Structure from Motion Causally Integrated Over Time

Alessandro Chiuso, Paolo Favaro, *Student Member*, *IEEE*,
Hailin Jin, and Stefano Soatto, *Member*, *IEEE*

Abstract—We describe an algorithm for reconstructing three-dimensional structure and motion causally, in real time from monocular sequences of images. We prove that the algorithm is minimal and stable, in the sense that the estimation error remains bounded with probability one throughout a sequence of arbitrary length. We discuss a scheme for handling occlusions (point features appearing and disappearing) and drift in the scale factor. These issues are crucial for the algorithm to operate in real time on real scenes. We describe in detail the implementation of the algorithm, which runs on a personal computer and has been made available to the community. We report the performance of our implementation on a few representative long sequences of real and synthetic images. The algorithm, which has been tested extensively over the course of the past few years, exhibits honest performance when the scene contains at least 20-40 points with high contrast, when the relative motion is "slow" compared to the sampling frequency of the frame grabber ($30Hz$), and the lens aperture is "large enough" (typically more than 30^o of visual field).

Index Terms—Structure from motion, real-time vision, shape, geometry.

1 Introduction

Inferring the three-dimensional (3D) shape of a moving scene from its two-dimensional images is one of the classical problems of computer vision and is known as "structure from motion" (SFM). Among all possible ways in which this can be done, we distinguish between *causal schemes* and *noncausal ones*. More than the fact that causal schemes use—at any given point in time—only information from the past, the main difference between these two approaches lies in their goals and in the way in which data are collected. When the estimates of motion are to be used in real time, for instance, to accomplish a control task, a causal scheme must be employed since "future" data are not available for processing and the control action must be taken "now." In that case, the sequence of images is often collected sequentially in time, while motion changes smoothly under the auspices of inertia, gravity, and other physical constraints. When, on the other hand, we collect a number of "snapshots" of a scene from disparate viewpoints and we are interested in reconstructing it, there is no natural ordering or smoothness involved; using a causal scheme in this case would be highly unwise, and batch optimization based on bundle adjustment will naturally achieve better performance.

No matter how the data are collected, SFM is subject to fundamental tradeoffs, which are a severe obstacle to real-time real-world operation as we articulate in Section 1.2. This paper aims at addressing such tradeoffs: It is possible to integrate visual information over time, hence, achieving a global estimate of 3D motion, while maintaining the correspondence problem local. Among the obstacles we encounter is the fact that individual points tend to become occluded during motion, while novel points become visible. While we show how visual information can be integrated, we have to tone down our hopes of being able to do so *optimally*, for there exists no known finite-dimensional optimal solution to this problem. Therefore, we have to resort to *approximations*. It is our goal to provide algorithms that work in practice as well as in theory; our contributions in the matter of analysis can be summarized as follows: On the *observability* of 3D structure and motion, we provide a simpler proof of the (well known) global observability; we then prove uniform observability, which we use to characterize the *minimal realization* of the model. These results are crucial for proving the *stability* of the estimation algorithm that we propose (a *nonlinear filter*). Finally, we describe a complete real-time *implementation* of the algorithm, which includes an approach to causally handle *occlusions* and partial self-calibration.

1.1 A First Formalization of the Problem

Consider an N-tuple of points in the three-dimensional Euclidean space, represented as a matrix

$$\mathbf{X} \doteq \left[\mathbf{X}^1 \mathbf{X}^2 \cdots \mathbf{X}^N\right] \in \mathbb{R}^{3\times N}$$

- *A. Chiuso is with the Dipartimento di Elettronica ed Informatica, Via Gradenigo 6/a, 35131 Padova, Italy. E-mail: chiuso@dei.unipd.it.*
- *P. Favaro can be reached at Via Erizzo N. 65/A, Montebelluna(TV) 31030, Italy. E-mail: favaro@cs.ucla.edu or fava@ee.wustl.edu.*
- *H. Jin is with the Computer Science Department, University of California at Los Angeles, 3811C Boelter Hall, 405 Hilgard Ave., Los Angeles, CA 90095. E-mail: hljin@ee.wustl.edu.*
- *S. Soatto is with the University of California at Los Angeles, Boelter Hall 4531E, 405 Hilgard Ave., Los Angeles, CA 90095. E-mail: soatto@ucla.edu.*

Manuscript received 10 May 2000; revised 8 Mar. 2001; accepted 13 June 2001.
Recommended for acceptance by M. Irani.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 112068.

and let them move under the action of a rigid motion represented by a translation vector T and a rotation matrix R. Rotation matrices are orthogonal with unit determinant $\{R \mid RR^T = I \text{ and } \det(R) = 1\}$. Rigid motions transform the coordinates of each point via $R(t)\mathbf{X}^i + T(t)$. Associated to each motion T, R there is a velocity, represented by a vector of linear velocity V and a skew-symmetric matrix $\widehat{\omega}$ of rotational velocity.[1] Under such a velocity, motion evolves according to

$$T(t+1) = e^{\widehat{\omega}(t)}T(t) + V(t)\ ; \quad R(t+1) = e^{\widehat{\omega}(t)}R(t).$$

The exponential of a skew-symmetric matrix can be computed conveniently using Rodrigues' formula:

$$e^{\widehat{\omega}} = I + \frac{\widehat{\omega}}{\|\omega\|}\sin(\|\omega\|) + \frac{\widehat{\omega}^2}{\|\omega\|^2}\Big(1 - \cos(\|\omega\|)\Big)$$

for $\|\omega\| \neq 0$, otherwise $e^{\widehat{0}} = I$. We assume that, to an extent discussed in later sections, the *correspondence problem* is solved, that is we know which point corresponds to which in different projections (views). Equivalently, we assume that, we can measure the (noisy) projection

$$\mathbf{y}^i(t) = \pi\big(R(t)\mathbf{X}^i + T(t)\big) + \mathbf{n}^i(t) \in \mathbb{R}^2 \ \ \forall\ i = 1 \ldots N,$$

where we know the correspondence $\mathbf{y}^i \leftrightarrow \mathbf{X}^i$. We take as projection model an ideal pinhole, so that

$$\mathbf{y}^i = \pi(\mathbf{X}^i) = \left[\frac{\mathbf{X}^i_1}{\mathbf{X}^i_3}\frac{\mathbf{X}^i_2}{\mathbf{X}^i_3}\right]^T.$$

This choice is not crucial and the discussion can be easily extended to other projection models (e.g., spherical, orthographic, paraperspective, etc.). We do not distinguish between $\mathbf{y}^i$ and its projective coordinate (with a 1 appended), so that we can write $\mathbf{X}^i = \mathbf{y}^i X^i_3$. Finally, by organizing the time-evolution of the configuration of points and their motion, we end up with a discrete-time, nonlinear dynamical system:

$$\begin{cases} \mathbf{X}^i(t+1) = \mathbf{X}^i(t) & \mathbf{X}^i(0) = \mathbf{X}^i_0 \\ T(t+1) = e^{\widehat{\omega}(t)}T(t) + V(t) & T(0) = T_0 \\ R(t+1) = e^{\widehat{\omega}(t)}R(t) & R(0) = R_0 \\ V(t+1) = V(t) + \alpha_V(t) & V(0) = V_0 \\ \omega(t+1) = \omega(t) + \alpha_\omega(t) & \omega(0) = \omega_0 \\ \mathbf{y}^i(t) = \pi\big(R(t)\mathbf{X}^i(t) + T(t)\big) + \mathbf{n}^i(t) & \mathbf{n}^i(t) \sim \mathcal{N}(0, \Sigma_n), \end{cases} \tag{1}$$

where $\sim \mathcal{N}(M, S)$ indicates that a vector has a Gaussian distribution with mean M and covariance matrix S. In the above system, α is the relative acceleration between the viewer and the scene. If some prior modeling information is available (for instance, when the camera is mounted on a vehicle or on a robot arm), this is the place to use it. Otherwise, a statistical model can be employed. In particular, we can formalize our ignorance on acceleration by modeling α as a Brownian motion process.[2] In principle, one would like, at least for this simplified formalization of SFM, to find the optimal solution. Unfortunately, it can be shown that there exists no finite-dimensional optimal filter for this model.[3]

1.2 Trade-Offs in Structure from Motion

The first tradeoff involves the *magnitude of the baseline* and the *correspondence problem*: when images are taken from disparate viewpoints, estimating relative orientation is simple, given the correspondence. However, solving the correspondence problem is difficult, for it amounts to a global matching problem which spoils the possibility of use in real-time control systems. When images are collected closely in time, on the other hand, correspondence becomes an easy-to-solve local variational problem. However, estimating 3D motion becomes rather difficult since—on small motions—the noise in the image overwhelms the feeble information contained in the 2D motion of the features.[4]

No matter how one chooses to increase the baseline in order to bypass the tradeoff with correspondence, one inevitably runs into deeper problems, namely, the fact that individual feature points can *appear and disappear due to occlusions*, or to changes in their appearance due to specularities, changes in the light distribution, etc. To increase the baseline, it is necessary to associate the scale factor to an invariant of the scene. Therefore, in order to process that information, the scale factor must be included in the model. This tradeoff is fundamental and there is no easy way around it: information on shape can only be integrated as long as the shape is visible. We choose to address this problem by associating the scale factor to a "reference feature" chosen automatically among the visible ones. When that feature disappears, the reference switches to (the best current estimate of) another feature. Any error in the localization of that feature results in a global error, which increases every time the reference feature switches, effectively causing a slow drift in the estimates. Such a drift is unavoidable (no matter what the algorithm or the choice of reference for the scale factor), but can be compensated for "a posteriori," for instance, if a feature previously used as a reference becomes visible again, so that the trajectory can be recomputed. This global registration is a higher-level process that we do not address in this paper.

1. Skew-symmetric 3×3 matrices are represented using the "hat" notation $\hat{a} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$.

2. We wish to emphasize that this choice is not crucial towards the conclusions reached in this paper. Any other model would do, as long as the overall system is observable.

3. There are numerous reasons why the above formalization is altogether simplistic from the point of view of vision scientists, chief in the fact that the position of N points in space is hardly a satisfactory representation of the shape of a scene. Furthermore, we have assumed that the scene is a single rigid object (or that it has been *segmented* into rigid objects and we restrict the attention to one of them), and that we know the *correspondence* between points; assumptions that are all but unrealistic in any scene of practical interest. However, at least for this simple instantation of SFM, we would like to offer a provably stable, robust, and efficient algorithm.

4. There are many heuristics to bypass this tradeoff. For instance, one could track individual feature points from frame to frame in a sequence, but discard intermediate frames and start processing data only when the baseline is "large enough." A more principled way to proceed is to *increase the baseline by integrating visual information over time*. Notice that time-integration does not mean time-averaging: if the noise is such that the residual cost being minimized is flat, averaging is meaningless. A scheme for time integration of visual information must result in an effective increase of the baseline, while using information from each frame.

1.3 Relation to Previous Work and Organization of the Paper

We are interested in estimating motion so that we can use the estimates to accomplish spatial control tasks such as moving, tracking, manipulation, etc. In order to do so, the estimates must be provided *in real time and causally*, while we can rely on the fact that images are taken at adjacent instants in time and the relative motion between the scene and the viewer is somewhat smooth. Therefore, we do not compare our algorithm with batch multiframe approaches to SFM (such as those based upon multilinear geometry). This includes iterative minimization techniques such as "bundle adjustment." If one can afford the time to process sequences of images offline, of course a batch approach that optimizes simultaneously on all frames will perform better.[5]

Our work falls within the category of causal motion and structure estimation (also referred to as "recursive," or improperly, "Kalman filter-based" methods), that has a long history. To our knowledge, Dickmanns and Graefe [10], and Gennery [13], were the first to address the causal estimation of motion, confined to structured environments (objects with known shape in the case of Gennery, freeways in the case of Dickmanns and Graefe). The past 15 years have seen a proliferation of recursive schemes to estimate Euclidean structure from known motion [22], motion from known structure [5], [28], or both simultaneously [1], [3], [11], [12], [14], [15], [17], [19], [20], [23], [27], [29], [31], [32], [36], [37], [38], [39], [40], [41], and references therein. The first attempts to prove stability of the schemes proposed are recent [26]. However, few of the schemes cited address occlusions [8], which makes them prone to the tradeoffs just described and, therefore, hardly usable in realistic scenes where occlusions are the norm. The first attempts to handle occlusions in a causal scheme[6] came only a few years ago: McLauchlan et al. [23] proposed a filter with variable states, that however requires a batch initialization, while Soatto and Perona [35] proposed several schemes in which the problem of occlusions was bypassed by eliminating structure from the model. Cui et al. [8] have also proposed a method to handle occlusions, although that relies on a batch step to estimate interframe motion and, therefore, suffers the tradeoffs described in Section 1.2. Our approach is similar in spirit to the work of Azarbayejani and Pentland [3], extended to handle occlusions. In addition, the model in [3] is submininimal which results in an incorrect weighting of the measurements (see the Appendix and Fig. 2 for more details).

Part of this study is concerned with *analysis*. In the appendix, we analyze the conditions that are necessary in order to be able to causally reconstruct structure and motion. We prove uniform observability, which is crucial for the proof of stability of the algorithm that we propose. The other part, which represents the core of the paper, is concerned with the *implementation* of a system for functioning in real time on real scenes. We discuss our scheme for handling occlusions, drift in the scale factor and tuning of the filter. We then report some experiments with the scheme proposed. We have made our implementation available to the public [18], so that readers can test first-hand the performance and robustness of our scheme (or lack thereof).

5. One may argue that batch approaches are now fast enough to be used for real-time processing. However, speed is not the problem, robustness and delays are.

6. There are several ways of handling missing data in a batch approach: since they do not extend to causal processing, we do not review them here.

2 Realization

In order to design a finite-dimensional approximation to the optimal filter, we need an observable realization of the original model. Observability in SFM was addressed first in 1994, [9], [33] (see also, [34] for a more complete account of these results). The concept of observability in batch SFM (i.e., when there are no causality constraints) reduces to a uniqueness question that has been studied extensively in the literature of photogrammetry (see [24] and references therein).

2.1 Minimal Realization

In Appendix A, we prove the following:

Proposition 1. *The model:*

$$\begin{cases} \mathbf{y}_0^i(t+1) = \mathbf{y}_0^i(t) & i = 4 \dots N \quad \mathbf{y}_0^i(0) = \mathbf{y}_0^i \\ \rho^i(t+1) = \rho^i(t) & i = 2 \dots N \quad \rho^i(0) = \rho_0^i \\ T(t+1) = \exp(\hat{\omega}(t))T(t) + V(t) & \quad T(0) = T_0 \\ \Omega(t+1) = Log_{SO(3)}(\exp(\hat{\omega}(t)) \exp(\hat{\Omega}(t)) & \quad \Omega(0) = \Omega_0 \\ V(t+1) = V(t) + \alpha_V(t) & \quad V(0) = V_0 \\ \omega(t+1) = \omega(t) + \alpha_\omega(t) & \quad \omega(0) = \omega_0 \\ \mathbf{y}^i(t) = \pi\Big(\exp(\hat{\Omega}(t))\mathbf{y}_0^i(t)\rho^i(t) + T(t)\Big) + n^i(t) & i = 1 \dots N \end{cases} \tag{2}$$

is a minimal realization of (1). The notation $Log_{SO(3)}(R)$ stands for Ω such that $R = e^{\widehat{\Omega}}$ and is computed by inverting Rodrigues' formula. Ω is called the "canonical exponential representation" of R.

Remark 1. Notice that in the above model the index for $\mathbf{y}_0^i$ starts at 4, while the index for ρ^i starts at 2. This corresponds to choosing the first three points as reference for the similarity group and is sufficient to guarantee that the representation is minimal. As explained in Proposition 4, this can be done without loss of generality and is not affected by noise in the measurements nor by the mutual position of the points (as long as they are not collinear).

In the model (2), we are free to choose the initial conditions Ω_0, T_0, which we will therefore let be $\Omega_0 = T_0 = 0$, thereby choosing the camera reference at the initial time instant as the world reference. In Appendix B, we prove the following:

Proposition 2. *The linearized filter based upon the model (2) is stable with probability one.*

In order to avoid confusion, we shall denote with $\mathbf{y}_0(t|t), \rho(t|t), T(t|t), \widehat{\Omega}(t|t), V(t|t), \omega(t|t)$ the estimates up to time t, obtained by the filter based upon model (2), of $\mathbf{y}_0(t) = \mathbf{y}_0, \rho(t) = \rho_0, T(t), \widehat{\Omega}(t), V(t), \omega(t)$.

2.2 Extensions

As we have anticipated, the model proposed can be extended to account for changes in calibration. For instance, if we consider an imaging model with focal length[7] f,

$$\pi_f(\mathbf{X}) = \frac{f[X_1 X_2]^T}{X_3},$$

where the focal length can change in time, but no prior knowledge on how it does so is available, one can model its evolution as a random walk $f(t+1) = f(t) + \alpha_f(t)\alpha_f(t) \sim \mathcal{N}(0, \sigma_f^2)$ and insert it into the states of the model (1). As long as the overall system is observable, the conclusions reached in the appendix will hold. It is possible to prove that this is the case for the model just described. Another imaging model proposed in the literature is the following [3]:

$$\pi_\beta(\mathbf{X}) = \frac{[X_1 X_2]^T}{1 + \beta X_3}$$

for which similar conclusions can be drawn.

2.2.1 Saturation and Pseudomeasurements

As an alternative to rendering the model observable by eliminating states, it is possible to design a nonlinear filter directly on the (unobservable) model (1) by *saturating* the filter along the unobservable components of the state space. As we discuss in Appendix A, one can saturate the states corresponding to $\mathbf{y}_0^1, \mathbf{y}_0^2, \mathbf{y}_0^3$ and ρ^1. This guarantees that the filter initialized at $\mathbf{y}_0,\ \rho_0,\ V_0,\ \Omega_0,\ v_0,\ \omega_0$ evolves in such a way that

$$\mathbf{y}_0^1(t|t) = \mathbf{y}_0^1, \mathbf{y}_0^2(t|t) = \mathbf{y}_0^2, \mathbf{y}_0^3(t|t) = \mathbf{y}_0^3, \rho^1(t|t) = \rho_0^1.$$

In fact, let P_0 be the variance of the initial condition and Σ_w, the variance of the model error. It is simple to prove that if we set to 0 the columns and rows of P_0 and Σ_w corresponding to $\mathbf{y}_0^i, i = 1, 2, 3,$ and ρ^1, we are guaranteed that the gain corresponding to the update equations for $\mathbf{y}_0^i, i = 1, 2, 3$ and ρ^1 is zero.

Yet another alternative to render the model observable is to add pseudomeasurement equations

$$\rho^1 = \psi_1,\ \mathbf{y}_0^i(t) = \mathbf{y}^i(0)\ i = 1, 2, 3,$$

where ψ_1 is an arbitrary (positive) constant and $\mathbf{y}^i(0)$ are the measurements associated to the first three noncollinear points. Since measurements are noisy, in general, there will not exist a rigid motion that maps the noisy points to the true ones. As a consequence, there will be a bias in the estimation process. One way to prevent the filter from diverging is to set the covariance of the measurement noise associated with the pseudomeasurements to a small positive value. One should note that observability is not affected by the presence of measurement noise in the model.

2.3 Subminimal and Nonminimal Models

Most recursive schemes for causally reconstructing structure and motion available in the literature represent structure using only one state per point (either its depth in an inertial frame, or its inverse, or other variations on the theme). This corresponds to reducing the number of the states of the model (2), with the states $\mathbf{y}_0^i$ substituted for the measurements $\mathbf{y}^i(0)$, which causes the model noise $n(t)$ to be nonzero-mean.[8] When the zero-mean assumption, implicit in the use of the Kalman filter, is violated, the filter settles at a biased estimate. When adding new points, such biases add up to catastrophic consequences, as we show in Fig. 2. Note that this effect is not visible on short sequences, which is probably why it has not been noticed by Azarbayejani and Pentland [3]. In this case we say that the model is *subminimal*.

On the other hand, when the model is nonminimal—such is the case when we do not force it to evolve on the observable base of the state-space bundle—the variance of the estimation error along the components of the state parallel to the fibers explodes (see [7] and the appendix for more details on this issue, as well as Fig. 2).

3 Implementation: Occlusions and Drift in SFM

3.1 Occlusions: Point Features Appearing and Disappearing

When a feature point, say $\mathbf{X}^i$, becomes occluded, the corresponding measurement $\mathbf{y}^i(t)$ becomes unavailable. It is possible to model this phenomenon by setting the corresponding variance to infinity or, in practice, $\Sigma_{n^i} = MI_2$ for a suitably large scalar $M > 0$. By doing so, we guarantee that the corresponding states $\mathbf{y}_0^i(t|t)$ and $\rho^i(t|t)$ will not be updated.

An alternative, which is actually preferable in order to avoid useless computation and ill-conditioned matrix inversions, is to eliminate the states $\mathbf{y}_0^i$ and ρ^i altogether, thereby reducing the dimension of the state-space. This is simple due to the diagonal structure of the model (2): the states $\rho^i,\ \mathbf{y}_0^i$ are decoupled and, therefore, it is sufficient to remove them, and delete the corresponding rows from the gain matrix $L(t)$ and the model error variance $\Sigma_w(t)$ for all t past the disappearance of the feature (see Section 3.3).

When a new feature point appears, on the other hand, it is not possible to simply insert it into the state of the model, since the initial condition is unknown. Any initialization error will propagate onto the current estimate of the remaining states, through the update equation of the filter, and generate a spurious transient. We address this problem by running a separate filter in parallel for each new feature point using the current estimates of motion from the main filter in order to reconstruct the initial condition. Such a "subfilter" is based upon the following model, where we assume that N_τ features appear at time τ, for $i = 1, 2, \ldots, N_\tau$ and $t > \tau$:

$$\begin{cases} \mathbf{y}_\tau^i(t+1) = \mathbf{y}_\tau^i(t) + \eta_{y^i(t)} & \mathbf{y}_\tau^i(0) \sim \mathcal{N}(\mathbf{y}^i(\tau), \Sigma_{n^i}) \\ \rho_\tau^i(t+1) = \rho_\tau^i(t) + \eta_{\rho^i(t)} & \rho_\tau^i(0) \sim \mathcal{N}(1, P_\rho(0)) \\ \mathbf{y}^i(t) = \pi\Big(\exp(\hat{\Omega}(t|t))\big[\exp\big(\hat{\Omega}(\tau|\tau)\big)\big]^{-1}\big[\mathbf{y}_\tau^i(t)\rho_\tau^i(t) - T(\tau|\tau)\big] + T(t|t)\Big) + n^i(t). \end{cases} \quad (3)$$

7. This f is not to be confused with the generic state equation of the filter in Section 3.3.

8. In fact, if we call $n^i(0)$ the error in measuring the position of the ith point at time 0, we have that $\forall\ t\ E[n^i(t)] = n^i(0)$.

Note that the motion parameters do not appear in the state; as a consequence, all points are decoupled, which renders the state estimation very efficient. In practice, rather than initializing ρ to 1, one can compute a first approximation by triangulating on two adjacent views, and compute the covariance of the initialization error from the covariance of the current estimates of motion. Several heuristics can be employed in order to decide when the estimate of the initial condition is good enough for it to be inserted into the main filter. A natural criterion is when the variance of the estimation error of ρ^i_τ in the subfilter is comparable with the variance of ρ^j_0 for $j \neq 1$ in the main filter. The last step in order to insert the feature i into the main filter consists in bringing the coordinates of the new points back to the initial frame. This is done by

$$\mathbf{X}^i = \left[\exp(\hat{\Omega}(\tau|\tau))\right]^{-1}\left[\mathbf{y}^i_\tau \rho^i_\tau - T(\tau|\tau)\right]. \tag{4}$$

When inserting new features, their variance has to be initialized as well. This can be simply approximated from (4) just using Gauss' formula, i.e., linearizing (4) around the current estimate and computing the variance as if $\mathbf{X}^i$ were a linear function of $\hat{\Omega}(\tau|\tau)$, $\mathbf{y}^i_\tau$, ρ^i_τ, $T(\tau|\tau)$.

3.2 Drift

The only case when losing a feature constitutes a problem is when it is used to fix the observable component of the state-space (in our notation, $i = 1, 2, 3$) as explained in the appendix.[9] The most obvious choice consists in associating the reference to any other visible point. This can be done by saturating the corresponding states and assigning as reference value the current best estimate. In particular, if feature i is lost at time τ, and we want to switch the reference index to feature j, we eliminate $\mathbf{y}^i_0,\ \rho^i$ from the state, and the corresponding blocks from Σ_w and $P(\tau)$, and set rows and columns of Σ_w and $P(\tau)$ corresponding to $\mathbf{y}^j_0,\ \rho^j$ to zero. Therefore, following the discussion in Section 2.2, we have that

$$\mathbf{y}^j_0(\tau + t|\tau + t) = \mathbf{y}^j_0(\tau|\tau) \forall\ t > 0. \tag{5}$$

If $\mathbf{y}^j_0(\tau|\tau)$ was equal to $\mathbf{y}^j_0$, its "true" value, switching the reference feature would have no effect on the other states, and the filter would evolve on the same observable component of the state-space determined by the reference feature i. However, in general the difference $\tilde{\mathbf{y}}^j_0(\tau|\tau) \doteq \mathbf{y}^j_0 - \mathbf{y}^j_0(\tau|\tau)$ is a random variable with variance

$$\Sigma_\tau = P_{3j-3\ldots3j-1,3j-3\ldots3j-1}.$$

Therefore, switching the reference to feature j causes the observable component of the state-space to move by an amount proportional to $\tilde{\mathbf{y}}^j_0(\tau|\tau)$. When a number of switches have occurred, we can expect, on average, the state-space to move by an amount proportional to the product of $\|\Sigma_\tau\|$ and the number of switches. As we discussed in Section 1.2, this is unavoidable. What we can do is at most to try to keep the bias to a minimum by switching the reference to the state that has the lowest variance.[10]

Of course, should the original reference feature i become available, one can immediately switch the reference back to it, and, therefore, recover the original base and annihilate the bias.

3.3 Complete Algorithm

Let f and h denote the state and measurement model, so that (2) can be written in concise form as

$$\begin{cases} \xi(t+1) = f(\xi(t)) + w(t) & w(t) \sim \mathcal{N}(0, \Sigma_w) \\ y(t) = h(\xi(t)) + n(t) & n(t) \sim \mathcal{N}(0, \Sigma_n). \end{cases} \tag{6}$$

With respect to (2), we have added the model noise $w(t) \sim \mathcal{N}(0, \Sigma_w)$ which accounts for modeling errors.

Initialization. Choose the initial conditions

$$\mathbf{y}^I_0 = \mathbf{y}^i(0),\ \rho^i_0 = 1, T_0 = 0, \Omega_0 = 0, V_0 = 0, \omega_0 = 0,$$
$$\forall\ i = 1 \ldots N.$$

For the initial variance P_0, choose it to be block diagonal with blocks $\Sigma_{n^i}(0)$ corresponding to $\mathbf{y}^i_0$, a large positive number M (typically 1,000-10,000 units of focal length) corresponding to ρ^i, zeros corresponding to T and Ω (fixing the inertial frame to coincide with the initial reference frame). We also choose a large positive number W for the blocks corresponding to V and ω (typically 100-1,000 units of focal length).

The variance $\Sigma_n(t)$ is usually available from the analysis of the feature tracking algorithm. We assume that the tracking error is independent in each point and, therefore, Σ_n is block diagonal. We choose each block to be the covariance of the measurement $\mathbf{y}^i(t)$ (in the current implementation they are diagonal and equal to 0.5 pixel std.). The variance $\Sigma_w(t)$ is a design parameter that is available for tuning. We describe the procedure in Section 3.4.

Finally, set

$$\begin{cases} \hat{\xi}(0|0) \doteq [\mathbf{y}^{4T}_0, \ldots \mathbf{y}^{NT}_0,\ \rho^2_0, \ldots, \rho^N_0,\ T^T_0, \Omega^T_0,\ V^T_0,\ \omega^T_0]^T \\ P(0|0) = P_0, \end{cases} \tag{7}$$

where $\hat{\xi}(t|\tau)$ denotes the estimate of $\xi(t)$ given the measurements up to time τ.

Transient. During the first transient of the filter, we do not allow for new features to be acquired. Whenever a feature is lost, its state is removed from the model and its best current estimate is placed in a storage vector. If the feature was associated with the scale factor, we proceed as in Section 3.2. The transient can be tested as either a threshold on the

9. When the scale factor is not directly associated to one feature, but is associated to a function of a number of features (for instance, the depth of the centroid, or the average inverse depth), then losing any of these features causes a drift.

10. Just to give the reader an intuitive feeling of the numbers involved, we find that in practice the average lifetime of a feature is around 10-30 frames depending on illumination and reflectance properties of the scene and motion of the camera. The variance of the estimation error for $\mathbf{y}^i_0$ is in the order of 10^{-6} units of focal length, while the variance of ρ^i is in the order of 10^{-4} units for noise levels commonly encountered with commercial cameras.

innovation, a threshold on the variance of the estimates, or a fixed time interval. We choose a combination with the time set to 30 frames, corresponding to one second of video. During this first phase, one can also remove from the state the biases $\mathbf{y}_0^i$, since their measurement error is negligible compared to the model error, as an anonymous reviewer has suggested. These states must be reinserted during the regular operation of the filter to guarantee stability during the addition of new features to the state.

The camera may stop moving or undergo purely. rotational motion, without consequences during the regular operation of the filter. If that occurs before the filter has reached steady-state, convergence will not occur and the filter has to be reinitialized.

The recursion to update the state ξ and the variance P proceeds as follows (see (6)):

Prediction.

$$\begin{cases} \hat{\xi}(t+1|t) = f(\hat{\xi}(t|t)) \\ P(t+1|t) = F(t)P(t|t)F^T(t) + \Sigma_w. \end{cases} \tag{8}$$

Update.

$$\begin{cases} \hat{\xi}(t+1|t+1) = \hat{\xi}(t+1|t) + \\ L(t+1)\Big(y(t+1) - h(\hat{\xi}(t+1|t))\Big) \\ P(t+1|t+1) = \Gamma(t+1)P(t+1|t)\Gamma^T(t+1) + \\ L(t+1)\Sigma_n(t+1)L^T(t+1) \end{cases} \tag{9}$$

Gain.

$$\begin{cases} \Lambda(t+1) \doteq H(t+1)P(t+1|t)H^T(t+1) + \Sigma_n(t+1) \\ L(t+1) \doteq P(t+1|t)H^T(t+1)\Lambda^{-1}(t+1) \\ \Gamma(t+1) \doteq I - L(t+1)H(t+1) \end{cases} \tag{10}$$

Linearization.

$$\begin{cases} F(t) \doteq \frac{\partial f}{\partial \xi}(\hat{\xi}(t|t)) \\ H(t+1) \doteq \frac{\partial h}{\partial \xi}(\hat{\xi}(t+1|t)). \end{cases} \tag{11}$$

The detailed calculations of the linearization above are reported in [7].

Regime. Whenever a feature disappears, we simply remove it from the state as during the transient. However, after the transient a feature selection module works in parallel with the filter to select new features so as to maintain roughly a constant number (equal to the maximum that the hardware can handle in real time), and to maintain a distribution as uniform as possible across the image plane. We implement this by randomly sampling points on the plane, searching then around that point for a feature with enough brightness gradient (we use an SSD-type test [21]).

Once a new point-feature is found (one with enough contrast along two independent directions), a new filter (which we call a "subfilter") is initialized based on the model (3). We denote (see (3)) with $\rho_\tau^i(t|t)\mathbf{y}_\tau^i(t|t)$ the estimate (at time t), $i = 1, \ldots, M$ of the position of the ith new feature in the reference frame of the camera at time τ. The estimate is computed by means of an Extended Kalman Filter based on the model (3). Its evolution is given by

Initialization.

$$\begin{cases} \mathbf{y}_\tau^i(\tau|\tau) = \mathbf{y}^i(\tau) \\ \rho_\tau^i(\tau|\tau) = 1 \\ P_\tau^i(\tau|\tau) = \begin{bmatrix} \Sigma_{n^i}(\tau) & \mathbf{0} \\ \mathbf{0} & M \end{bmatrix}. \end{cases} \tag{12}$$

Prediction.

$$\begin{cases} \mathbf{y}_\tau^i(t+1|t) = \mathbf{y}_\tau^i(t|t) \\ \rho_\tau^i(t+1|t) = \rho_\tau^i(t|t) \qquad t > \tau \\ P_\tau^i(t+1|t) = P_\tau^i(t|t) + \Sigma_w(t). \end{cases} \tag{13}$$

Update.

$$\begin{bmatrix} \mathbf{y}_\tau^i(t+1|t+1) \\ \rho_\tau^i(t+1|t+1) \end{bmatrix} = \begin{bmatrix} \mathbf{y}_\tau^i(t+1|t) \\ \rho_\tau^i(t+1|t) \end{bmatrix} + L_\tau(t+1)\big(\mathbf{y}^i(t+1) - \mathbf{y}^i(t+1|t)\big),$$

where

$$\mathbf{y}^i(t+1|t) = \pi\Big(\exp(\hat{\Omega}(t+1|t+1))\big[\exp(\hat{\Omega}(\tau|\tau))\big]^{-1} \big[\mathbf{y}_\tau^i(t|t)\rho_\tau^i(t|t) - T\big(\tau|\tau\big)\big] + T(t+1|t+1)\Big),$$

and P_τ^i is updated according to a Riccati equation in all similar to (9).

After a probation period, whose length is chosen according to the same criterion adopted for the main filter, the feature is inserted into the state using the transformation (4). The initial variance is chosen to be the variance of the estimation error of the subfilter.

3.4 Tuning

The variance $\Sigma_w(t)$ is a design parameter. We choose it to be block diagonal: we assign the blocks corresponding to $T(t)$ and $\Omega(t)$ to 10^{-8}. We choose the remaining parameters using standard statistical tests, such as the cumulative periodogram [4]. The idea is that the parameters in Σ_w are changed until the innovation process $\epsilon(t) \doteq y(t) - h(\hat{\xi}(t))$ is as close as possible to being white. The periodogram is one of many ways to test the "whiteness" of a stochastic process. We choose the blocks corresponding to $\mathbf{y}_0^i$ equal to the variance of the measurements, and the elements corresponding to ρ^i all equal to σ_ρ. We then choose the blocks corresponding to V and ω to be diagonal with element σ_v, and then we change σ_v relative to σ_ρ depending on whether we want to allow for more or less regular motions. We then change both, relative to the variance of the measurement noise, depending on the level of desired smoothness in the estimates.

Our tuning procedure typically settles for values in the order of 10^{-2} to 10^{-3} units of focal length for σ_v and in the order of 10^{-6} to 10^{-8} units of focal length for σ_ρ.

4 Experiments

The complexity of SFM makes it difficult to demonstrate the performance of an algorithm by means of a few plots. This

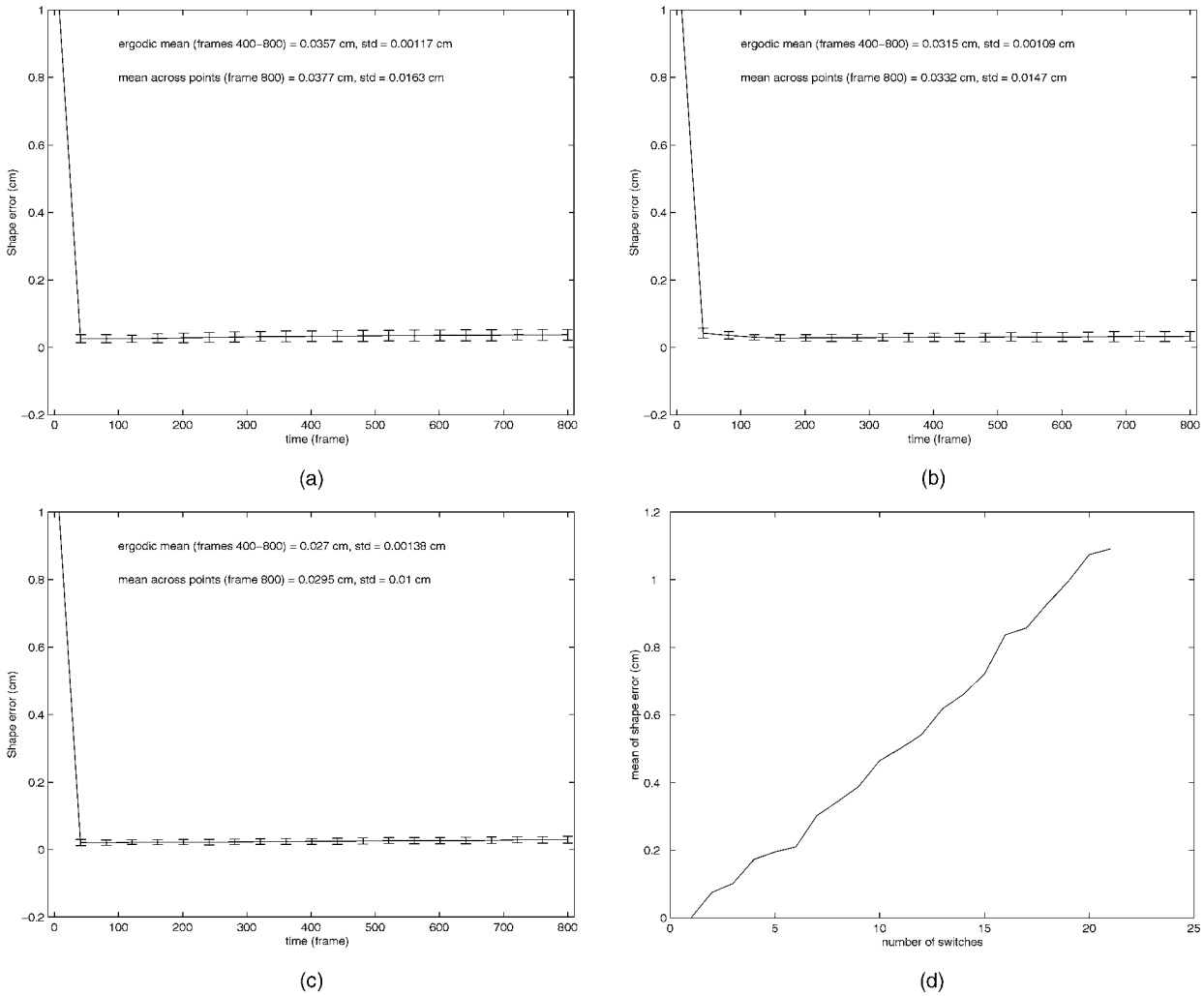

Fig. 1. **Structure error and drift.** Three different motions are tested on the same simulated scene with known ground truth. Ten trials of 800 frames each are performed. The error in mutual distance between the estimates and the ground truth of a set of 40 points is plotted. (a) Shows the structure error for forward translation (periodic translation along the z-axis). (b) Shows the shape error for sideway translation (periodic translation along the x-axis). (c) Shows the shape error for fixating motion (points rotating rigidly around an axis passing through their center of mass). Mean and standard deviation, both computed across the set of points at the last frame and across the last 400 frames, are all below one millimeter. The experiment is performed offline, and only unoccluded features are considered. **Scale drift.** (d) During a sequence of 200 frames, the reference feature is switched 20 times. The mean of the shape error drifts away, but at a slow pace, reaching about one centimeter by the end of the sequence.

is what motivated us to 1) obtain analytical results, which are presented in the appendix and 2) make our real-time implementation available to the public, so that the performance of the filter can be tested first-hand [18]. In this section, for the sake of exemplification, we present a set of representative experiments that illustrate the performance of the filter on real and synthetic datasets. In Fig. 2, we compare the performance of filters based on a minimal and subminimal model, as described in Section 2.3.

4.1 Structure Error

One of the byproducts of our algorithm is an estimate of the position of a number of point-features in the camera reference frame at the initial time. We use such estimates for a known object in order to characterize the performance of the filter. In particular, we randomly generate a cloud of points within a sphere of radius $0.25\ m$ centered about $1\ m$ away from the camera, and use a point with depth $1\ m$ to fix the scale factor. We run the filter on a sequence of 800 frames and plot the mean and standard deviation of the error between the estimated structure and the ground truth in Fig. 1. It can be seen that the error, despite an arbitrary initialization, remains well below $1\ mm$. In the case of forward translation, the filter occasionally displays significantly higher reconstruction errors, which we attribute to the presence of local minima observed and described by [6], [25]. The performance displayed is typical of several experimental trials we performed on both real and synthetic image sequences with ground truth. In Fig. 3 we show an equivalent plot for an experiment performed on real images with our real-time implementation of the algorithm on a similar scenario. We place an object on a turntable and measure the pose error at zero after the object has undergone one cycle of a periodic motion. The norm is computed in the same way as in the simulation experiment. Interestingly, the reconstruction error is smaller for the real

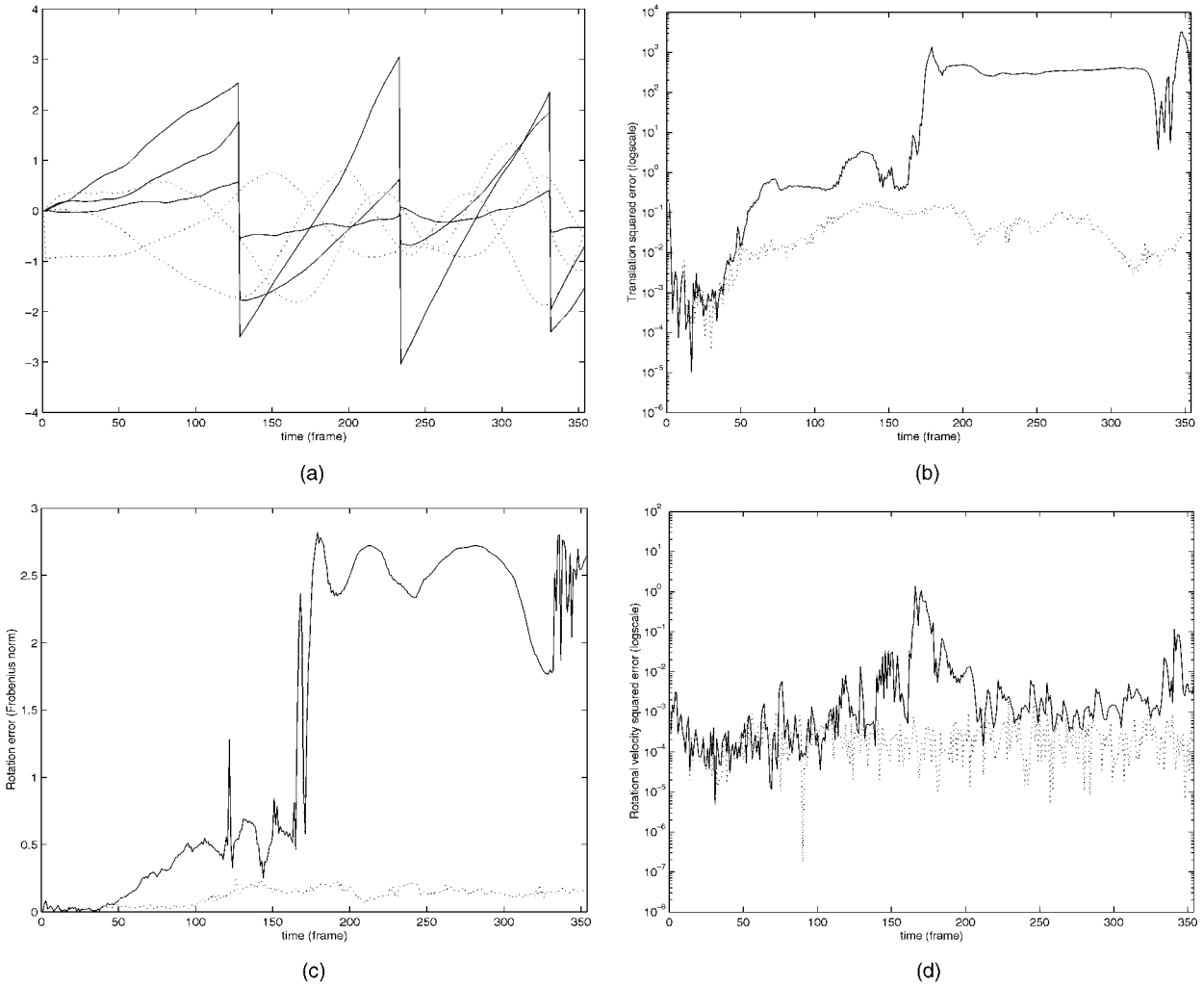

Fig. 2. **Comparison between minimal and the subminimal models.** A number of points move according to the motion in (a) (exponential coordinates of rotation are in solid lines, components of the translation vector are in dotted lines). New features are inserted approximately every 10 frames (insertion times have a Poisson distribution with intensity 10). The 2-norm of the translation error for the minimal filter (dotted line). New features are inserted approximately every 10 frames (insertion times have a Poisson distribution with intesity 10). The 2-norm of the translation error for the minimal filter (dotted line) and a subminimal implementation (solid line) is shown in the (b). Rotation error, measured by the Frobenius norm $\|I - \hat{R}^T R\|_F^2$ ($\hat{R}$ is the estimated rotation and R is the true one) is shown in (c), and velocity error is shown in the (d). As it can be seen, a subminimal model eventually saturates and results in very large errors (note that the scale is logarithmic), while the minimal model maintains a bounded error as predicted by the analysis in the appendix.

experiments. This is due to the high noise level used in our simulation experiments.

4.2 Motion Error

We have run experiments with three different periodic motions (forward translation, sideway translation and fixating motion) which are representative of the behavior of our algorithm. Exploiting the periodic nature of motion, it is possible to determine the accuracy of the motion estimates by measuring the distance between the estimated pose (rotation and translation) of the camera and its initial position. In particular, the translation error is the $\mathcal{L}^2$ norm of the difference between the estimated translation and the true one (zero), while rotation error is measured by the Frobenius norm of the discrepancy between the true rotation R and the estimated one $\hat{R}$: $\|I - \hat{R}R^T\|_F^2$. In particular, in this case, $R = I$. As it can be seen, the errors are comparable for the three types of motion. Note that although structure estimates for forward translation are prone to local minima, as we have discussed, such local minima are not visible in the motion estimates, as one would expect after the analysis in [6], [25]. In Fig. 3, we show that this distance is around $2\ cm$ for all the three motions.

Notice that in these experiments we have fixed the scale factor using a point in the scene with depth $1\ m$.

4.3 Scale Drift

In order to quantify the drift that occurs when the reference feature becomes occluded, we have generated a sequence of 200 frames and artificially switched the reference feature every 10 frames. The mean of the structure error is shown in Fig. 1. Despite being unavoidable, the drift is quite modest, around $1\ cm$ after 20 switches. When a subminimal model is used, the norm of the estimation error grows as a result of the drift, and the filter eventually becomes saturated, as it can be seen in Fig. 2.

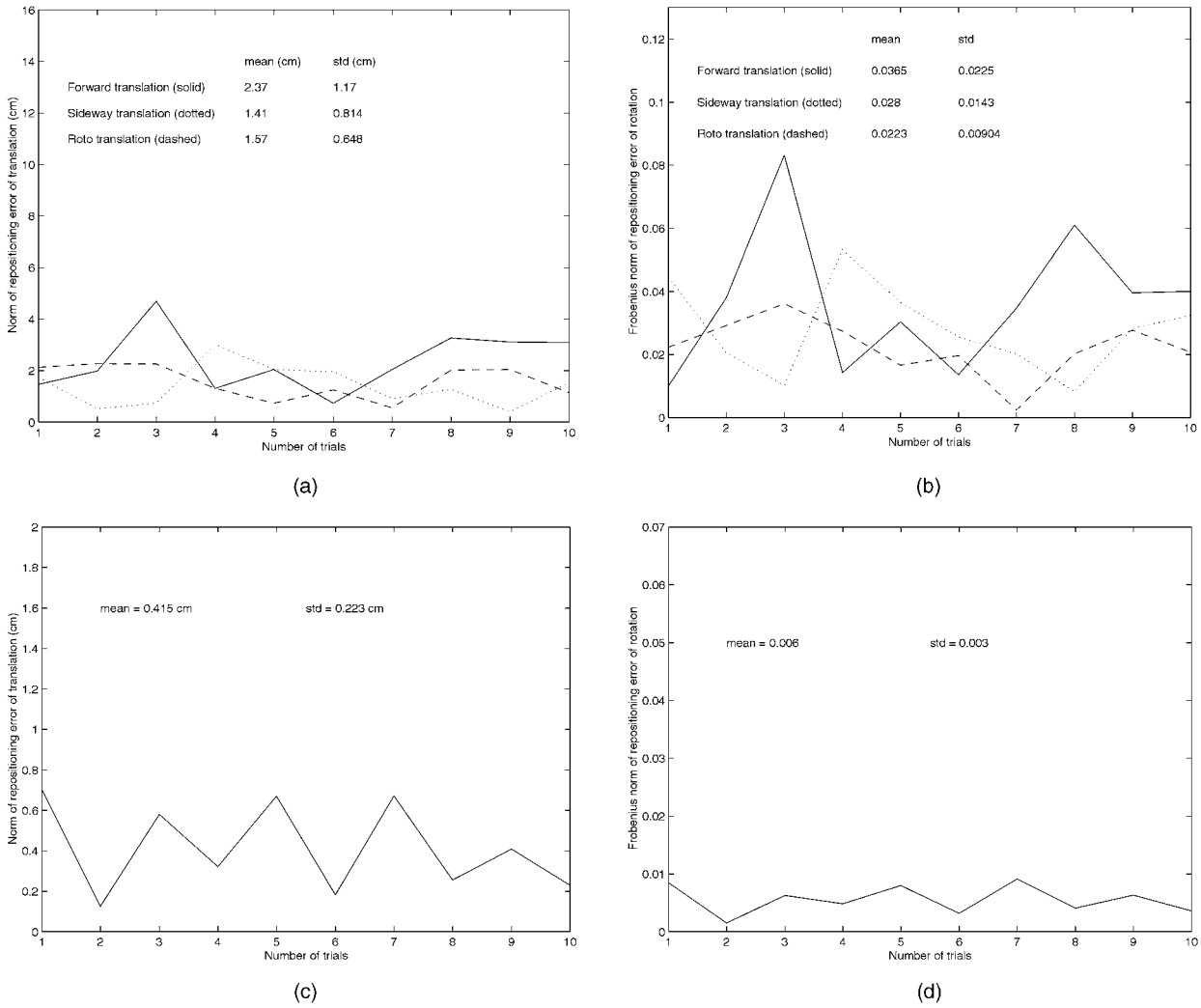

Fig. 3. **Motion error (top).** The three types of motion considered in Fig. 1 are periodic. The motion estimation error is thus defined as the repositioning error of the camera after a number of complete cycles indicated in abscissa. As it can be observed, mean and std for translation are around $2\ cm$ and $1\ cm$, respectively, (a) and around 0.03 and 0.02 rads, respectively, for rotation (b). **Real sequence (bottom).** The experimental conditions simulated in the experiments reported on the top plots have been recreated on a real scene. A box with dimension $15\ cm \times 20\ cm \times 30\ cm$ is rotated on a turntable. The camera is positioned about $1m$ away from its center. We rotate the box by 90 degrees, and record the sequence for 10 times. For each trial, we repeat the sequence backwards as if the box were rotated back to the initial position. The repositioning error of the camera is shown. As one can see, the error is comparable with that in the tests on the synthetic data and, indeed, it is consistently smaller. This is due to the high level of noise used in the simulation experiments. For the purpose of comparison, we manually preprocess the data and only point features that survive from the beginning to the end of the experiment are used (40 for each trial). Note that the scale factor is fixed at $1\ m$.

5 Conclusions

The causal estimation of three-dimensional structure and motion can be posed as a nonlinear filtering problem. In this paper, we have described the implementation of a real-time algorithm whose global observability, uniform observability, minimal realization and stability have been proven. The filter has been implemented on a personal computer, and the implementation has been made available to the public. The filter exhibits good performance when the scene contains at least 20-40 points with high contrast, when the relative motion is "slow" (compared to the sampling frequency of the frame grabber), when the scene occupies a significant portion of the image and the lens aperture is "large enough" (typically more than 30^o of visual field).

Appendix A

Observability

Let a rigid motion $g \in SE(3)$ be represented by a translation vector $T \in \mathbb{R}^3$ and a rotation matrix $R \in SO(3)$, and let $\alpha \neq 0$ be a scalar. The *similarity group*, which we indicate by $g_\alpha \in SE(3) \times \mathbb{R}\backslash\{0\}$ is the composition of a rigid motion and a scaling, which acts on points in $\mathbb{R}^3$ as follows:

$$g_\alpha(\mathbf{X}) = \alpha R\mathbf{X} + \alpha T.$$

We also define an action of g_α on $SE(3)$ as $g_\alpha(g') = \{\alpha RT' + \alpha T, RR'\}$ and an action on $se(3)$, represented by V and ω, as $g_\alpha(v) = \{\alpha V, \widehat{\omega}\}$. The similarity group, acting on an N-tuple of points in $\mathbb{R}^3$, generates an equivalence class: $[\mathbf{X}] = \{\mathbf{Y} \in \mathbb{R}^{3\times N} | \exists g_\alpha | \mathbf{Y} = g_\alpha \mathbf{X}\}$. Two configurations of points $\mathbf{X}$ and $\mathbf{Y} \in \mathbb{R}^{3\times N}$ are equivalent if there exists a

similarity transformation g_α that brings one onto the other: $\mathbf{Y} = g_\alpha \mathbf{X}$.

Consider a discrete-time nonlinear dynamic system of the form

$$\begin{cases} \xi(t+1) = f(\xi(t)) \quad \xi(t_0) = \xi_0 \\ y(t) = h(\xi(t)) \end{cases} \tag{15}$$

and let $y(t; t_0, \xi_0)$ indicate the output of the system at time t, starting from the initial condition ξ_0 at time t_0. In this section, we want to characterize the states ξ that can be reconstructed from the measurements y. Such a characterization depends upon the structure of the system $\{f,\ h\}$ but not on the measurement noise, which is therefore assumed to be absent for the purpose of the analysis in this section.

Definition 1. *Consider a system in the form (15) and a point in the state-space ξ_0. We say that ξ_0 is* indistinguishable *from ξ_0' if $y(t; t_0, \xi_0') = y(t; t_0, \xi_0) \forall\ t, t_0$. We indicate with $\mathcal{I}(\xi_0)$ the set of initial conditions that are indistinguishable from ξ_0.*

Definition 2. *We say that the system (15) is* observable up to a (group) transformation ψ *if*

$$\mathcal{I}(\xi_0) = [\xi_0] \doteq \{\xi_0' \mid \exists \psi | \xi_0' = \psi(\xi_0)\}.$$

Clearly, from measurements of the output $y(t)$ over any period of time, it is possible to recover at most the equivalence class where the initial condition belongs, which is $\mathcal{I}(\xi_0)$, but not ξ_0 itself. The only case when this is possible is that the system is observable up to the identity transformation. In this case, we have $\mathcal{I}(\xi_0) = \{\xi_0\}$ and we say that the system is observable.

For a generic linear time-varying system of the form

$$\begin{cases} \xi(t+1) = F(t)\xi(t) \quad \xi(t_0) = \xi_0 \\ y(t) = H(t)\xi(t) \end{cases} \tag{16}$$

the *k-observability Grammian* is defined as

$$M_k(t) \doteq \sum_{i=t}^{t+k} \Phi_i^T(t) H^T(t) H(t) \Phi_i(t), \forall k > 0,$$

where $\Phi_t(t) = I$ and $\Phi_i(t) \doteq F(i-1) \dots F(t)$ for $i > t$. The following definition will come handy in Appendix B.

Definition 3. *We say that the system (16) is* uniformly observable *if there exist real numbers $m_1 > 0,\ m_2 > 0$ and an integer $k > 0$ such that $\forall\ t\ m_1 I \leq M_k(t) \leq m_2 I$.*

Before stating our result on observability we need to impose some restrictions on the admissible motions. These restrictions essentially mean that the visible points are in front of the camera at a finite distance, the translation is not identically zero, and that we are not moving towards any point in the scene. This condition will be also necessary to have a well defined linearization.

Definition 4. *We say that a motion $\{V, U\}$ is admissible if $V(t)$ is not identically zero (i.e., there is an interval (a, b) such that $V(t) \neq 0,\ t \in (a, b)$), $U^T V$ is not along the direction of any point of the structure, and the corresponding trajectory of the system (2) is such that $c \leq \rho^i(t) \leq C, \forall\ i = 1, \dots, N, \forall\ t > 0$ for some constants $c > 0$, $C < \infty$.*

The following proposition revisits the fact that, when points are in general configuration,[11] stucture and motion are observable up to a (global) similarity transformation.

Proposition 3. *The model (1), where the points $\mathbf{X}$ are in general configuration, is observable up to a similarity transformation of $\mathbf{X}$ provided that motion is admissible (see Definition 4). In particular, the set of initial conditions that are indistinguishable from $\{\mathbf{X}_0, T_0, R_0, v_0\}$, where $e^{\widehat{v_0}} = \{V_0, U_0\}$, is given by*

$$\{\tilde{R}\mathbf{X}_0\alpha + \tilde{T}\alpha, T_0\alpha - R_0\tilde{R}^T\tilde{T}\alpha, R_0\tilde{R}^T, \tilde{v}_0\},$$

where $e^{\widehat{\tilde{v}_0}} = \{V_0\alpha, U_0\}$, $\tilde{R} \in SO(3)$, $\tilde{T} \in \mathbb{R}^3$ and $\alpha > 0$.

Proof. Suppose there exist two initial conditions $\{\mathbf{X}_1, T_1, R_1, v_1\}$ and $\{\mathbf{X}_2, T_2, R_2, v_2\}$ such that they generate the same measurements for all times t. In particular, at time $t = 0$ this is equivalent to the existence $\forall i = 1, \dots, N$, of scalings $A^i(0)$, such that

$$R_2\mathbf{X}_2^i + T_2 = (R_1\mathbf{X}_1^i + T_1)A^i(0).$$

Consider time $t = 1$, the indistinguishability condition can, therefore, be written as

$$e^{\widehat{v_2}}(R_2\mathbf{X}_2^i + T_2) = e^{\widehat{v_1}}(R_1\mathbf{X}_1^i + T_1)A^i(1).$$

Since all the points have to be visible, we have $\forall i = 1, \dots, N, A^i(0) > 0$ and $A^i(1) > 0$. Given $\mathbf{X}_1, T_1, R_1, v_1$, in order to find the initial conditions that are indistinguishable, we need to find $\forall i = 1, \dots, N, \mathbf{X}_2, T_2, R_2, v_2, A^i(0)$ and $A^i(1)$ such that, after some substitutions, we have

$$\forall i = e^{\widehat{v_2}}\big((R_1\mathbf{X}_1^i + T_1)A^i(0)\big) = \Big(e^{\widehat{v_1}}(R_1\mathbf{X}_1^i + T_1)\Big)A^i(1).$$

Making the representation of $SE(3)$ explicit, we write the previous conditions as:

$$U_2\tilde{X}^i A^i(0) + V_2 = U_1\tilde{X}^i A^i(1) + V_1 A^i(1) \quad i = 1, 2, \dots, N, \tag{17}$$

where $\tilde{X}^i \doteq R_1 X_1^i + T_1$, $e^{\widehat{v_1}} = \{V_1, U_1\}$ and $e^{\widehat{v_2}} = \{V_2, U_2\}$. Taking the inner product of both sides of (17) with $V_2 \times (U_1\tilde{\mathbf{X}}^i + V_1)$, we have the following identities:

$$(U_2\tilde{X}^i)^T\widehat{V}_2(U_1\tilde{X}^i + V_1) = 0 \qquad i = 1, 2, \dots, N, \tag{18}$$

where the hat notation is used for the cross product. Adding (18) to its transpose, we obtain:

$$(\tilde{X}^i)^T(U_2^T\widehat{V}_2U_1 - U_1^T\widehat{V}_2U_2)\tilde{X}^i + 2(\tilde{X}^i)^T U_2^T(V_2 \times V_1) = 0 \quad i = 1, 2, \dots, N. \tag{19}$$

This is a quadric equation in $\tilde{X}^i$ and, by the assumption of general configuration, it is not satisfied unless:

$$U_2^T\widehat{V}_2U_1 = U_1^T\widehat{V}_2U_2 \tag{20}$$

$$U_2^T(V_2 \times V_1) = 0. \tag{21}$$

The latter constraint implies that $V_2 = \alpha V_1$ for some $\alpha \neq 0$, since by assumption both V_1 and V_2 are nonzero.

11. We define that points are in *general configuration* when they do not lie onto any quadric surface containing the origin, or any of its degenerate cases (planes, lines, etc.), and they are in a number $N \geq 9$.

Multiplying both sides of (20) on the left by U_2 and on the right by U_2^T, we have

$$\widehat{V}_2 U_1 U_2^T = U_2 U_1^T \widehat{V}_2. \tag{22}$$

Multiplying both sides of (17) on the left by $U_1 U_2^T$ and recalling that $V_2 = \alpha V_1$, we have:

$$\begin{aligned} &\big(A^i(0)I - A^i(1) = U_1 U_2^T\big) U_1 \tilde{X}^i \\ &= \left(\frac{1}{\alpha} A^i(1) - 1\right) U_1 U_2^T V_2 \qquad i = 1, 2, \ldots, N. \end{aligned} \tag{23}$$

Multiplying both sides of (23) on the left by $\widehat{V}_2$ and employing the identity (22), we get:

$$\big(A^i(0)I - A^i(1) U_2 U_1^T\big) \widehat{V}_2 U_1 \tilde{X}^i = 0 \qquad i = 1, 2, \ldots, N. \tag{24}$$

Since points are in general configuration, there exist at least two points $\tilde{X}^1$ and $\tilde{X}^2$ such that the vectors $W_1 = \widehat{V}_2 U_1 \tilde{X}^1$ and $W_2 = \widehat{V}_2 U_1 \tilde{X}^2$ are nonzero and independent from each other. We have

$$U_2 U_1^T \big[A^1(1) W_1 \; A^2(1) W^2\big] = \big[A^1(0) W_1 \; A^2(0) W_2\big]. \tag{25}$$

This implies that $A^1(0) = A^1(1)$ and $A^2(0) = A^2(1)$ since $U_2 U_1^T$ is a rotation matrix and the scalings are strictly positive. Hence, $U_2 U_1^T = I$, i.e., $U_1 = U_2$. It follows immediately that $A^i(0) = A^i(1) = \alpha, i = 1, 2, \ldots, N$ from (17), under the condition that $\{V_1, U_1\}$ is admissible. Going back to $R_2 \mathbf{X}_2^i + T_2 = (R_1 \mathbf{X}_1^i + T_1) A^i(0)$, we conclude that $R_2 = R_1 \tilde{R}^T$, for some $\tilde{R} \in SO(3)$, $\mathbf{X}_2 = (\tilde{R}\mathbf{X}_1 + \tilde{T})\alpha$ for some $\tilde{T} \in \mathbb{R}^3$, and $T_2 = (T_1 - R_1 \tilde{R}^T \tilde{T})\alpha$. This concludes the proof. □

The following proposition states that it is possible to make the model observable by fixing the direction of three points and one depth in the structure. It is closely related to what some authors call invariance to "gauge transformation" [24]. Without loss of generality (i.e., modulo a reordering of the states), we will assume the indices of such three points to be 1, 2, and 3. We consider a point $\mathbf{X}$ as parameterized by its direction $\mathbf{y}$ and depth ρ, so that $\mathbf{X} = \mathbf{y}\rho$.

Proposition 4. *Given the direction of three noncollinear points, $\mathbf{y}^1, \mathbf{y}^2, \mathbf{y}^3$ and the scale of one point, $\rho^1 > 0$, and given vectors ϕ^i, $i = 1 \ldots N$, there exist at most 4 distinct motions $g = \{T, R\} \in SE(3)$ and corresponding scales $\alpha \in \mathbb{R}$ such that $\alpha R \mathbf{y}^i \rho^i + \alpha T = \phi^i$, $\forall i = 1 \ldots N \geq 3$.*

Proof. Suppose that the statement holds for $N = 3$, then it holds for any $N > 3$, as any additional equation of the form $\phi^i = \alpha R \mathbf{y}^i \rho^i + \alpha T$ is linear in the variable $\mathbf{X}^i \doteq \mathbf{y}^i \rho^i$ and, therefore, can be solved uniquely in $\mathbf{X}^i$ given α, R, and T. Since $\mathbf{X}_3^i = \rho^i$, the latter is uniquely determined and, so, is $\mathbf{y}^i = \frac{\mathbf{X}^i}{\rho^i}$. Therefore, we only need to prove the statement for $N = 3$:

$$\begin{cases} \phi^1 = \alpha R \mathbf{y}^1 \rho^1 + \alpha T \\ \phi^2 = \alpha R \mathbf{y}^2 \rho^2 + \alpha T \\ \phi^3 = \alpha R \mathbf{y}^3 \rho^3 + \alpha T. \end{cases} \tag{26}$$

Solving the first equation for αT and substituting it into the second and third equation, we get

$$\begin{cases} \phi^1 - \phi^2 = \alpha R(\mathbf{y}^1 \rho^1 - \mathbf{y}^2 \rho^2) \\ \phi^1 - \phi^3 = \alpha R(\mathbf{y}^1 \rho^1 - \mathbf{y}^3 \rho^3). \end{cases} \tag{27}$$

Noticing that $R^T R = I$, we get

$$(\phi^1 - \phi^2)^T (\phi^1 - \phi^2) = \alpha^2 (\mathbf{y}^1 \rho^1 - \mathbf{y}^2 \rho^2)^T (\mathbf{y}^1 \rho^1 - \mathbf{y}^2 \rho^2) \tag{28}$$

$$(\phi^1 - \phi^3)^T (\phi^1 - \phi^3) = \alpha^2 (\mathbf{y}^1 \rho^1 - \mathbf{y}^3 \rho^3)^T (\mathbf{y}^1 \rho^1 - \mathbf{y}^3 \rho^3) \tag{29}$$

$$(\phi^1 - \phi^2)^T (\phi^1 - \phi^3) = \alpha^2 (\mathbf{y}^1 \rho^1 - \mathbf{y}^2 \rho^2)^T (\mathbf{y}^1 \rho^1 - \mathbf{y}^3 \rho^3). \tag{30}$$

Solving ρ^3 and α^2 from (20) and (30), we have

$$\rho^3 = \frac{\frac{1}{\alpha^2} (\phi^1 - \phi^2)^T (\phi^1 - \phi^3) - \rho^{1^2} \mathbf{y}^{1^T} \mathbf{y}^1 + \rho^1 \rho^2 \mathbf{y}^{1^T} \mathbf{y}^2}{\rho^2 \mathbf{y}^{2^T} \mathbf{y}^3 - \rho^1 \mathbf{y}^{1^T} \mathbf{y}^3}$$

$$\frac{1}{\alpha^2} = \frac{(\mathbf{y}^1 \rho^1 - \mathbf{y}^2 \rho^2)^T (\mathbf{y}^1 \rho^1 - \mathbf{y}^2 \rho^2)}{(\phi^1 - \phi^2)^T (\phi^1 - \phi^2)}.$$

Plugging these two expressions into (29), we get a fourth-order equation for ρ^2 only which, in general, yields 4 possible solutions for ρ^2, each of which in turn gives a unique solution for ρ^3, but 2 for α. Therefore, we will have at most 8 sets of solutions for ρ^2, ρ^3, and α. Once ρ^2, ρ^3, and α are determined, R can be uniquely computed from (27) and then T can be computed from any of the (26). This concludes the proof. □

Proposition 4 suggests a way to render the model (1) locally observable by eliminating the states that fix the unobservable subspace.

Corollary 1. *The model (2), which is obtained by eliminating $\mathbf{y}_0^1, \mathbf{y}_0^2, \mathbf{y}_0^3$, and ρ^1 from the state of the model (1), is locally observable.*

Let

$$\xi \doteq [\mathbf{y}_0^{4^T}, \ldots, \mathbf{y}_0^{N^T}, \rho^2, \ldots, \rho^N, T^T, \Omega^T, V^T, \omega^T]^T$$

be the state vector of a minimal realization, and $F(t) \doteq \frac{\partial f(\xi)}{\partial \xi}$, $H(t) \doteq \frac{\partial \pi(\xi)}{\partial \xi}$ denote the linearization of the state and measurement equation in (2), respectively. Here, we just wish to remark that, in order for the linearization to be well-defined, we need to ensure that the depth of each point $\rho^i(t)$ is strictly positive as well as bounded. This is a reasonable assumption since it corresponds to each visible point being in front of the camera at a finite distance. Therefore, we restrict our attention to motions that guarantee that this condition is verified, i.e., satisfy Definition 4.

Proposition 5. *Let $F(t) \doteq \frac{\partial f(\xi)}{\partial \xi}$, $H(t) \doteq \frac{\partial \pi(\xi)}{\partial \xi}$ denote the linearization of the state and measurement equation in (2), respectively. Let $N \geq 5$, assume the motion is admissible and points are in general configuration, then the linearized system is uniformly observable.*

Proof. Let $k = 1$. To guarantee that $M_1(t)$ is bounded from below, let us first show that the matrix

$$O_1(t) \doteq [H^T(t) \; F^T(t) H^T(t+1)]^T$$

has full column rank for all values of t, since

$$M_1(t) = O_1(t)^T O_1(t).$$

The first $3N-7$ columns of $O_1(t)$ are

$$\begin{bmatrix} 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & \frac{\partial \pi(t)^2}{\partial \rho^2} & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \frac{\partial \pi(t)^3}{\partial \rho^3} & 0 & \dots & 0 \\ \frac{\partial \pi(t)^4}{\partial \mathbf{y}^4} & \dots & 0 & 0 & 0 & \frac{\partial \pi(t)^4}{\partial \rho^4} & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \frac{\partial \pi(t)^N}{\partial \mathbf{y}^N} & 0 & 0 & 0 & \dots & \frac{\partial \pi(t)^N}{\partial \rho^N} \\ 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & \frac{\partial \pi(t+1)^2}{\partial \rho^2} & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \frac{\partial \pi(t+1)^3}{\partial \rho^3} & 0 & \dots & 0 \\ \frac{\partial \pi(t+1)^4}{\partial \mathbf{y}^4} & \dots & 0 & 0 & 0 & \frac{\partial \pi(t+1)^4}{\partial \rho^4} & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \frac{\partial \pi(t+1)^N}{\partial \mathbf{y}^N} & 0 & 0 & 0 & \dots & \frac{\partial \pi(t+1)^N}{\partial \rho^N} \end{bmatrix}. \quad (31)$$

Under the condition that the motion is admissible and points are in general configuration, one can verify that the above matrix has full column rank. Similarly, it can be verified that the remaining 12 columns of $O_1(t)$ have rank 12 and are, in general, independent from the first $3N-7$ columns. Therefore, $O_1(t)$ has rank $3N+5$ and, thus, $M_1(t)$ is positive definite. This guarantees that in any finite interval $M_1(t)$ is strictly positive definite as long as the estimate of the state trajectory is admissible itself. The upper bound for $M_1(t)$ comes from the fact that all components of $O_1(t)$ are bounded from above when the motion is admissible, which concludes the proof. □

APPENDIX B

STABILITY

To streamline the notation, we call the estimation error $\tilde{\xi}(t) \doteq \xi(t) - \hat{\xi}(t)$, and $P(t)$ its variance at time t. The initial conditions for the estimator are

$$\begin{cases} \hat{\xi}(0) = \xi_0 \\ P(0) = P_0 > 0 \end{cases} \quad (32)$$

and its evolution is governed by

$$\begin{cases} \hat{\xi}(t+1) = f(\hat{\xi}(t)) + L(t+1)[y(t+1) - h(f\hat{\xi}(t))] \\ P(t+1) = \mathcal{R}(P(t), F(t), H(t), \Sigma_n, \Sigma_w), \end{cases} \quad (33)$$

where $\mathcal{R}$ denotes the usual Riccati equation which uses the linearization of the model $\{F, H\}$ computed at the current estimate of the state, as described in [16]. We call $\Sigma_{n_0},\ \Sigma_{w_0}$ the variance of the measurement and model noises, and $\Sigma_n,\ \Sigma_w$ the tuning parameters that appear in the Riccati equation.

The aim of this section is to prove that the estimation error generated by the filter just described is bounded. In order to do so, we need a few definitions.

Definition 5. *A stochastic process $\tilde{\xi}(t)$ is said to be* exponentially bounded in mean-square *(or MS-bounded) if there are real numbers $\eta,\ \nu > 0$ and $0 < \theta < 1$ such that*

$$|E\|\tilde{\xi}(t)\|^2 \leq \eta \|\tilde{\xi}(0)\|^2 \theta^t + \nu$$

for all $t \geq 0$. $\tilde{\xi}(t)$ is said to be bounded with probability one (or bounded WP1) if $P[\sup_{t\geq 0} \|\tilde{\xi}(t)\| < \infty] = 1$.

Definition 6. *The filter (6) is said to be* stable *if there exist real numbers $\epsilon, \delta > 0$ such that*

$$\|\tilde{\xi}(0)\| \leq \epsilon,\ \Sigma_n(t) \leq \delta I,\ \Sigma_w(t) \leq \delta I \implies \tilde{\xi}(t) \text{ is bounded.}$$

*Depending on whether $\xi(t)$ is bounded in mean square or with probability one, we say that the filter is "*MS-stable*" or "*stable WP1*".*

We are now ready to state the core proposition of this section.

Proposition 6. *If the hypothesis of Proposition 5 is fulfilled, then the filter based on the model (2) is MS-stable and stable WP1.*

To prove the proposition, we need the following lemma:

Lemma 1. *In the filter based on the model (2), let motion be admissible and $P_0 > 0$. Then there exist positive real numbers p_1 and p_2 such that $p_1 I \leq P(t) \leq p_2 I \forall\ t \geq 0$.*

Proof. The proof follows from Corollary 5.2 of [2], using Proposition 5 on the uniform observability of the linearization of (2). □

Proof of Proposition 6. The proposition follows directly from Theorem 3.1 in [30], making use in the assumptions of the boundedness of $F(t)$, $H(t)$, Lemma 1 and the differentiability of f and g when $0 < \rho^i < \infty \forall i$. □

ACKNOWLEDGMENTS

This research was supported by the US National Science Foundation (NSF) grant IIS-98766145, the US Army Research Office (ARO) grant DAAD19-99-1-0139, and Intel grant 8029.

REFERENCES

[1] G. Adiv, "Determining Three-Dimensional Motion and Structure from Optical Flow Generated by Several Moving Objects," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 7, no. 4, pp. 348-401, 1985.
[2] B. Anderson and J. Moore, *Optimal Filtering.* Prentice-Hall, 1979.
[3] A. Azarbayejani and A. Pentland, "Recursive Estimation of Motion, Structure and Focal Length," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 6, pp. 562-575, May 1995.
[4] M.S. Bartlett, *An Introduction to Stochastic Processes.* CUP, 1956.
[5] T. Broida and R. Chellappa, "Estimation of Object Motion Parameters from Noisy Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 8, no. 1, pp. 90-99, Jan. 1986.
[6] A. Chiuso, R. Brockett, and S. Soatto, "Optimal Structure from Motion: Local Ambiguities and Global Estimates," *Int'l J. Computer Vision,* vol. 39, no. 3, pp. 195-228, Sept. 2000.
[7] A. Chiuso and S. Soatto, "3D Motion and Structure Causally Integrated over Time: Analysis," Technical Report ESSRL, 99-03, Washington Univ., 1999.
[8] N. Cui, J. Weng, and P. Cohen, "Recursive-Batch Estimation of Motion and Structure from Monocular Image Sequences," *CVGIP: Image Understanding,* vol. 59, no. 2, pp. 154-170, 1994.
[9] W. Dayawansa, B. Ghosh, C. Martin, and X. Wang, "A Necessary and Sufficient Condition for the Perspective Observability Problem," *Systems and Control Letters,* vol. 25, no. 3, pp. 159-166, 1994.
[10] E.D. Dickmanns and V. Graefe, "Applications of Dynamic Monocular Machine Vision," *Machine Vision and Applications,* vol. 1, pp. 241-261, 1988.
[11] O. Faugeras, *Three Dimensional Vision, a Geometric Viewpoint.* MIT Press, 1993.
[12] C. Fermüller and Y. Aloimonos, "Tracking Facilitates 3-D Motion Estimation," *Biological Cybernetics,* vol. 67, pp. 259-268, 1992.
[13] D.B. Gennery, "Tracking Known 3-Dimensional Object," *Proc. AAAI Second Nat'l Conf. Artifical Intelligence,* pp. 13-17, 1982.

[14] J. Heel, "Dynamic Motion Vision," *Robotics and Autonomous Systems,* vol. 6, no. 1, 1990.
[15] X. Hu and N. Ahuja, "Motion and Structure Estimation Using Long Sequence Motion Models," *Image and Vision Computing,* vol. 11, no. 9, pp. 549-569, 1993.
[16] A.H. Jazwinski, *Stochastic Processes and Filtering Theory.* Academic Press, 1970.
[17] A. Jepson and D. Heeger, "Subspace Methods for Recovering Rigid Motion ii: Theory," RBCV TR-90-35, Univ. of Toronto—CS Dept., Nov. 1990, revised, July 1991.
[18] H. Jin, P. Favaro, and S. Soatto, "Real-Time 3-D Motion and Structure from Point Features: A Front-End System for Vision-Based Control and Interaction," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* code available from http://www.ee.wustl.edu/hljin/research, June 2000.
[19] J.J. Koenderink and A.J. Van Doorn, "Affine Structure from Motion," *J. Optical Soc. Am.* vol. 8, no. 2, pp. 377-385, 1991.
[20] R. Kumar, P. Anandan, and K. Hanna, "Shape Recovery from Multiple Views: A Parallax Based Approach," *Proc. Image Understanding Workshop,* 1994.
[21] B.D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proc. Seventh Int'l Joint Conf. Artificial Intelligence,* 1981.
[22] L. Matthies, R. Szelisky, and T. Kanade, "Kalman Filter-Based Algorithms for Estimating Depth from Image Sequences," *Int'l J. Computer Vision,* pp. 2989-2994, 1989.
[23] P. McLauchlan, I. Reid, and D. Murray, "Recursive Affine Structure and Motion from Image Sequences," *Proc. European Conf. Comp. Vision,* May 1994.
[24] P. McLauchlan, "Gauge Invariance in Projective 3D Reconstruction," *IEEE Workshop Multi-View Modeling and Analysis of Visual Scenes,* June 1999.
[25] J. Oliensis, "A New Structure-from-Motion Ambiguity," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 7, pp. 685-700, July 2000.
[26] J. Oliensis, "Provably Correct Algorithms for Multi-frame Structure from Motion," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 1996.
[27] J. Oliensis and J. Inigo-Thomas, "Recursive Multi-Frame Structure from Motion Incorporating Motion Error," *Proc. DARPA Image Understanding Workshop,* 1992.
[28] J. Philip, "Estimation of Three Dimensional Motion of Rigid Objects from Noisy Observations," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 13, no. 1, pp. 61-66, Jan. 1991.
[29] C. Poelman and T. Kanade, "A Paraperspective Factorization Method for Shape and Motion Recovery," *Proc. European Conf. Comp. Vision,* 1994.
[30] K. Reif, S. Gunther, E. Yaz, and R. Unbenhauen, "Stochastic Stability of the Discrete-Time Extended Kalman Filte," *IEEE Trans. Automatic Control,* vol. 44, no. 4, pp. 714-728, 1999.
[31] H.S. Sawhney, "Simplifying Motion and Structure Analysis Using Planar Parallax and Image Warping," *Proc. Int'l Conf. Pattern Recognition,* June 1994.
[32] L. Shapiro, A. Zisserman, and M. Brady, "Motion from Point Matches Using Affine Epipolar Geometry," *Proc. European Conf. Comp. Vision,* 1994.
[33] S. Soatto, "Observability/Identifiability of Rigid Motion under Perspective Projection," *Proc. 33rd IEEE Conf. Decision and Control,* pp. 3235-3240, Dec. 1994.
[34] S. Soatto, "3-D Structure from Visual Motion: Modeling, Representation and Observability," *Automatica,* vol. 33, pp. 1287-1312, 1997.
[35] S. Soatto and P. Perona, "Reducing Structure from Motion,: A General Framework for Dynamic Vision. Part 1: Modeling," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 9, pp. 993-942, Sept. 1998.
[36] M. Spetsakis and J. Aloimonos, "A Multi-Frame Approach to Visual Motion Perception," *Int'l J. Computer Vision,* vol. 6, no. 3, pp. 245-255, Aug. 1991.
[37] R. Szeliski, "Recovering 3D Shape and Motion from Image Streams Using Nonlinear Least Squares," *J. Visual Comm. and Image Representation,* 1994.
[38] M.A. Taalebinezhaad, "Direct Recovery of Motion and Shape in the General Case by Fixation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 14, no. 8, pp. 847-853, Aug. 1992.
[39] C. Tomasi and T. Kanade, "Shape and Motion from Image Streams Under Orthography: a Factorization Method," *Int'l J. Computer Vision,* vol. 9, no. 2, pp. 137-154, 1992.
[40] J. Weng, N. Ahuja, and T. Huang, "Optimal Motion and Structure Estimation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 15, no. 9, pp. 864-884, Sept. 1993.
[41] Z. Zhang and O.D. Faugeras, "Three Dimensional Motion Computation and Object Segmentation in a Long Sequence of Stereo Frames," *Int'l J. Computer Vision,* vol. 7, no. 3, pp. 211-241, 1992.

Alessandro Chiuso received the DIng. degree Cum Laude in 1996 and the PhD degree in systems engineering in 2000, both from the University of Padova. In 1998/99, he was a visiting research scholar with the Electronic Signal and Systems Research Laboaratory (ESSRL) at Washington University, St. Louis, Missouri. From March 2000 to July 2000, he has been visiting postdoctoral (EU-TMR) fellow with the Division of Optimization and System Theory, Department of Mathematics, KTH, Stockholm, Sweden. After that, he joined the Department of Electronics and Informatics, University of Padova, where since March 2001, he is research faculty ("ricercatore"). His research interests are mainly in computer vision, system theory, system identification, and estimation theory.

Paolo Favaro received the DIng degree from the University of Padua, Italy in 1999. He is currently a PhD student in electrical engineering, Washington University in Saint Louis, Missouri. His research interests are in computer vision. His research topics include nonlinear stochastic filtering, optical sensors and inverse filtering. He is a student member of the IEEE and the IEEE Computer Society.

Hailin Jin received the Bachelors' degree in engineering from Tsinghua University in 1998 and the MS degree in electrical engineering from Washington University in 2000. He is currently a Phd student in Department of Electrical Engineering, Washington University, St. Louis, Missouri. Since July 2000, he has been a visiting scholar in Computer Science Department, Univerisity of California at Los Angeles. His research interests are in computer vision. His research topics include geometry theory of computer vision, nonlinear stochastic filtering, and applications of level sets method.

Stefano Soatto received the DIng degree cum laude from the University of Padova in 1992, and the PhD degree in control and dynamical systems from the California Institute of Technology in 1996. In 1996/97, he was a postdoctoral fellow with the Division of Applied Sciences at Harvard University and then an assistant and associate professor of electrical engineering and biomedical engineering at Washington University, St. Louis, Missouri. From 1995 to 1998, he was with the research faculty in the Department of Mathematics and Computer Science at the University of Udine—Italy, since 2000 he has been an assistant professor with the Department of Computer Science at the University of California at Los Angeles. His research interests are in computer vision and in nonlinear systems and control theory. Dr. Soatto is the receipient of the David Marr Prize (with Y. Ma, J. Kosecka, and S. Sastry) in 1999, the Siemens Prize (with R. Brockett) at Computer Vision and Pattern Recognition in 1998, the US National Science Foundation Career Award in 1999, and the Okawa Fellowship in 2001. He is a member of the IEEE and the IEEE Computer Society.