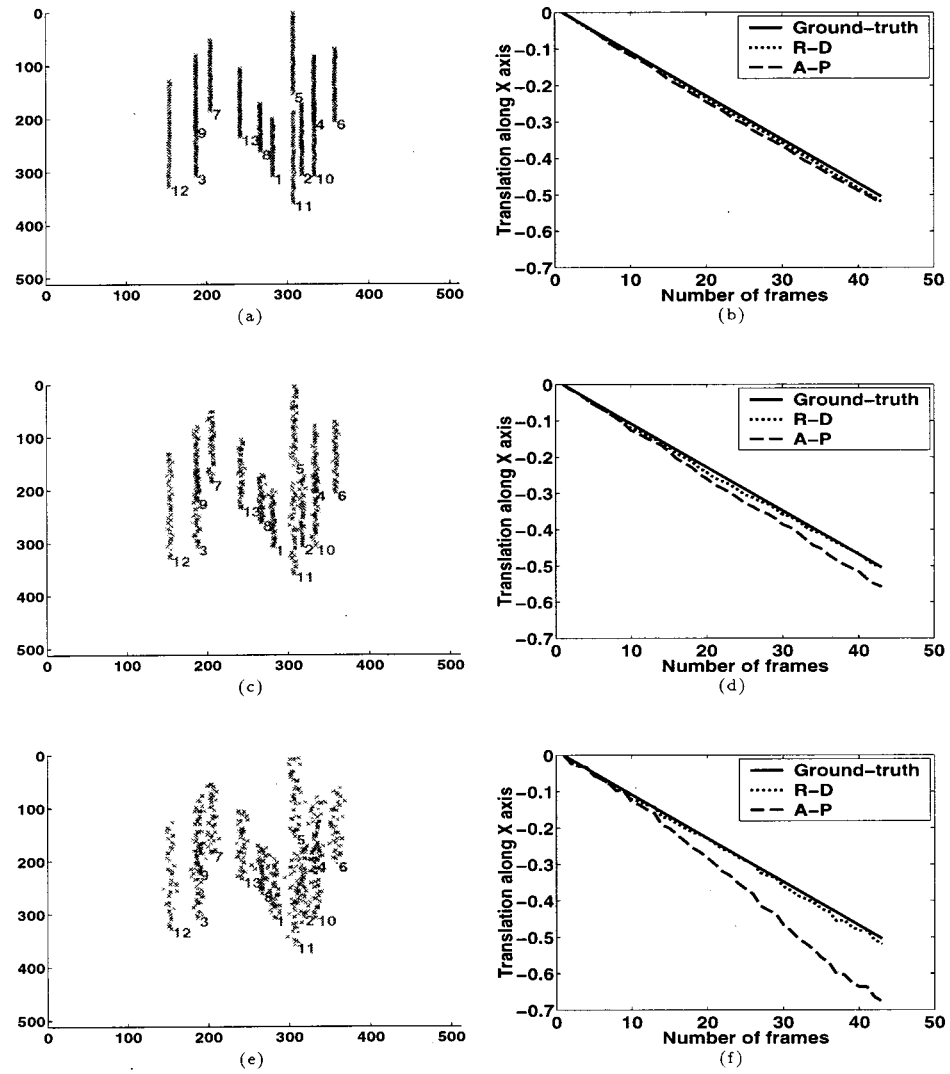


Fig. 3. Performance comparison of the R-D and A-P algorithms under various noise levels. Plots (a), (c), and (e) are the feature trajectories corrupted by AWGN with STDs of 0.5, 2, and 4 pixels, respectively, and plots (b), (d), and (f) are the corresponding motion estimates (translation along the X axis). It can be observed that when the noise level increases, the performance of the A-P algorithm deteriorates much faster than that of the R-D algorithm, which is affected only slightly by the tracking noise level.

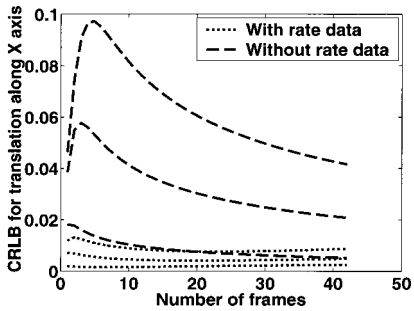


Fig. 4. CRLBs for the X translation with (dotted curves) and without (dashed curves) using inertial rate data. CRLBs with three levels of feature tracking errors are shown here. It can be observed that although the CRLBs in all cases increase with the increase in tracking noise, the CRLBs with rate data are affected only very slightly while the CRLBs without rate data increase very rapidly.

In fact, Δ_S = CRLB (1 pixel, ∞), since no use of inertial data is equivalent to using rate data corrupted by noise with infinite variance. Figure 5 shows plots of σ_t versus σ_{in} characteristic for each of the motion parameters when the camera moves with a uniform translation velocity of $(-0.1, 0.1, 0)^{\text{T}}$ unit length per second and a uniform rotational rate of $(0.4, 0.3, 0.1)^{\text{T}}$ rad/s. $W = (\sigma_t, \sigma_{\text{in}})$ can be viewed as an operating point of the EKF motion estimator. For operating points on the curves of Fig. 5, a similar performance will be expected. For operating points below the curves, we may expect better performance than that in the standard case. When W is located in that region, the rate data and feature tracking noise are in such a balance that they can help each other achieve good performance. For operating points above the curves, we have worse estimates than those in the standard case, since the variances of the noise variables are too large and the balance between feature tracking and inertial measurement noise does not exist any longer.

Robustness to mismatched feature points. Mismatching of feature points is another source of measurement errors in the SfM problems. It happens when two or several features with similar appearance are located in image frames close to each other. This makes the feature

tracker easily track a wrong feature point instead of the correct one. The errors created by mismatched feature points cannot be statistically modeled by using Gaussian random variables.[18] In the A-P algorithm, the EKF uses only the feature correspondences as measurements. The motion and structure parameters are found so that the feature reprojection error is minimized. With mismatched features, solutions of the motion parameter away from the ground truth might produce a lower reprojection error than those close to the ground truth. Therefore mismatched feature correspondences can dramatically deteriorate the motion estimates obtained by using the A-P algorithm. However, in the R-D algorithm, since the rate data are used as additional measurements, more constraints are introduced into the estimation of the motion parameter. The use of rate data can greatly help alleviate the ill effects of mismatched features on motion estimates. Figure 6 shows the simulation results obtained by using feature correspondences with and without mismatched points. The features used in this experiment are those listed in Table 1. In the simulation with mismatched features, points 2 and 10 are mismatched from the third frame. In Fig. 6, the values of the ground truth are plotted by using solid lines, and the motion estimates without mismatching using A-P and R-D algorithms are plotted by using dashed and dotted curves, respectively. The motion estimates with mismatching using the A-P algorithm are marked by plus signs, and those obtained by using the R-D algorithm are marked by circles. We can see that when no features are mismatched, the motion estimates obtained by using the A-P and R-D algorithms are close to the ground truth, although the R-D algorithm gives better results. However, when points 2 and 10 are mismatched, the motion estimates from the A-P algorithm diverge from the ground truth rapidly and, at the same time, mismatching of feature points affects the motion estimates obtained from the R-D algorithm only slightly.

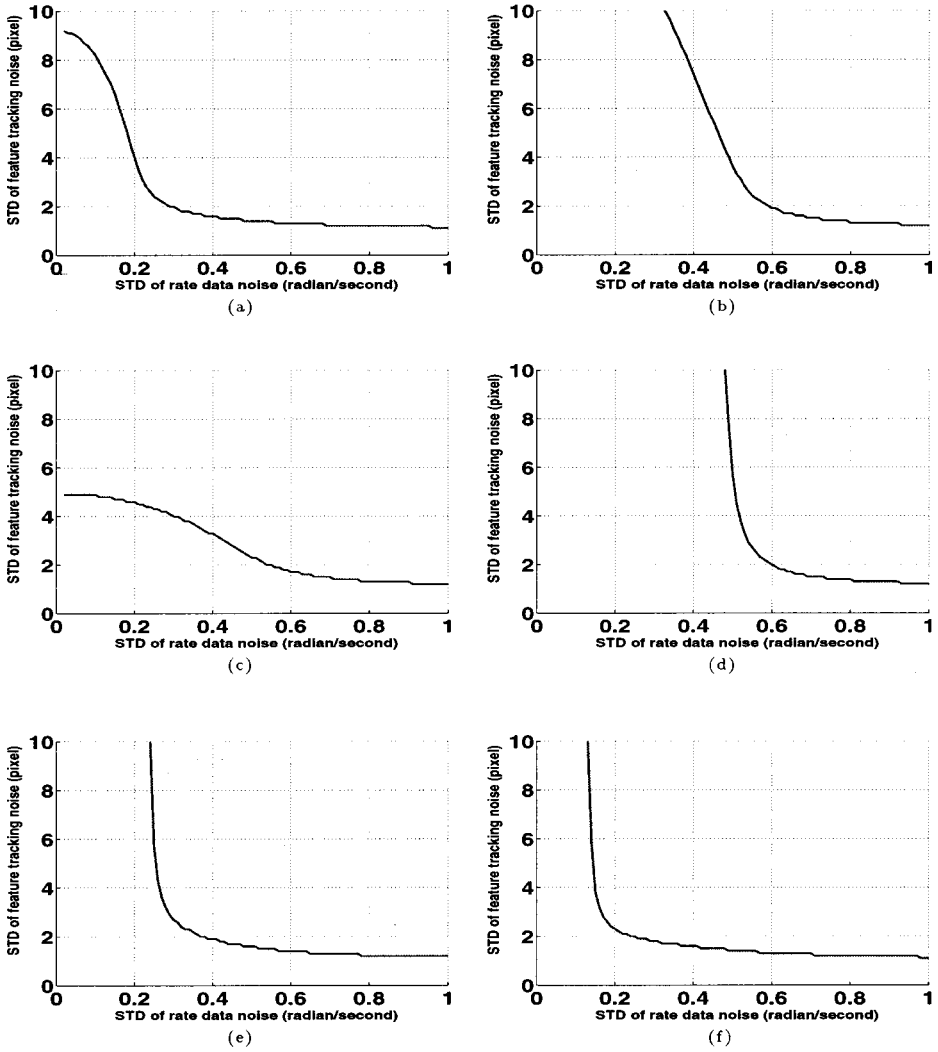


Fig. 5. $(\sigma_t, \sigma_{\text{in}})$ pairs that generate the same CRLBs as the standard CRLB produced by (1 pixel, ∞). The camera moves with a uniform translational velocity of $(-0.1, 0.1, 0)^{\text{T}}$ unit length per second and a uniform rotational rate of $(0.4, 0.3, 0.1)^{\text{T}}$ rad/s. The abscissa (horizontal axis) is the STD of the inertial data σ_{in}, ranging from 0.01 to 1 rad/s. The ordinate (vertical axis) is the STD of the tracking noise, ranging from 0 to 10 pixels. Plots (a)–(f) are for the translation and global rotation about the *X*, *Y*, and *Z* axes, respectively.

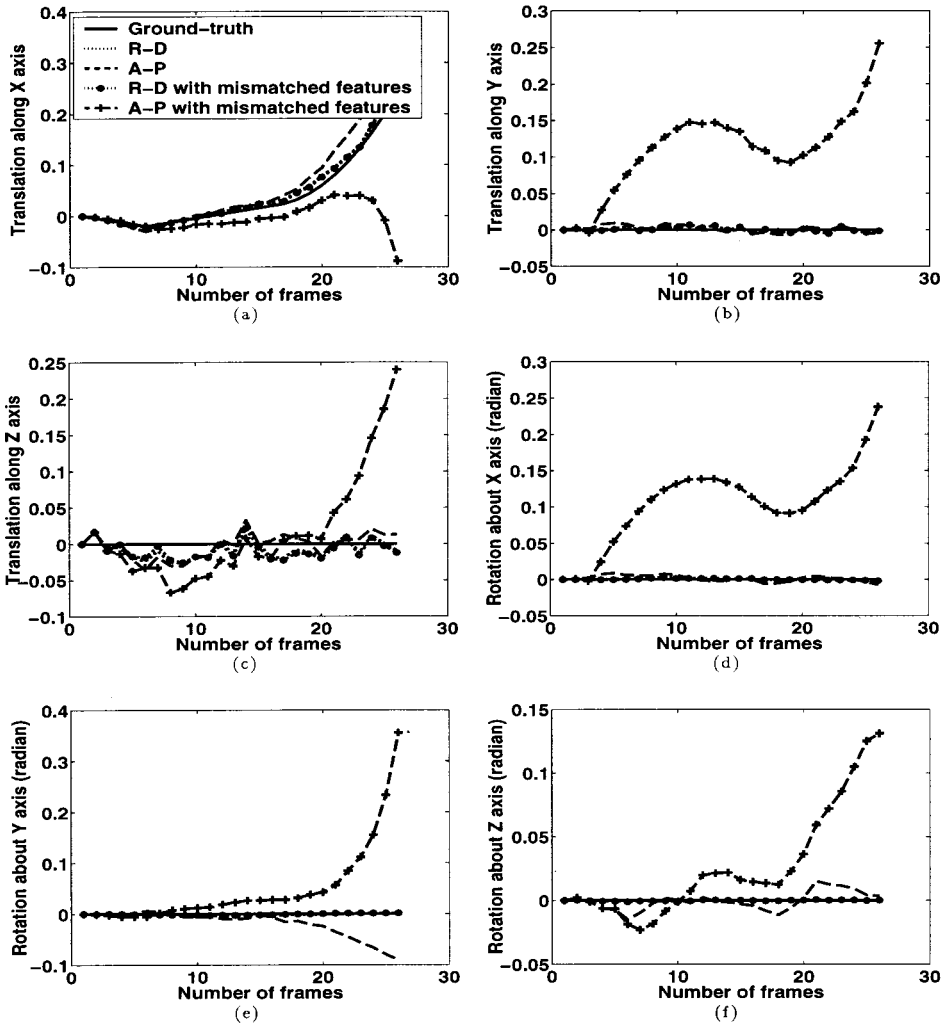


Fig. 6. Camera ego-motion estimates using feature correspondences with mismatched feature points. Plots (a), (b), and (c) are the estimates of translation along the *X, Y*, and *Z* axes, respectively. Plots (d), (e), and (f) are the estimates of global rotation about the *X, Y*, and *Z* axes, respectively.

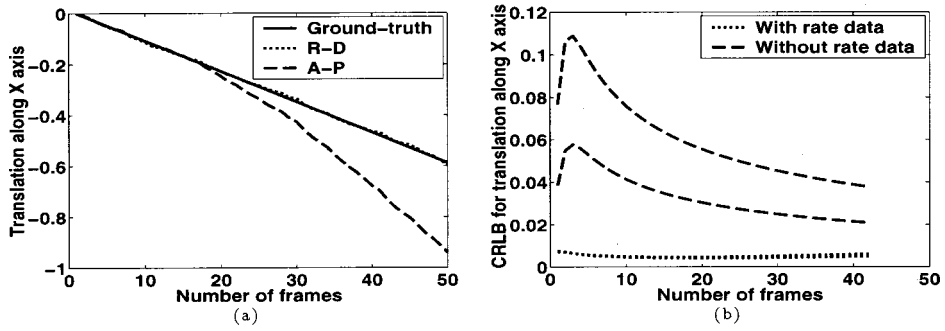


Fig. 7. (a) Camera ego-motion estimates using only seven feature points. Compared with the estimate shown in Fig. 3(d), where 13 features were used, this estimate shows that the A-P algorithm drifts away from the ground truth quickly when only seven features were tracked although the tracking noise level was the same (AWGN with STD of 2 pixels). However, the results obtained using the R-D algorithm are still very close to the ground truth and hardly affected by the reduction in the number of feature points.

C. Reduction in Number of Features

Although seven points are sufficient to recover arbitrary camera motion from two perspective views,[19] this is often not enough for accurate and robust SfM estimation as a result of errors in feature correspondences in real applications. The redundancy resulting from using more feature points is commonly explored to achieve accurate and robust SfM estimation; e.g., 26 feature points were used in the simulation results of the A-P algorithm given in Ref. 13. Tracking this number of feature points is not realistic in many uncontrolled real applications, especially when the video quality is poor.

In our research, we found that adding inertial information can dramatically increase the number of feature points needed for accurate and robust SfM estimation. Figure 7(a) shows the motion estimates obtained by using

the A-P and R-D algorithms when only seven of the above feature points were tracked. They were feature points 1, 3, 4, 9, 10, 11, and 13 in Table 1. The STD of the feature tracking errors was 2 pixels. From Fig. 7(a), we can see that the results obtained by using the A-P algorithm are unacceptable when only seven points are tracked while the results from the R-D algorithm are still very accurate. We also computed the CRLBs for the motion parameters in this case, which are shown in Fig. 7(b). We see that the number of feature points significantly affects the CRLBs of the motion parameters when no rate data are used.

4. AMBIGUITY REDUCTION

In this section, we show how inertial rate data can be applied to reduction of ambiguity in the general SfM problem, including SfM estimation from noisy flow fields and from sparse feature correspondences. Ambiguities in 3D motion recovery from noisy flow fields have been reported by many researchers.[16,20,21] Because of the observation noise, under many circumstances, multiple admissible camera motion and scene structure interpretations exist for a given noisy flow field. One dominant ambiguity arises from the similarity between the flow fields generated by translation parallel to the image plane and associated rotation[21] when the size of the FOV is small. Since this translation–rotation confusion is inherent, actively fixating on the focus of expansion is suggested in Ref. 21 to keep the lateral translation as small as possible, so that this ambiguity can be reduced. One alternative way to eliminate this translation–rotation confusion is to exploit inertial rate data obtained from a camera-mounted rate sensor. If the inertial rate data are used, the flow field generated by pure rotation can be roughly predicted, and hence the flow field generated by pure camera translation can be computed by subtracting the rotational flow field from the original observed flow field. Based on this noisy pure translational flow field, camera translation and 3D scene structure can be robustly estimated.[22] The translation–rotation confusion can be completely eliminated when the inertial data are accurate.

Although many batch and recursive estimators have been proposed for the problem SfM using a set of noisy feature correspondences from long sequences, there is no guarantee that these methods can find a correct solution. In this section, we show that multiple admissible solutions exist for some sequences when the size of the FOV is small under a certain measurement noise level. We also show that inertial rate data can be used to reduce the admissible solution space.

A. Translation–Rotation Ambiguity Elimination from Structure-from-Motion Estimation Using a Noisy Flow Field

The flow field is related to the 3D camera motion and feature structures by the well-known flow equation[23]:

$$u(x,y) = u_t(x,y) + u_r(x,y),$$
$$v(x,y) = v_t(x,y) + v_r(x,y), \tag{27}$$

where (u_t, v_t) and (u_r, v_r) are, respectively, the translational and rotational components of the flow field, given by

$$u_t(x,y) = (xt_z - t_x)\frac{1}{z(x,y)},$$
$$v_t(x,y) = (yt_z - t_y)\frac{1}{z(x,y)}, \tag{28}$$
$$u_r(x,y) = xy\omega_x + (1+x^2)\omega_y + y\omega_z,$$
$$v_r(x,y) = (1+y^2)\omega_x - xy\omega_y - x\omega_z, \tag{29}$$

where (t_x, t_y, t_z) is the camera translation and $(\omega_x, \omega_y, \omega_z)$ is the camera rotation. In practical applications, the observed flow field is corrupted by additive noise:

$$\tilde{u}(x,y) = u(x,y) + n_u, \qquad \tilde{v}(x,y) = v(x,y) + n_v, \tag{30}$$

where n_u and n_v are white noise with covariances σ_u^2 and σ_v^2, respectively.

Assume that the noises in the rate data, (n_x, n_y, n_z) in Eqs. (1), are white with covariances σ_x^2, σ_y^2, and σ_z^2, respectively. When the inertial data are used, the flow field generated by camera rotation can be predicted by

$$\tilde{u}_r(x,y) = xy\tilde{\omega}_x - (1+x^2)\tilde{\omega}_y + y\tilde{\omega}_z = u_r(x,y) + n_{u,r}, \tag{31}$$

and, similarly, we have

$$\tilde{v}_r(x,y) = v_r(x,y) + n_{v,r}, \tag{32}$$

where

$$n_{u,r} = xyn_x - (1+x^2)n_y + yn_z,$$
$$n_{v,r} = (1+x^2)n_x - xyn_y + xn_z$$

are the prediction errors of the rotational flow field with covariances given by

$$\sigma_{u,r}^2 = (xy)^2\sigma_x^2 + (1+x^2)^2\sigma_y^2 + y^2\sigma_z^2,$$
$$\sigma_{v,r}^2 = (xy)^2\sigma_y^2 + (1+y^2)^2\sigma_x^2 + x^2\sigma_z^2. \tag{33}$$

A noisy translational flow field can be computed by subtracting the predicted rotational flow field from the originally observed flow field;

$$\tilde{u}_t(x,y) = \tilde{u}(x,y) - \tilde{u}_r(x,y) = u_t(x,y) + n_{u,t}, \tag{34}$$

and, similarly, we have

$$\tilde{v}_t(x,y) = v_t(x,y) + n_{v,t}, \tag{35}$$

where $n_{u,t}$ and $n_{v,t}$ are the observation noise for the pure translational flow field with covariances $\sigma_{u,t}^2$ and $\sigma_{v,t}^2$, respectively, and

$$\sigma_{u,t}^2 = \sigma_u^2 + (xy)^2\sigma_x^2 + (1+x^2)^2\sigma_y^2 + y^2\sigma_z^2,$$
$$\sigma_{v,t}^2 = \sigma_v^2 + (xy)^2\sigma_y^2 + (1+y^2)^2\sigma_x^2 + x^2\sigma_z^2. \tag{36}$$

Once the noisy translational flow field and the associated measurement noise characterization are obtained, trans-

Table 2. Ground Truth and Two Estimates of Feature Point Structure

Feature	l_u	l_v	α	α_t	α_f
1	−0.10	0.05	0.71	0.7206	2.4283
2	−0.097	0.12	0.45	0.4553	4.7040
3	−0.1	0.135	0.7	0.7219	2.4248
4	0.1	−0.15	0.5	0.5113	3.9169
5	0.2	0.1	0.6	0.6105	2.8635
6	0.1	0.1	0.9	0.8908	1.3974
7	0.14	−0.1	0.3	0.3035	6.4214
8	0.07	0.12	0.85	0.8561	1.5853
9	0.071	−0.135	0.34	0.3496	5.9230
10	−0.1	0.15	0.4	0.3964	5.4324
11	0.1	0.25	0.5	0.4910	3.9650
12	−0.2	0.1	0.8	0.7968	1.9762
13	−0.14	0.1	0.3	0.3029	7.0546
14	0.043	0.129	1.0	1.0000	1.0000

lation and 3D scene structure can be robustly estimated[22] when the inertial data are accurate; i.e., the inertial data observation noise covariances (σ_x^2, σ_y^2, σ_z^2) are small.

B. Ambiguity Reduction for Structure-from-Motion Estimation from Multiple Views

In this subsection, we synthesize an image sequence and show that at least two admissible solutions exist for this image sequence. Here, admissible solutions are defined as solutions that produce cost comparable with that associated with the true solution.

In this simulation, the image size is kept as 512 × 512 pixels, while the FOV of the camera is reduced to 11.43° × 11.43°. We let the virtual camera translate along the X axis with a constant velocity of −0.2 unit length per second and rotate about the Y axis with a con-

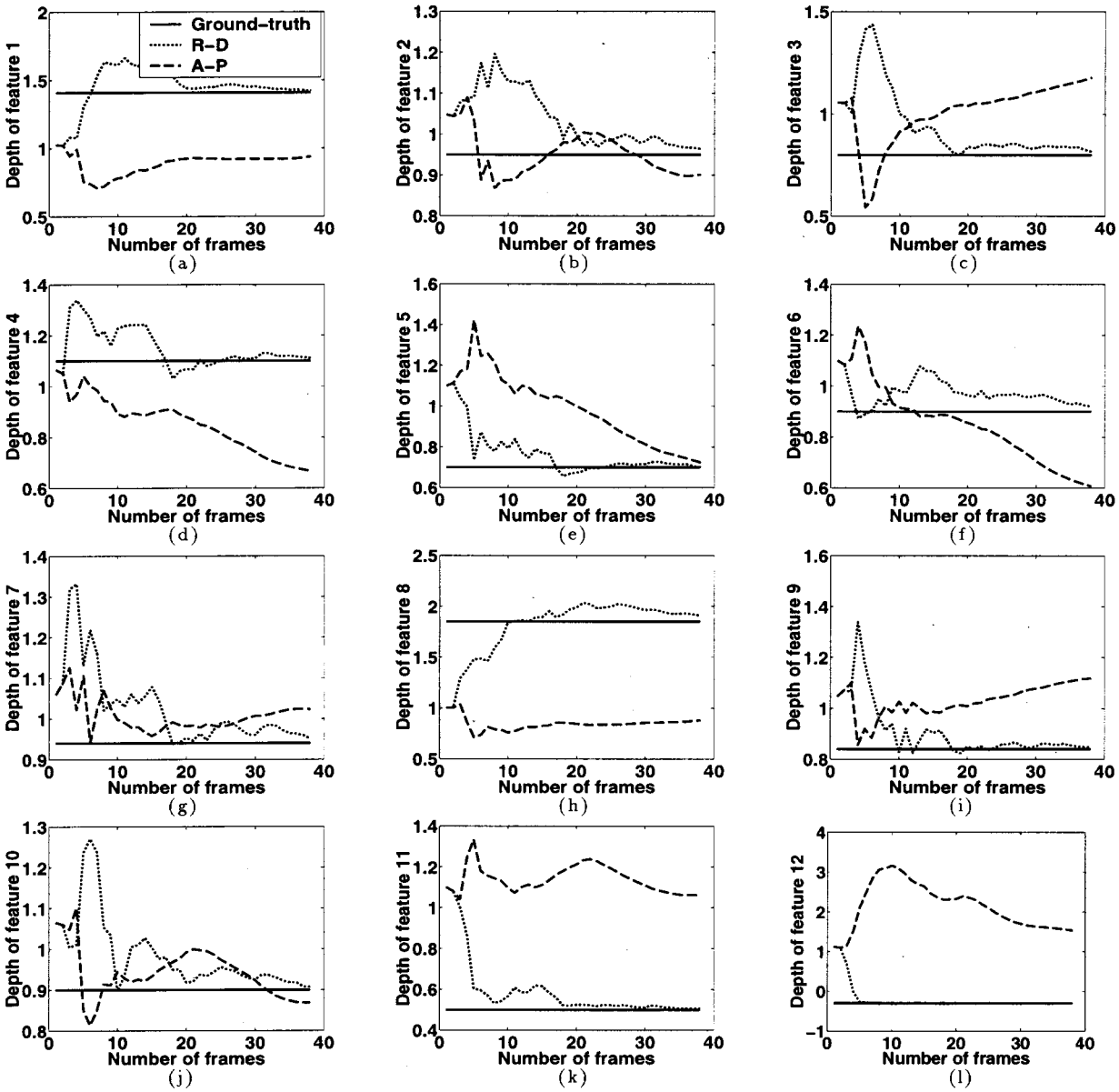


Fig. 8. Depth estimates using a mixed-domain sequence. Plots (a)–(l) are the depth estimates of feature points 1–12, respectively.

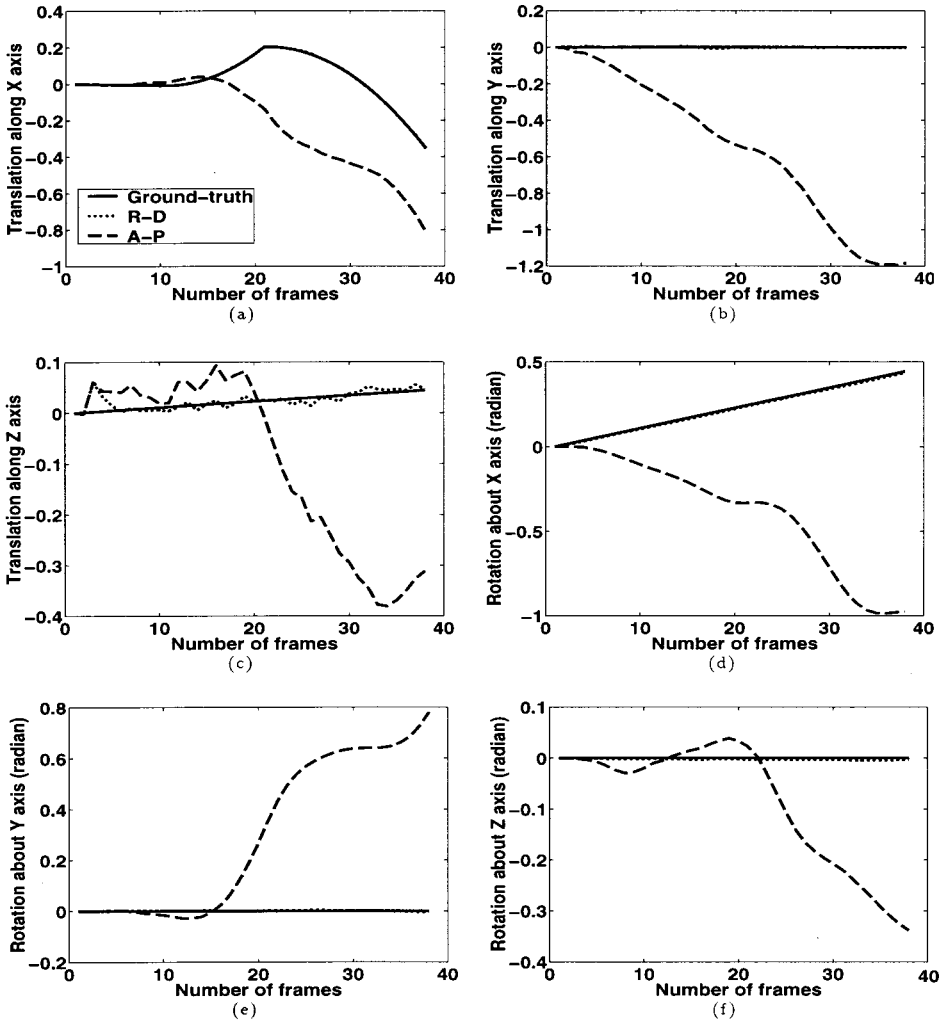


Fig. 9. Camera ego-motion estimates using a mixed-domain sequence. Plots (a), (b), and (c) are the estimates of translation along the *X, Y*, and *Z* axes, respectively. Plots (d), (e), and (f) are the estimates of global rotation about the *X, Y*, and *Z* axes, respectively.

stant angular velocity of 0.1 rad/s. Fourteen feature points are tracked through 57 image frames. AWGNs with zero mean and 2-pixel STD are added to the synthesized image sequences. A random point cloud is generated, and the structure parameters are shown in Table 2. For each feature point, (u, v) is the position of the perspective projection of the feature point onto the image plane in the first image frame, and α is the true depth of the feature point. It is easy to check that the feature points are not coplanar or on a cone surface containing the center of projection. Hence they satisfy the assumption of the uniqueness theorem in Ref. 19. By using the recursive estimator proposed in Ref. 13 with good initial-state values and appropriate dynamic covariance matrix settings, we can obtain one solution near the ground truth. The estimates of the depth of the feature points $\{\alpha_t\}$ are also shown in Table 2. We denote this true solution as S_t. We can also obtain another solution for the same observation. The associated depth estimates are $\{\alpha_f\}$, listed in the last column of Table 2. We denote this false solution as S_f. We see that this set of solution is far from the ground truth. Now let us check the admissibility of these two solutions. The cost function for a weighted-least-squares estimate $\mathbf{x}$ is given by

$$\mathbf{C}(\mathbf{x}) = [\mathbf{h}_0 - \mathbf{h}(\mathbf{x})]^{\mathrm{T}}\mathbf{R}^{-1}[\mathbf{h}_0 - \mathbf{h}(\mathbf{x})], \tag{37}$$

where $\mathbf{h}_0$ is the observation, $\mathbf{h}(\mathbf{x})$ is the measurement equation, and $\mathbf{R}$ is the covariance matrix of the measurement noise. The resulting costs of S_t and S_f are 2553.9 and 2255.1, respectively. We can see that it is hard to distinguish between these two solutions. The false solution even gives a lower cost than the true one. This example illustrates that multiple admissible solutions exist for SfM estimation using noisy feature correspondence over image sequences. This example also sheds light on the reverse-depth phenomenon in Ref. 13, where depth reversal of structure and motion was occasionally experienced when the size of the FOV was small. The existence of another solution plays a key role when the recursive estimator proposed in Ref. 13 converges to reversed depth.

One way to remove the false solution S_f is to use the inertial rate data as a direct measurement of the inter-frame rotation in the Kalman filter.[13] We can obtain a solution similar to S_t, and the resulting cost function is 3096. The cost function of S_f increases dramatically to 51,763. Hence S_f is no longer admissible once the inertial data are included.

C. Mixed-Domain Sequences

Although the SfM problem has been studied for more than a quarter of a century, existing SfM algorithms work well only for the domains of image sequences for which they were designed, either implicitly or explicitly. For example, most algorithms work well when the camera translation is large and the 3D scene is extended. In our experiments, we also found that the A-P algorithm works well in this case. For small translation, algorithms fully exploiting this assumption[24] are claimed to be robust and reliable. However, to the best of our knowledge, no algorithm can handle sequences arbitrarily mixed by subsequences belonging to both domains. In Ref. 5, it is pointed out that the reason for this difficulty in designing a universal SfM algorithm is that "SfM estimate is such a complex function which depends on the image data differently in different domains that no one technique is likely to always approximate it well." Therefore it is suggested

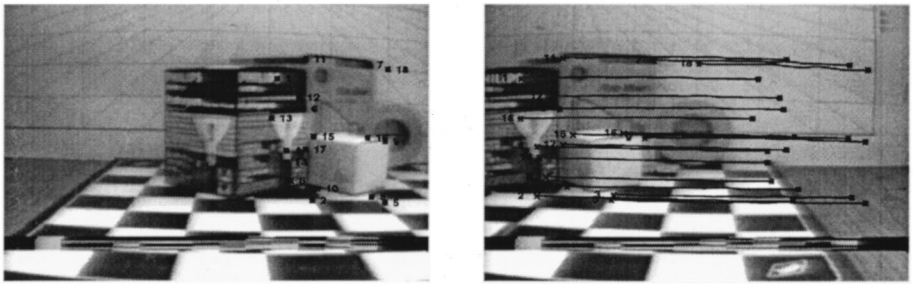


Fig. 10. Feature points and their trajectories tracked through a translation sequence collected in experiment 1. The bar code containing inertial rate data can be seen at the bottom of both frames.

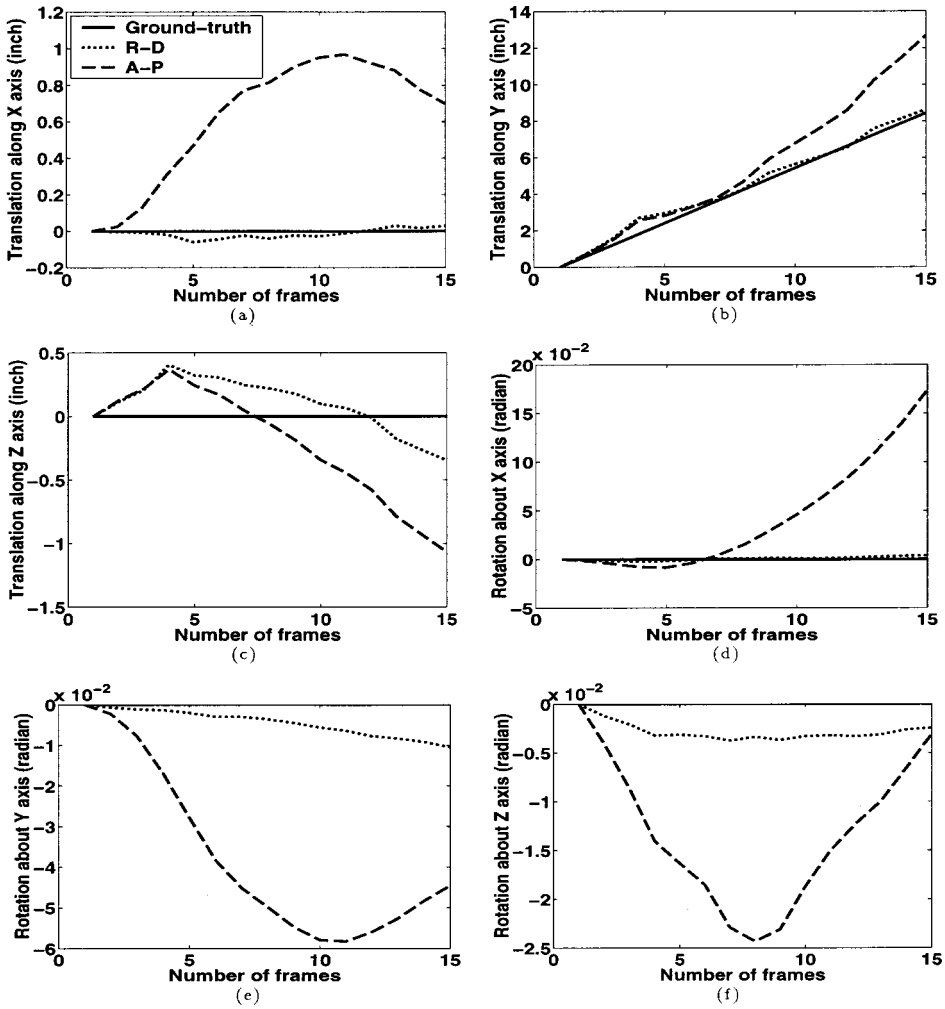
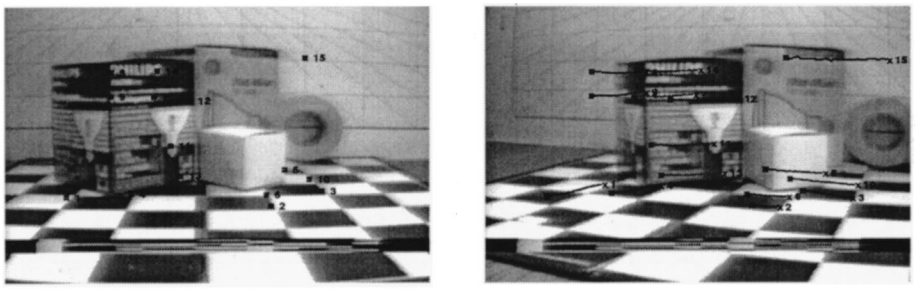


Fig. 11. Camera ego-motion estimates using a translational sequence collected in experiment 1. Plots (a), (b), and (c) are the estimates of translation along the *X, Y*, and *Z* axes, respectively. Plots (d), (e), and (f) are the estimates of global rotation about the *X, Y*, and *Z* axes, respectively.



(a) (b)

Fig. 12. Feature points and their trajectories tracked through a sequence collected in experiment 2.

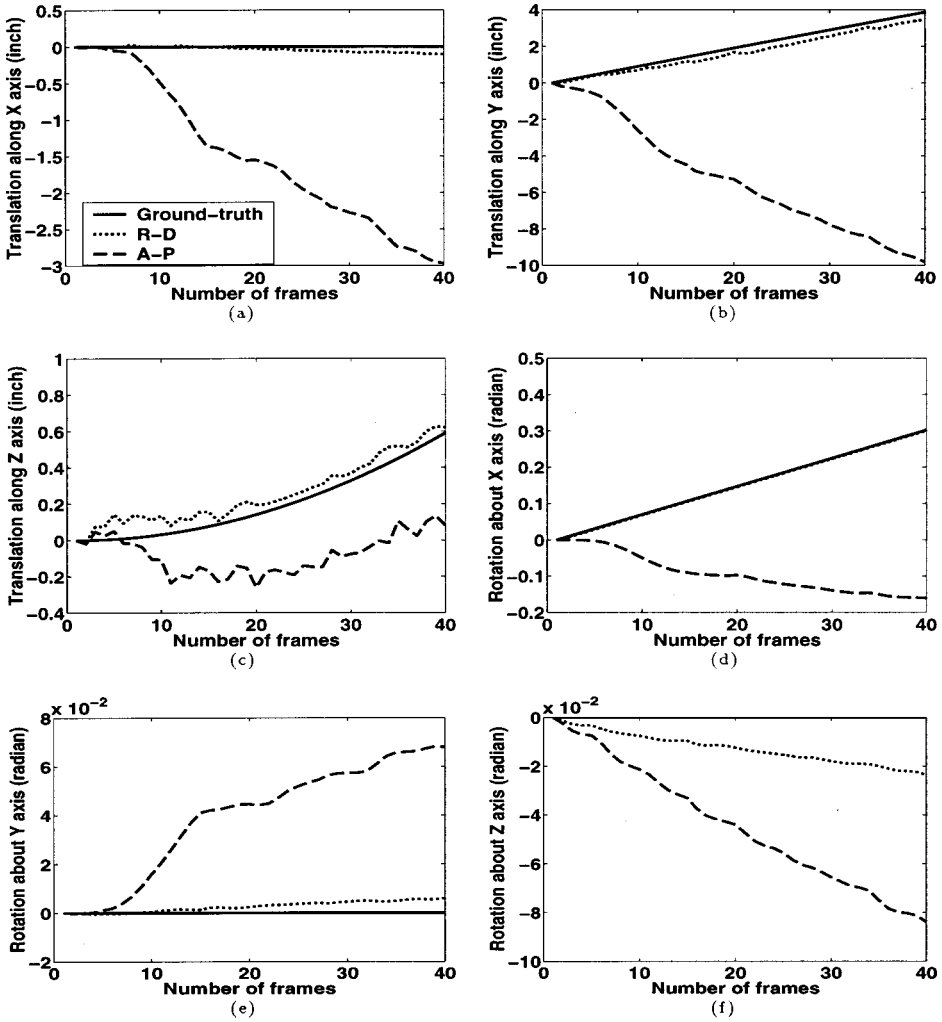


Fig. 13. Camera ego-motion estimates using a sequence collected in experiment 2. Plots (a), (b), and (c) are the estimates of translation along the *X, Y*, and *Z* axes, respectively. Plots (d), (e), and (f) are the estimates of global rotation about the *X, Y*, and *Z* axes, respectively.

in Ref. 5 that "one should design algorithms specifically for their problem domains and fully exploit the special characteristics of each domain to get robust, reliable and accurate algorithms." Sequences containing the extremes of different domains can be split into individual subsequences belonging to single domains, and then one can process those subsequences using corresponding SfM algorithms. It is not always an easy job to do this type of camera-ego-motion-based sequence segmentation and classification. Hence a challenging problem in SfM studies is how to handle mixed-domain sequences.

In our research, we found that the R-D algorithm can successfully work for sequences containing relatively smooth interframe motion and also large translation comparable with the depth of the feature points. This simulation is systematically designed such that both smooth and large translations are contained and also so that the large translation is larger than half of the nearest feature point. This makes this synthesized sequence a difficult example for the algorithms designed for small translation.[24] Hence this sequence is a mixed-domain sequence. Figure 8 shows the depth estimates obtained by

using both the R-D and A-P algorithms. It can be seen that since the inertial rate data are used in the R-D algorithm, the depth estimates of most feature points can still converge to the ground truth even when the translation is small (before the 11th frame). However, in contrast to the good performance of the R-D algorithm, the depth estimates of most feature points using the A-P algorithm converge to wrong values when the translation is small, and this furthermore causes the motion estimates to drift away from the ground truth when the translation is large. Figure 9 shows the motion estimation results obtained by using the R-D and A-P algorithms. It can be seen that the R-D algorithm is significantly better than the A-P algorithm, which can work well for large translation but fails when processing this mixed-domain sequence. This simulation shows that inertial rate data can be a method for handling mixed-domain sequences.

5. EXPERIMENTS USING REAL IMAGE SEQUENCES

Several video sequences containing inertial data were used to test our algorithm. The results show that the use of inertial rate data did make the motion estimator perform better than when only video cues were used. Two sets of results are included in this paper. In these sequences, the observed scenes were static. All the apparent motion was introduced by the camera ego-motion. In the first experiment, the sequence was captured by a camera moving on a straight train track with approximate constant velocity of 11.25 in./s. The parameters characterizing the video sequence are as follows. The camera captured 25 image frames per second with a FOV of $46° \times 34°$. The size of the output images is 320×240 pixels. Feature points are detected in the first frame and automatically tracked through the sequence. Figure 10(a) is the first frame of this sequence with labeled feature points, and Fig. 10(b) is the last frame of the sequence with feature trajectories. The feature points were detected and then tracked by using the Kanade–Lucas–Tomasi feature tracker.[25] In Fig. 10, the bar code containing inertial information can be seen at the bottom of the image frames. Both A-P and R-D algorithms were used to recover the motion and structure parameters from this sequence. Figure 11 shows the motion estimates obtained by using these two algorithms. It can be observed that the R-D algorithm greatly outperforms the A-P algorithm.

In the second experiment, the camera moved slowly along a curved track with a roughly constant angular velocity of 0.2 rad/s. The track was formed in an arc of a circle with a radius of 17.375 in. Figures 12(a) and 12(b) show, respectively, the detected feature points and the detected feature trajectories tracked through the image sequence. Figure 13 shows the motion estimation results using both the A-P and R-D algorithms, and it can be seen that in this experiment the R-D algorithm also performs better than the A-P algorithm.

6. CONCLUSIONS

In this paper, the effectiveness of inertial data for achieving robust structure-from-motion (SfM) estimation has been investigated. An extended-Kalman-filter-based (EKF-based) SfM algorithm has been developed to fully exploit the inertial information. We show that inertial data can be used in several subproblems of SfM such as inherent ambiguities reduction and handling of mixed-domain sequences. We have also shown that the number of feature points needed for accurate and robust SfM estimation can be significantly reduced once inertial data are used. Using both synthetic and real image sequences, we established the efficacy of using inertial information in the SfM problem. Cramér–Rao lower bounds have been used to compare the performance of different algorithms. Although, in this paper, we assume that the calibration of the camera is known, in our research, we did investigate the influence of the rate data on the recovery of camera focal length. However, no obvious improvement in camera calibration is observed by adding inertial rate data. In the applications where inertial rate data are available, camera intrinsic parameters are typically known, and therefore estimation of focal length may not be needed. Although the algorithm presented in this paper is based on the EKF, there is no doubt that the inertial data can be integrated in the SfM algorithms by using other recursive filters. It will be interesting to see how other types of inertial sensors such as accelerometers can help in solving the SfM problem of recovering the scaling factor in estimating the translation and structure parameters.

ACKNOWLEDGMENTS

This paper was prepared through collaborative participation in the Advanced Sensors Consortium sponsored by the U.S. Army Research Laboratory under Federated Laboratory Program Cooperative Agreement DAAL01-96-2-0001. We greatly thank Jim Ortolf for providing us data. The helpfulness of the anonymous reviewers is also acknowledged.

Address correspondence to Rama Chellappa, Room 2365, Center for Automation Research, A. V. Williams Building 115, University of Maryland, College Park, Maryland 20742-3275, or contact by phone, 301-405-3656; fax, 301-314-9115; or e-mail: chella@eng.umd.edu.

REFERENCES

1. R. Tsai and T. Huang, "Estimating 3-D motion parameters of a rigid planar patch. 1," IEEE Trans. Acoust. Speech Signal Process. **ASP-29**, 1147–1152 (1981).
2. T. J. Broida and R. Chellappa, "Estimation of object motion parameters from noisy image," IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-8**, 90–99 (1986).
3. T. J. Broida, S. Chandrashekhar, and R. Chellappa, "Recursive estimation of 3-D kinematics and structure from a noisy monocular image sequence," IEEE Trans. Aerosp. Electron. Syst. **26**, 639–656 (1990).
4. T. Jebara, A. Azarbayejani, and A. Pentland, "3-D structure from 2D motion," IEEE Signal Process. Mag. **16**(3), 66–84 (1999).
5. J. Oliensis, "A critique of structure from motion algorithms," Tech. Rep. (NEC Research Institute, Princeton, N.J., 2000), www.neci.nj.com/ homepages/oliensis/ poleiccv. ps.

6. C. Jerian and R. Jain, "Structure from motion: a critical analysis of methods," IEEE Trans. Syst. Man Cybern. **21**, 572–588 (1991).
7. P. Narayanan, P. Rander, and T. Kanade, "Constructing virtual worlds using dense stereo," in *Proceedings of the 6th International Conference on Computer Vision* (Narosa, New Delhi, 1998), pp. 3–10.
8. T. Viéville and O. Faugeras, "Computation of inertial information on a robot," in *Proceedings of the Fifth International Symposium on Robotics Research*, H. Miura and S. Arimoto, eds. (MIT, Cambridge, Mass., 1989), pp. 57–65.
9. T. Viéville, Romann, B. Hotz, H. Mathieu, M. Buffa, L. Robert, P. Facao, O. Faugeras, and J. Audren, "Autonomous navigation of a mobile robot using inertial and visual cues," in *Intelligent Robots and Systems*, M. Kikode, T. Sato, and K. Tatsuno, eds. (Publisher unknown, Yokohama, 1993).
10. T. Mukai and N. Ohnishi, "The recovery of object shape and camera motion using a sensing system with a video camera and a gyro sensor," in *Proceedings of the 7th International Conference on Computer Vision* (IEEE Computer Society, Los Alamitos, Calif., 1999), pp. 411–417.
11. S. Neill, "Synchronous data sampling system for image/data fusion," in *Proceedings of the 3rd Annual Fedlab Symposium* (U.S. Army Research Laboratory, Adelphi, Md., 1999), pp. 107–109.
12. J. M. Ortolf, P. Zemany, W. J. Kaiser, M. J. Dong, K. G. Yung, R. Howe, and A. Seshia, "Microsensors for army applications," in *Proceedings of the 2nd Annual Fedlab Symposium* (U.S. Army Research Laboratory, Adelphi, Md., 1998), pp. 93–98.
13. A. Azarbayejani and A. Pentland, "Recursive estimation of motion, structure, and focal length," IEEE Trans. Pattern Anal. Mach. Intell. **17**, 562–575 (1995).
14. A. H. Jazwinski, *Stochastic Processes and Filtering Theory* (Academic, New York, 1970).
15. H. Poor, *An Introduction to Signal Detection and Estimation* (Springer-Verlag, New York, 1988).
16. G. Young and R. Chellappa, "Statistical analysis of inherent ambiguities in recovering 3-D motion from a noisy flow field," IEEE Trans. Pattern Anal. Mach. Intell. **14**, 995–1013 (1992).
17. T. Broida and R. Chellappa, "Performance bounds for estimating three-dimensional motion parameters from a sequence of noisy images," J. Opt. Soc. Am. A **6**, 879–889 (1989).
18. Z. Zhang, "Determining the epipolar geometry and its uncertainty: a review," Tech. Rep. No. 2927 (French National Institute for Research in Computer Science and Control, Paris, 1996).
19. R. Tsai and T. Huang, "Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces," IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-6**, 13–27 (1984).
20. G. Adiv, "Inherent ambiguities in recovering 3-D motion and structure from a noisy flow field," IEEE Trans. Pattern Anal. Mach. Intell. **11**, 477–489 (1989).
21. K. Daniilidis and H. Nagel, "The coupling of rotation and translation in motion estimation of planar surfaces," in *Proceedings of the 8th IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (IEEE Computer Society, Los Alamitos, Calif., 1993), pp. 188–193.
22. D. Lawton, "Processing translational motion sequences," Comput. Vision Graph. Image Process. **22**, 116–144 (1983).
23. G. Adiv, "Determining 3-D motion and structure from optical flow generated by several moving objects," IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-7**, 384–401 (1985).
24. J. Oliensis, "A multi-frame structure-from-motion algorithm under perspective projection," Int. J. Comput. Vision **34**, 1–30 (1999).
25. C. Tomasi and J. Shi, "Good features to track," in *Proceedings of the 9th IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (IEEE Computer Society, Los Alamitos, Calif., 1994), pp. 593–600.