The Visual Computer

A semi-direct approach to structure from motion

Hailin Jin[1], Paolo Favaro[1], Stefano Soatto[2]

[1] Department of Electrical Engineering, Washington University, Saint Louis, MO 63 130, USA
[2] Computer Science Department, UCLA, Los Angeles, CA 90 095, USA

Published online: 8 July 2003

The problem of structure from motion is often decomposed into two steps: feature correspondence and three-dimensional reconstruction. This separation often causes gross errors when establishing correspondence fails. Therefore, we advocate the necessity to integrate visual information not only in time (i.e. across different views), but also in space, by matching regions – rather than points – using explicit photometric deformation models. We present an algorithm that integrates image-feature tracking and three-dimensional motion estimation into a closed loop, while detecting and rejecting outlier regions that do not fit the model. Due to occlusions and the causal nature of our algorithm, a drift in the estimates accumulates over time. We describe a method to perform global registration of local estimates of motion and structure by matching the appearance of feature regions stored over long time periods. We use image intensities to construct a score function that takes into account changes in brightness and contrast. Our algorithm is recursive and suitable for real-time implementation.

Key words: Structure from motion – Direct methods – Extended Kalman filter – Observability – Tracking

The Visual Computer (2003) 19:377–394
Digital Object Identifier (DOI) 10.1007/s00371-003-0202-6

1 Introduction

Structure from motion (SFM) is concerned with estimating both the three-dimensional shape[1] of the scene and its motion relative to the camera, from a collection of images. The task is traditionally separated into two steps. First, point-to-point *correspondence* is established among different views of the same scene, using assumptions and constraints on its photometry. Then the correspondence is used to infer the *geometry* of the scene and its motion. This division is conceptually appealing because it confines the analysis of the images to the correspondence problem, after which SFM becomes a purely geometric problem. However, the photometric model imposed to establish correspondence typically relies on a constraint that is local in space and time, and therefore prone to gross errors. Global constraints, such as the fact that large portions of the scene move with a coherent rigid motion, or that the appearance changes due to the motion of the scene relative to the light, are not easy to embed into point-feature correspondence algorithms. Point correspondence is usually established by first selecting a large number of putative point features in each image, and then testing their compliance with a global projective model using standard robust statistical techniques. Even though efficient techniques are available that avoid brute-force combinatorial testing, one still has to first gather the images, then select point features, and finally test compliance with a global model. Since our interest is in using vision as a sensor for control, this approach is not viable because it introduces significant delays in the overall estimate. Delays can be catastrophic in a feedback setting since, during the delay, the system operates in open loop. In this paper we will describe *causal* estimation algorithms, which only use measurements gathered up to the current time to produce an estimate.

The alternative to separating the correspondence problem from the inference of shape and motion is to instead model the image irradiance explicitly and minimize a discrepancy measure between the measured images and the model. This is done in so-called 'direct methods', which we review in Sect. 1.1. Unfortunately, in general the deformation undergone by image irradiance as a consequence of rigid motion can only be described by an infinite-dimensional model, since it depends on the shape of the scene,

[1] In this paper we use the term 'shape' informally, as the three-dimensional structure of the scene described by the coordinates of a collection of points relative to *any* Euclidean reference frame.

which is unknown. At this point, one is faced with two alternatives. One is to enforce a model on the entire image, which will necessarily be highly complex and nonlinear. Another is to choose a finitely parameterizable class of image-deformation models, and segment the image into regions that satisfy the model (as verified in a statistical hypothesis test). Visual information will then be integrated locally in space (within a region), and globally in time (within a rigid object), while occluding boundaries and specular reflections are detected explicitly as violating the hypothesis. Of course, the size of the region will depend upon the maximum discrepancy from the model that we are willing to tolerate, and in general there will be a tradeoff between robustness (calling for larger regions) and accuracy (calling for smaller ones). In practice it is not necessary to cover the whole image with regions, since regions with small irradiance gradient do not impose shape constraints, and therefore significant speedups can be achieved.

In this context, our approach is half way between a dense method (that enforces a global model on the entire image) and a point-feature-based method (that enforces a separate model on each feature point). One can also view our effort as a step towards a dense representation of shape, moving from points to surfaces, with an explicit model of illumination. Indeed, we seek to integrate into a unified scheme photometry (feature tracking), dynamics (motion estimation), and geometry (point-wise reconstruction and surface interpolation). In particular, in our experimental assessment, we represent a piecewise-smooth surface with a collection of rigidly connected planes whose projections in the image undergo projective deformations. Spatial grouping allows a significant reduction of complexity, since points need not be detected and tracked individually.

1.1 Relation to previous work

The present work falls within the category of structure from motion, a field that encompasses a vast variety of research efforts, such as [1, 3, 5, 7, 8, 10, 12–14, 16, 17, 19–23, 26, 27, 29]. Of all the work in SFM, we consider in particular causal estimation algorithms. A batch approach would obviously perform better, but at the expense of compromising the usability for control actions such as manipulation, navigation or, more generally, real-time interaction.

Since we integrate tracking and motion estimation, our work also relates to the large literature on image motion. However, most tracking schemes rely on local deformation models and do not exploit feedback from higher levels. If the scene is a rigid collection of features that undergo the same rigid motion, this global constraint can be enforced by a feature tracker for robustness and precision. A small body of literature on direct methods addresses this issue, for example [9, 25]. The basic idea is to use the same brightness constancy constraint equation that is used to estimate optical flow or feature displacement as an implicit measurement of some SFM algorithm that estimates motion parameters. Image motion is then integrated globally, as long as the brightness constraint is satisfied. The exact constraint, however, depends upon the shape of the scene, which is unknown. Most work in direct methods represents shape as a collection of points whose projections are subject to brightness constancy and undergo the same rigid motion. Integrating motion information over the whole image, however, cannot be done since the brightness constancy assumption is not satisfied, notably at occluding boundaries.

Of all possible shape models, planes occupy a special place in that the projection of a plane undergoing rigid motion evolves according to a projective transformation. It is, therefore, natural to represent a scene as a collection of planes, which has been done often in the past, as for instance in [2, 24, 30].

Recently, Dellaert et al. [7] proposed a direct method for SFM that poses the problem as finding the maximum-likelihood estimate of structure and motion from all possible assignments of three-dimensional features to image measurements. In our work we avoid computing directly the correspondences. We use an explicit photometric model of the image deformation. The deformation results from the motion of the camera looking at piecewise-smooth surfaces. The model also allows reducing the accumulated drift over long time periods by registering image patches. Such a global registration has also been addressed recently by Rahimi et al. [18] in their study of differential trackers. However, our approach differs in two ways: first, we explicitly model the illumination changes which often occur over long time spans. Therefore, we can match features without being affected by bias commonly accumulated by differential methods. Second, the chosen surface representation allows us to efficiently search for the reappearance of previously selected features.

We seek to build on the strengths of direct methods, in order to avoid common problems with feature

tracking by embedding the process in higher-level motion estimation, while keeping computational complexity at bay by representing shape using a collection of simple templates.

2 From local photometry to global dynamics

Let S be a piecewise-smooth surface in three-dimensional space, and $\boldsymbol{X}$ be the coordinates of a generic point on it defined with respect to the reference frame attached to the camera[2]. We assume that the scene is static and the camera undergoes a motion $\{T(t), R(t)\}$, where[3] $R(t) \in SO(3)$ and $T(t) \in \mathbb{R}^3$ describe the rigid change of coordinates between the inertial frame (at time 0) and the moving frame (at time t). If we let $\boldsymbol{X}_0 \doteq \boldsymbol{X}(0)$, then we have $\boldsymbol{X}(t) = R(t)\boldsymbol{X}_0 + T(t)$. We assume to be able to measure, at each instant t, the image intensity $I(\boldsymbol{x}(t), t)$ at the point $\boldsymbol{x}(t) = \pi(\boldsymbol{X}(t))$, where π denotes the camera projection. For instance, in the case of perspective projection, $\pi(\boldsymbol{X}) = [\frac{X}{Z}, \frac{Y}{Z}]^T$, where $\boldsymbol{X} = [X, Y, Z]^T$. We also assume to work with calibrated cameras, i.e. cameras whose intrinsic parameters (such as focal length, principal points) have been calibrated. For ease of notation we will not make a distinction between image coordinates and homogeneous coordinates (with 1 appended). For Lambertian surfaces[4] with constant lighting conditions, as a consequence of camera motion, the image deforms according to a nonlinear time-varying function of the surface S, $g_t^S(\cdot)$ as follows:

$$I(\boldsymbol{x}_0, 0) = I(g_t^S(\boldsymbol{x}_0), t), \tag{1}$$

where $\boldsymbol{x}_0 \doteq \boldsymbol{x}(0) = \pi(\boldsymbol{X}_0)$. In general g_t^S depends on an infinite number of parameters (a representation of the surface S):

$$g_t^S(\boldsymbol{x}_0) = \pi(R(t)\boldsymbol{x}_0\rho(\boldsymbol{x}_0) + T(t)) \tag{2}$$

with $\rho(\boldsymbol{x}_0)$ subject to $\boldsymbol{x}_0\rho(\boldsymbol{x}_0) = \boldsymbol{X}_0 \in S$.

[2] The camera reference frame is chosen such that the origin coincides with the optical center of the camera, the x axis is parallel to the horizontal image axis and goes from left to right, the y axis is parallel to the vertical image axis and goes from top to bottom, and the z axis is parallel to the optical axis and points towards the scene.

[3] $SO(3)$ stands for the space of three-dimensional rotation matrices: $SO(3) = \{M \in \mathbb{R}^{3\times 3} \mid M^T M = I \text{ and } \det(M) = 1\}$.

[4] A surface is called *Lambertian* if it appears equally bright from all viewing directions.

However, one can restrict the class of functions g_t^S to depend upon a finite number of parameters (corresponding to a finite-dimensional parameterization of S), and therefore represent image deformations as a parametric class. Similarly, since in real scenes the lighting condition is subject to changes during motion, we will also consider a parameterized model for the photometric changes.

2.1 A generative model

There is a very simple instance when image deformations are captured by a finite-dimensional deformation model. That is when we restrict the class of surfaces to planes with unknown normal vector $\nu \in \mathbb{R}^3$. In fact, it is well known that a plane not passing through the origin (the optical center) can be described as $\Pi_0 = \{\boldsymbol{X}_0 \in \mathbb{R}^3 \mid \nu^T \boldsymbol{X}_0 = 1\}$, and therefore

$$\begin{aligned} g_t^{\Pi_0}(\boldsymbol{x}_0) &= \pi\left\{R(t)\boldsymbol{X}_0 + T(t)\right\} \\ &= \pi\left\{(R(t) + T(t)\nu^T)\boldsymbol{X}_0\right\} \\ &= \pi\left\{(R(t) + T(t)\nu^T)\boldsymbol{x}_0\right\} \qquad (3) \\ &= \frac{M_{1,2}\boldsymbol{x}_0}{M_3\boldsymbol{x}_0}, \qquad (4) \end{aligned}$$

where $M = R(t) + T(t)\nu^T$ and $M_{1,2}$ and M_3 denote the matrices made of the first two rows of M and the last row of M respectively. This transformation Eq. (4) for $\boldsymbol{x}_0$ is a planar projective transformation (also known as a *homography*).

In the appendix, we show in Lemma 1 that any two matrices $M^1(t)$ and $M^2(t)$, both with rank at least two, are in one-to-one correspondence with matrices of the form $R(t) + T(t){\nu^1}^T$ and $R(t) + T(t){\nu^2}^T$ (if we require that the scene has to be in front of the camera). Hence, if a scene contains at least two planar surfaces with sufficiently *exciting* texture (a precise definition will be given in Sect. 3.1), we can infer $T(t)$, $R(t)$, ν^1, and ν^2 by finding the matrices $M^1(t)$ and $M^2(t)$ that minimize some discrepancy measure between $I(\boldsymbol{x}_0^i, 0)$ and $I(M^i(t)\boldsymbol{x}_0^i, t)$, with $\boldsymbol{x}_0^i$ ranging in the image domain D^i, $i = 1, 2$:

$$\begin{aligned} \hat{M}^1(t) &= \arg\min_{M^1(t)} \sum_{\boldsymbol{x}_0 \in D^1} \|I(\boldsymbol{x}_0, 0) - I(M^1(t)\boldsymbol{x}_0, t)\|, \\ \hat{M}^2(t) &= \arg\min_{M^2(t)} \sum_{\boldsymbol{x}_0 \in D^2} \|I(\boldsymbol{x}_0, 0) - I(M^2(t)\boldsymbol{x}_0, t)\|, \end{aligned} \tag{5}$$

for some choice of norm $\|\cdot\|$. D^i is chosen to be inside the projection of the ith planar surface.

In practice, scenes are not always made of Lambertian surfaces, and the lighting condition may change over time. Hence, when modeling the observed images, it is necessary to take into account photometric variations. We observe that an affine model can locally approximate the changes in image intensity between the initial patch $I(\boldsymbol{x}_0, 0)$ and the current patch $I(\pi((R(t)+T(t)\nu^T)\boldsymbol{x}_0), t)$:

$$I(\boldsymbol{x}_0, 0) = \lambda I(\pi((R(t)+T(t)\nu^T)\boldsymbol{x}_0), t) + \delta \quad \forall \boldsymbol{x}_0 \in D, \tag{6}$$

where $\lambda \in \mathbb{R}$ and $\delta \in \mathbb{R}$. This has been shown to be a good compromise between modeling error and computational speed [11]. We can, therefore, extend Eq. (5) to estimate simultaneously the illumination parameters λ and δ together with $M^1(t)$ and $M^2(t)$:

$$\begin{aligned}
&\hat{\lambda}^1, \hat{\delta}^1, \hat{M}^1(t) = \\
&\arg\min_{\lambda^1, \delta^1, M^1(t)} \sum_{\boldsymbol{x}_0 \in D^1} \|I(\boldsymbol{x}_0, 0) - (\lambda^1 I(M^1(t)\boldsymbol{x}_0, t) + \delta^1)\|, \\
&\hat{\lambda}^2, \hat{\delta}^2, \hat{M}^2(t) = \\
&\arg\min_{\lambda^2, \delta^2, M^2(t)} \sum_{\boldsymbol{x}_0 \in D^2} \|I(\boldsymbol{x}_0, 0) - (\lambda^2 I(M^2(t)\boldsymbol{x}_0, t) + \delta^2)\|.
\end{aligned} \tag{7}$$

Notice that the residual to be minimized is computed in the space of image intensities, i.e. the real measurements. We can use the current model $\hat{\lambda}^i$, $\hat{\delta}^i$, and $\hat{M}^i(t)$ and the first image $I(\boldsymbol{x}_0, 0)$, $\boldsymbol{x}_0 \in D^i$ to predict the future image $I(\boldsymbol{x}(t+1), t+1)$. In this sense this model is *generative*.

If the scene is made of K planar patches with normals $\nu^1, \nu^2, \ldots, \nu^K$, all undergoing the same rigid motion $T(t)$, $R(t)$, instead of computing λ^i, δ^i, $M^i(t)$ $i = 1, 2, \ldots, K$ and then inferring $R(t)$, $T(t)$, we can model all the unknowns in a dynamical system. Photometric information is integrated within each patch, while geometric and dynamic information is integrated across patches. In this sense, this model describes the scene using *local photometry* and *global dynamics*.

Because $T(t)$ and ν^i appear as a product in Eq. (3), there is a scale-factor ambiguity between them. To remove this ambiguity, it is sufficient (the meaning of sufficiency will be made clear in Sect. 3.1) to fix a scalar among the coordinates of $T(t)$ or ν^i, $i = 1, 2, \ldots, K$. Since it is not convenient to fix any scalar quantity associated with $T(t)$, which is time-varying, we seek to fix a quantity associated with one of the normals ν^i, $i = 1, 2, \ldots, K$. Recall that our inertial reference frame is chosen such that the origin coincides with the camera center at time 0, and the z axis is parallel to the optical axis and points towards the scene. For any plane in the scene to be in front of the camera and visible, the z coordinate of its normal has to be positive. Therefore, we choose to fix the z coordinate of ν^1 to be some positive value, and use 1 for convenience. A dynamical model of the time evolution of all the unknown quantities is therefore

$$\begin{cases}
\lambda^i(t+1) = \lambda^i(t) + \alpha_\lambda(t) & i = 1, 2, \ldots, K \\
\delta^i(t+1) = \delta^i(t) + \alpha_\delta(t) & i = 1, 2, \ldots, K \\
\nu^{1,j}(t+1) = \nu^{1,j}(t) & j = 1, 2 \\
\nu^i(t+1) = \nu^i(t) & i = 2, 3, \ldots, K \\
T(t+1) = \exp(\widehat{\omega}(t))T(t) + V(t) & \\
R(t+1) = \exp(\widehat{\omega}(t))R(t) & \\
V(t+1) = V(t) + \alpha_V(t) & \\
\omega(t+1) = \omega(t) + \alpha_\omega(t) & \\
I(\boldsymbol{x}_0^i, 0) = \lambda^i(t) I(\pi((R(t)+T(t){\nu^i}^T(t))\boldsymbol{x}_0^i), t) & \\
\qquad +\delta^i(t) + n(t) \quad \forall\, \boldsymbol{x}_0^i \in D^i, & i = 1, 2, \ldots, K
\end{cases} \tag{8}$$

where $\lambda^i \in \mathbb{R}$, $\delta^i \in \mathbb{R}$, $\nu^i \in \mathbb{R}^3$, $T \in \mathbb{R}^3$, $R \in SO(3)$, $V \in \mathbb{R}^3$, and $\omega \in \mathbb{R}^3$. Let $\omega = [\omega_1, \omega_2, \omega_3]^T$; then $\widehat{\omega} \doteq \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}$ and $\exp(\widehat{\omega})$ is the matrix exponential[5] of $\widehat{\omega}$. $\nu^{i,j}$ stands for the jth component of ν^i. $\alpha_\lambda(t)$ and $\alpha_\delta(t)$ account for the change of illumination, $\alpha_V(t)$ and $\alpha_\omega(t)$ model the unknown translational acceleration and the rotational acceleration respectively, and D^i is the region of the image that corresponds to the approximation of the surface S by the ith planar patch with normal ν^i.

Since we have no knowledge of how $\alpha_V(t)$ and $\alpha_\omega(t)$ change over time, we choose to model them as white noise. As a consequence, the resulting $V(t)$ and $\omega(t)$ will be Brownian-motion processes. Of course, if some prior information is available (for instance when the camera is mounted on a vehicle or a moving robot), then we can use it to further refine our model. The same reasoning applies to $\alpha_\lambda(t)$ and $\alpha_\delta(t)$, which we also consider as white noise. The

[5] The exponential of $\widehat{\omega}$ can be efficiently computed as follows: $\exp(\widehat{\omega}) = I + \frac{\widehat{\omega}}{\|\omega\|}\sin(\|\omega\|) + \frac{\widehat{\omega}^2}{\|\omega\|^2}(1-\cos(\|\omega\|))$ for $\omega \neq 0$; $\exp(\widehat{\omega}) = I$ for $\omega = 0$. This formula is commonly referred to as Rodrigues' formula.

term $n(t)$ is defined as an independent sequence identically distributed in such a way as to guarantee that the measured image I is always positive.

3 Causal estimation of a photo-geometric model

We represent the scene as a rigid collection of planar patches whose projections in images deform according to a projective model, and model the unknown parameters (illumination parameters, plane normals, rigid motion, and velocity) as the state of the nonlinear dynamical system Eq. (8). Causally inferring a model of the scene then corresponds to reconstructing the state of the model Eq. (8) from its output (measured images).

3.1 Observability

It is fundamental to ask whether this reconstruction yields a unique solution or not. In system theory a necessary condition of uniqueness is captured by the concept of *observability*. Since we do not explicitly compute correspondences between planar patches, we shall make some assumptions on the texture of the patches. We define a texture to be *sufficiently exciting* if the constraints it imposes are sufficient to uniquely determine the correspondence for at least four points in *general configuration*[6]. With this definition in hand, we can state the main theoretical result of this paper:

Proposition 1. *If there are two planes with different normals in the scene, the translational velocity is nonzero, and the texture is sufficiently exciting, then the model Eq. (8) is observable.*

We refer to the appendix for the proof.

3.2 Nonlinear filtering and implementation

Observing the nonlinear nature of the state equation and measurement equation of the system Eq. (8), we pose the problem of reconstructing the state of the system from its output in an extended Kalman-filter framework. A necessary step towards an algorithmic implementation is to choose a local coordinate for the dynamical system Eq. (8). To this end, we represent $SO(3)$ in canonical exponential coordinates: let $\Omega = [\Omega_1,\ \Omega_2,\ \Omega_3]^T$ be a vector in $\mathbb{R}^3$; then a rotation matrix can be represented as $R = \exp(\widehat{\Omega})$. Substituting the chosen parameterization, we can rewrite system Eq. (8) in local coordinates as:

$$\begin{cases} \lambda^i(t+1) = \lambda^i(t) + \alpha_\lambda(t) & i = 1, 2, \ldots, K \\ \delta^i(t+1) = \delta^i(t) + \alpha_\delta(t) & i = 1, 2, \ldots, K \\ \nu^{1,j}(t+1) = \nu^{1,j}(t) & j = 1, 2 \\ \nu^i(t+1) = \nu^i(t) & i = 2, 3, \ldots, K \\ T(t+1) = \exp(\widehat{\omega}(t))T(t) + V(t) & \\ \Omega(t+1) = \log_{SO(3)}(\exp(\widehat{\omega}(t)) \exp(\widehat{\Omega}(t))) & \\ V(t+1) = V(t) + \alpha_V(t) & \\ \omega(t+1) = \omega(t) + \alpha_\omega(t) & \\ I(\boldsymbol{x}_0, 0) = \lambda^i(t) I(\pi((\exp(\Omega(t)) + T(t){\nu^i}^T(t))\boldsymbol{x}_0), t) & \\ \quad + \delta^i(t) + n(t)\ \forall\, \boldsymbol{x}_0 \in D^i, & i = 1, 2, \ldots, K \end{cases} \tag{9}$$

where $\log_{SO(3)}(\cdot)$ stands for the inverse of the exponential map[7], i.e. $\Omega \doteq \log_{SO(3)}(R)$ is such that $R = \exp(\widehat{\Omega})$.
In the following paragraphs we will give details of how to initialize the corresponding extended Kalman filter, how to update the filter, and how to add and/or remove planar patches during the estimation process. To streamline the notation, let f and h denote the state and measurement model, ξ denotes the state, and y denotes the measurement, so that the system Eq. (9) can be written in a concise form as:

$$\begin{cases} \xi(t+1) = f(\xi(t)) + w(t) & w(t) \sim \mathcal{N}(0, \Sigma_w), \\ y(t) = h(\xi(t)) + n(t) & n(t) \sim \mathcal{N}(0, \Sigma_n). \end{cases} \tag{10}$$

With respect to Eq. (9) we have added the model noise $w(t) \sim \mathcal{N}(0, \Sigma_w)$ to account for modeling errors.

Initialization

As mentioned in Sect. 2.1, for the dynamical system to be observable, it is necessary and sufficient

[6] We say points are in general configuration if there exist at least four points, among which none of any three are collinear.

[7] The logarithm $\log_{SO(3)}(\cdot)$ can be computed explicitly via the following formula: $\log_{SO(3)}(R) = \frac{B}{\|B\|}\sin^{-1}(\|B\|)$, where $\widehat{B} = \frac{R - R^T}{2}$ for $R \neq R^T$; $\log_{SO(3)}(R) = [0, 0, 0]^T$ for $R = I$; $\log_{SO(3)}(R) = \pi\Omega$, where $\widehat{\Omega}^2 = \frac{R - I}{2}$ for $R = R^T$ and $R \neq I$.

to set the z coordinate of one normal to some positive value. In the context of Kalman filtering, fixing one component of the state can be done in a number of ways. For example, the fixed state can be simply removed from the model, or its error covariance can be set to 0, or a corresponding pseudo-measurement can be added to the measurement equation. As all these techniques are equivalent from a theoretical point of view, we will not make any choice here. Also, for ease of notation, we will write the normals in the state with all three components.

We choose as initial conditions $\lambda_0^i = 1$, $\delta_0^i = 0$, $\nu_0^i = [0\ 0\ 1]^T$, $T_0 = 0$, $\Omega_0 = 0$, $V_0 = 0$, $\omega_0 = 0$, $i = 1, 2 \ldots, K$. For the initial variance P_0, choose it to be zeros for λ^i and δ^i, a large positive number M for each component of ν^i, and zeros corresponding to T and Ω (note that this has effectively fixed the inertial reference 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 100–1000 units of focal length). Since we have explicitly modeled the change of illumination, we set the variance $\Sigma_n(t)$ to be low (typically $(0.05 \times 255)^2$, where 0–255 is the range of intensity values. The variance $\Sigma_w(t)$ is a design parameter that is available for tuning. We describe the procedure to set $\Sigma_w(t)$ in Sect. 3.3. Finally, set

$$\begin{cases} \hat{\xi}(0|0) \doteq [\lambda_0^1 \ldots \lambda_0^N\ \delta_0^1\ \ldots \delta_0^N \\ \qquad {\nu_0^1}^T \ldots {\nu_0^N}^T\ T_0^T\ \Omega_0^T\ V_0^T\ \omega_0^T]^T, \\ P(0|0) = P_0, \end{cases} \tag{11}$$

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

The recursion to update the state ξ and the variance P proceeds as follows (see Eq. (10)):

Prediction:

$$\begin{cases} \hat{\xi}(t+1|t) = f(\hat{\xi}(t|t)), \\ P(t+1|t) = F(t)P(t|t)F^T(t) + \Sigma_w, \end{cases} \tag{12}$$

Update:

$$\begin{cases} \hat{\xi}(t+1|t+1) = \hat{\xi}(t+1|t) + L(t+1)(y(t+1) \\ \qquad -h(\hat{\xi}(t+1|t))), \\ P(t+1|t+1) = \Gamma(t+1)P(t+1|t)\Gamma^T(t+1) \\ \qquad +L(t+1)\Sigma_n(t+1)L^T(t+1), \end{cases} \tag{13}$$

Gain:

$$\begin{cases} \Lambda(t+1) \doteq H(t+1)P(t+1|t)H^T(t+1) \\ \qquad +\Sigma_n(t+1), \\ L(t+1) \doteq P(t+1|t)H^T(t+1)\Lambda^{-1}(t+1), \\ \Gamma(t+1) \doteq I_d - L(t+1)H(t+1), \end{cases} \tag{14}$$

Linearization:

$$\begin{cases} F(t) \doteq \frac{\partial f}{\partial \xi}(\hat{\xi}(t|t)), \\ H(t+1) \doteq \frac{\partial h}{\partial \xi}(\hat{\xi}(t+1|t)), \end{cases} \tag{15}$$

where I_d is the identity matrix.

3.3 Tuning

The variance $\Sigma_w(t)$ is a design parameter and it is chosen to be block diagonal. The blocks corresponding to $T(t)$ and $\Omega(t)$ are also diagonal and have values 10^{-8} to take into account numerical errors in motion integration. 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 λ_0^i equal to σ_λ and to δ_0^i equal to σ_δ. We choose the blocks corresponding to ν_0^i equal to σ_ν and the blocks corresponding to V and ω to be diagonal with element σ_v. σ_v is adjusted relative to σ_ν depending on the desired regularity of the motions. We then vary both σ_ν and σ_ν together with σ_λ and σ_δ, with respect 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^{-4} for σ_λ and $(10^{-4} \times 255)^2$ for σ_δ, while it settles between 10^{-2} to 10^{-3} units of focal length for σ_ν.

3.4 Outlier rejection

We have chosen to model the scene as a collection of planar patches. As such, we need to test the hypothesis that a region of the image corresponds to (is well approximated by) a plane in space. To this end we consider the residual of the matching for each patch. We compute the normalized cross correlation

between the transformed image from time 0 to the current time t and the measured image at the time t and compare it with a fixed threshold.
If the residual is higher than the threshold, we declare it to be an outlier. Due to the nature of the approximation, the test will depend on the size of the regions. Away from discontinuities, the larger the curvature, the smaller the region that will pass the test. By running the test all over the image (or on the portion of it that corresponds to high gradient values in image intensities, so as to eliminate at the outset regions with little or no texture information), we can segment the image into a number of patches that correspond to planar approximations of the surface S. Obviously, discontinuities and occluding boundaries will fail the test and therefore be rejected as outliers.

3.5 Occlusions

Whenever a patch disappears or becomes occluded, we simply remove the corresponding normal from the state. To keep the filter estimation reliable, it is necessary to maintain a minimum number of patches. Hence, we continuously select new candidates. Let τ be the time when the ith patch is selected. We shall reconstruct its normal $\nu_\tau^i(t)$ using a simplified dynamical system:

$$\begin{cases} \nu_\tau^i(t+1) = \nu_\tau^i(t) & t > \tau, \\ I(\boldsymbol{x}_\tau^i, \tau) = \\ \quad I(\pi((R(t,\tau) + T(t,\tau){\nu_\tau^i}^T)\boldsymbol{x}_\tau^i), t) & \forall \boldsymbol{x}_\tau^i \in D_\tau^i, \end{cases} \tag{16}$$

where $\big(T(t,\tau), R(t,\tau)\big)$ denotes the relative pose between time τ and time t, which can be computed via the following equations:

$$\begin{aligned} T(t,\tau) &= T(t|t) - R(t|t)R(\tau|\tau)^{-1}T(\tau|\tau), \\ R(t,\tau) &= R(t|t)R(\tau|\tau)^{-1}, \end{aligned} \tag{17}$$

where $R(t|t) = \exp(\widehat{\Omega}(t|t))$. $\Omega(t|t)$ and $T(t|t)$ are the estimates of the global dynamical system. We have used the subscript τ for ν^i, $\boldsymbol{x}^i$, and D^i to emphasize that they are introduced at time τ. During this preliminary phase we do not consider illumination changes. However, λ^i and δ^i will be added once the novel patches are admitted into the state of the dynamical system Eq. (9).
Let $\nu_\tau^i(t|t)$ denote the estimate (at time t) of the normal of the ith new feature in the reference frame of the camera at time τ. $\nu_\tau^i(t|t)$ is computed by means of an extended Kalman filter based on the model Eq. (16). Its evolution is governed by:

Initialization:

$$\begin{cases} \nu_\tau^i(\tau|\tau) = [\,0\;0\;1\,]^T, \\ P_\tau^i(\tau|\tau) = M, \end{cases} \tag{18}$$

Prediction:

$$\begin{cases} \nu_\tau^i(t+1|t) = \nu_\tau^i(t|t), \\ P_\tau^i(t+1|t) = P_\tau^i(t|t) + \Sigma_w(t), \end{cases} \quad t > \tau \tag{19}$$

Update:

$$\begin{aligned} &\nu_\tau^i(t+1|t+1) = \nu_\tau^i(t+1|t) + L_\tau(t+1)(I^i(t+1) \\ &\quad - I^i(t+1|t)), \end{aligned}$$

where

$$\begin{aligned} &I^i(t+1|t) = I(\pi((R(t|t)R(\tau|\tau)^{-1} + (T(t|t) \\ &\quad - R(t|t)R(\tau|\tau)^{-1}T(\tau|\tau))\nu_\tau^i(t+1|t)^T)\boldsymbol{x}_\tau^i), t), \end{aligned} \tag{20}$$

where P_τ^i is updated according to a Riccati equation similar to Eq. (13).
After a probation period δt, the normals relative to patches passing the outlier-rejection test described in Sect. 3.4 are inserted into the state of Eq. (9) using the following transformation:

$$\nu_0^i = \frac{R(\tau|\tau)^{-1}\nu_\tau^i(\tau+\delta t|\tau+\delta t)}{1 - T(\tau|\tau)^T\nu_\tau^i(\tau+\delta t|\tau+\delta t)}. \tag{21}$$

The measurements are back-projected to time 0 from time τ through the following relationship:

$$\begin{aligned} I(\boldsymbol{x}_0^i, 0) = I(\pi((R(\tau|\tau) + T(\tau|\tau){\nu_0^i}^T)\boldsymbol{x}_0^i), \tau) \\ \forall\, \boldsymbol{x}_0^i \in D_0^i. \end{aligned} \tag{22}$$

3.6 Drift

Recall that in order to solve the scale-factor ambiguity we have chosen to fix the z coordinate of one normal. We shall call the patch corresponding to the selected normal the *reference patch*. As long as the reference patch is visible, all the states will be estimated according to it. However, when one reference patch disappears, another reference patch has to be chosen and the z coordinate of its normal has to be

fixed. Since we do not have the exact value of the new fixed component with respect to the previous one, using its current estimate necessarily introduces an error that will propagate to all the other states. In particular, this error affects the current global motion estimates $R(t)$ and $T(t)$. Therefore, any time the reference patch disappears or is occluded, our observation of motion and structure accumulates a drift which is not bounded in time. Notice that it does not make a difference whether the scale factor is associated to one particular planar patch or to a collective property of all patches.

As we discussed, this drift does not occur if at least one patch is visible from the beginning to the end of the sequence (and it happens to be the reference patch). While this is unlikely in any real sequence, it is often the case that reference patches that disappear become visible again. This can be because they become unoccluded, or due to the relative motion between the camera and the object (e.g. the viewer returns to a previously visited position). The re-appearing of reference patches carries substantial information because it allows us to compensate for the drift in the estimates. In order to exploit this information one must be able to match patches that were visible at previous times during the sequence. In the next subsection we describe how this can be done.

3.7 Global registration

Every time a reference patch disappears, we store the geometric representation of the patch (coordinates of the center and normal to the plane in the inertial reference frame) as well as the photometric representation (the texture patch it supports). When the camera motion is such that the stored reference feature becomes visible again (e.g. when there is a loop in the trajectory), we match the stored texture within a region corresponding to the predicted position in the current frame. If a high score is achieved, we conclude that the old patch reappeared. Once this decision is made, we use the difference between the matched position and the predicted position to compensate for the drift. Note that when there are multiple matches, only the 'oldest' reference patch (the first reference patch in time) carries the information, because all later ones are fixed with respect to this one.

Let $\boldsymbol{x}_\tau$ and ν_τ be respectively the center and the normal to the plane of the oldest reference patch we have stored, and let $\tilde{\boldsymbol{x}}$ be the matched position. Then, we have the following relationship:

$$(R(t,\tau) + \beta T(t,\tau)\nu_\tau^T)\boldsymbol{x}_\tau = \varepsilon\tilde{\boldsymbol{x}}, \tag{23}$$

where ε is the ratio between the depth of the reference patch at time τ and the depth of the same patch at time τ, and β is the scale-factor drift. Due to the noise in determining the matched position, Eq. (23) does not hold exactly. Therefore, we look for β and ε that minimize the distance between the estimated position and the matched position:

$$\hat{\beta}, \hat{\varepsilon} = \arg\min_{\beta,\varepsilon} \left\| (R(t,\tau) + \beta T(t,\tau)\nu_\tau^T)\boldsymbol{x}_\tau - \varepsilon\tilde{\boldsymbol{x}} \right\|^2, \tag{24}$$

where we have used an SSD-type (sum of squared differences) error. The optimal β and ε can be computed using least squares as follows:

$$\begin{pmatrix} \hat{\beta} \\ \hat{\varepsilon} \end{pmatrix} = \left(\left[-T(t,\tau)\nu_\tau^T\boldsymbol{x}_\tau \;\; \tilde{\boldsymbol{x}} \right]^T \left[-T(t,\tau)\nu_\tau^T\boldsymbol{x}_\tau \;\; \tilde{\boldsymbol{x}} \right] \right)^{-1} \times \left[-T(t,\tau)\nu_\tau^T\boldsymbol{x}_\tau \;\; \tilde{\boldsymbol{x}} \right]^T R(t,\tau)\boldsymbol{x}_\tau. \tag{25}$$

Once the scale drift is computed, we update the value of the fixed coordinate as:

$$\nu_3^1 = \beta\tilde{\nu}_3^1, \tag{26}$$

where $\tilde{\nu}_3^1$ is the current value. Note that the rest of the states will be continuously estimated and updated according to the newly fixed value.

The global registration performed at a certain instant of time does not affect the entire trajectory, but only the current pose of the camera relative to the inertial frame. This is because – in a causal recursive framework – we are only concerned with the estimate of shape and motion at the present point in time. If off-line operation is allowed, one may want to re-adjust the entire trajectory, but this is beyond the scope of this paper [18].

As the length of the experiment grows, matching the entire database at each novel frame becomes unfeasible for real-time applications. Since we assume that the sequence is taken with a calibrated camera, at each time instant the field of view of the camera can be computed in the inertial frame, and all features that fall outside the visibility cone can be discarded at the outset.

The visibility of each patch with respect to the camera can be computed based on its normal at the initial time. More precisely, if $\boldsymbol{x}_\tau{}^T R(t,\tau)\nu_\tau > 0$ we declare the point to be visible, otherwise we declare it occluded. Finally, to speed up the search, we restrict our matching area to a neighborhood of the predicted position of each patch in the database (e.g. regions of interest with a radius of 10 pixels).
Although the drift reduction can be made more and more sophisticated by considering robust statistics, soft-matching, and a number of other statistical techniques, we found that the procedure described above is a good compromise between accuracy and computational efficiency.

4 Experiments

Establishing the performance of a structure from motion algorithm is in general not an easy task due to the complexity of the estimated parameters and the large number of possible scenarios. Aware that this is still far from being a comprehensive analysis of the algorithm, we choose to test our algorithm on the error of both structure and motion under a few representative cases, namely *forward motion*, *sideways motion*, and *fixating motion* on synthetic data-sets. For real data, since we do not have the true values for motion and structure when running the algorithm, we choose to evaluate the estimation process using periodic motions and measuring the motion error at the end of a single period (i.e. when the camera reference system is expected to come back to the initial position).

4.1 Structure error

The structure estimated with our algorithm consists in a set of normal vectors of planar patches selected from the initial image of the scene. In our synthetic experiments we generate three planes and textures associated with them. We place the center point of one planar patch at depth 1 m. This patch is also used to fix the scale factor of the whole estimation process (i.e. the z coordinate of the normal vector is fixed to 1). We run the filter on a sequence of 200-frames long and plot the mean and standard deviation of the error between the estimated structure and the ground truth in Fig. 1. Among the three kinds of motions, the error corresponding to the fixating motion is the lowest (of the order of 3 mm), while the error for sideways motions grows by a factor of three (of the order of 10 mm). The error corresponding to the forward motion is the highest (of the order of 30 mm), which we attribute to the presence of local minima observed for instance in [6, 15].

4.2 Motion error

Exploiting the periodic nature of the chosen motion, we determine the accuracy of the estimates by measuring the distance between the estimated pose (rotation and translation) of the camera at the end of a motion period and its initial position. In particular, the translation error is the norm of the difference between the estimated translation and the true one, and the rotation error is measured through the Frobenius norm of the discrepancy between the true rotation R and the estimated one $\hat{R}$: $\| I_d - \hat{R} R^T \|_{\mathrm{F}}^2$. In our case $T = 0$ and $R = I_d$. In Fig. 2, we plot the motion error of both the synthetic data and the real data. The motion error of fixating motion and sideways translation is comparable, while the error of forward translation is the highest. This is consistent with the structure error observed in Sect. 4.1.

4.3 Drift reduction

In Fig. 3 we show a few images of a sequence obtained by moving a camera around an object (the actual motion is performed by rotating the object on a turntable, which is equivalent to moving a camera around it). This motion is designed in such a way that no feature remains visible throughout the course of the experiment. Therefore, drift accumulates, as can be seen in Fig. 4.
The actual trajectory of the camera is a circle that passes through the origin, but the estimated trajectory misses the origin due to the scale-factor drift. Even though the drift may seem small when visualized in terms of the estimates of motion, it severely affects the estimates of shape, since it results in misalignment of photometric patches and, therefore, spoils the meaningful merging of estimates from multiple passes around the object. By matching visible features, however, the drift can be compensated for, as shown in the solid line at the bottom of Fig. 4. Failure to perform global registration results in a significant drift during the second pass around the object, shown as a dotted line.

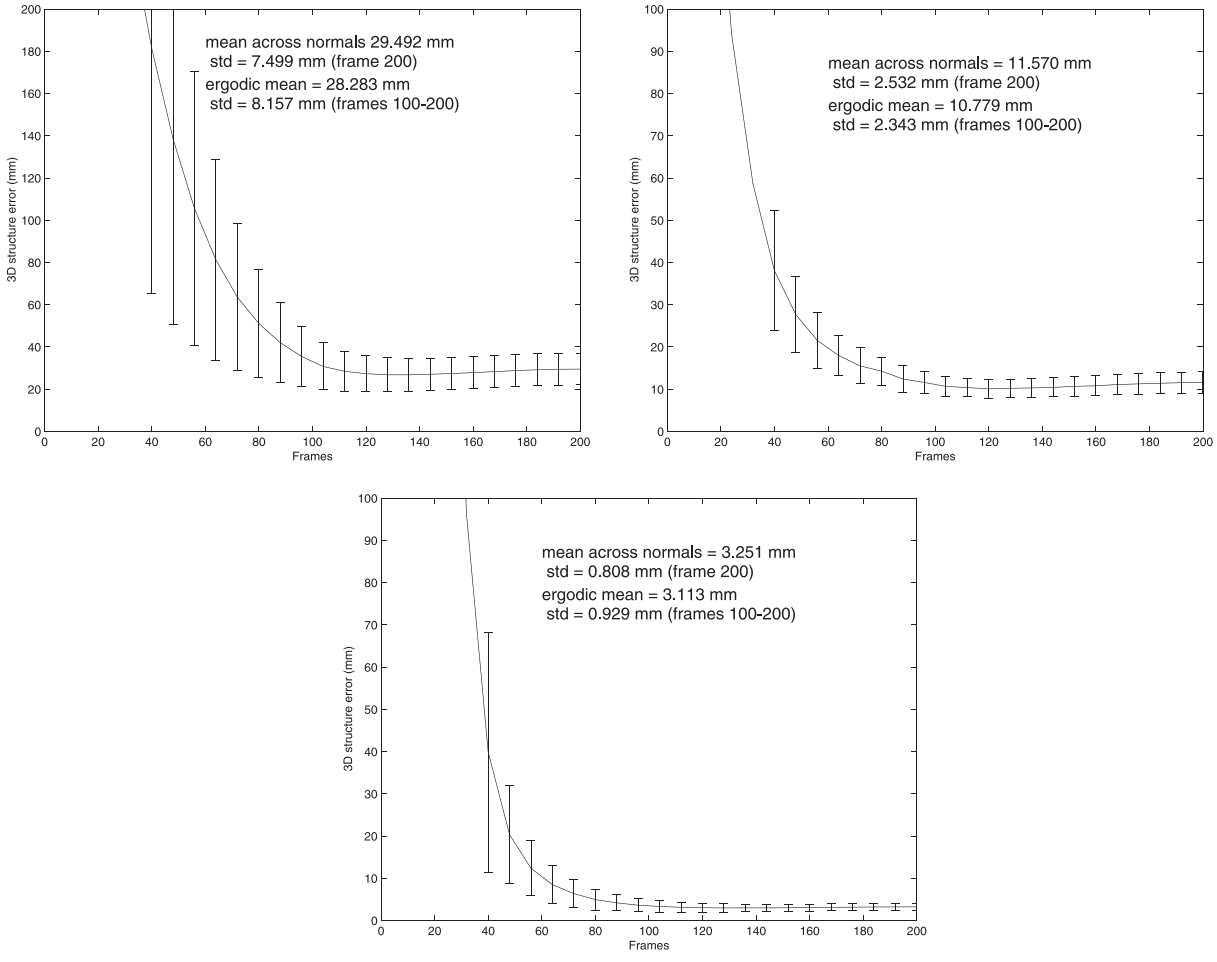

Fig. 1. Structure error: three different motions are tested on the same simulated scene with known ground truth. 30 trials of 200 frames each are performed. The error in mutual distance between the estimates and the ground truth of a set of 15 planar patches is plotted. The *top-left figure* shows the structure error for forward translation (periodic translation along the z axis); the *top-right figure* shows the structure error for sideways translation (periodic translation along the x axis); the *bottom figure* shows the structure error for fixating motion (points rotating rigidly around an axis passing through their center of mass). Note that while the sideways and fixating motion graphs share the same axis scale, the forward motion ordinate axis scale is doubled

Once registered, different sequences around the object can be merged and the shape (position and orientation of planar patches) and photometry (texture supported on such planes) can be reconstructed. In Fig. 5 we overlay the estimates to a set of images of the object, to show that the texture patches nicely align to the appearance of the object. Note that the illumination changes have been estimated and corrected.

5 Conclusions

We have presented a novel recursive algorithm to estimate structure and motion. The input to the algorithm is a sequence of images collected in a causal fashion, and the output is the collective rigid motion and a structural representation of the scene.

Our algorithm integrates visual information in space as well as in time, by using a finitely parameterizable

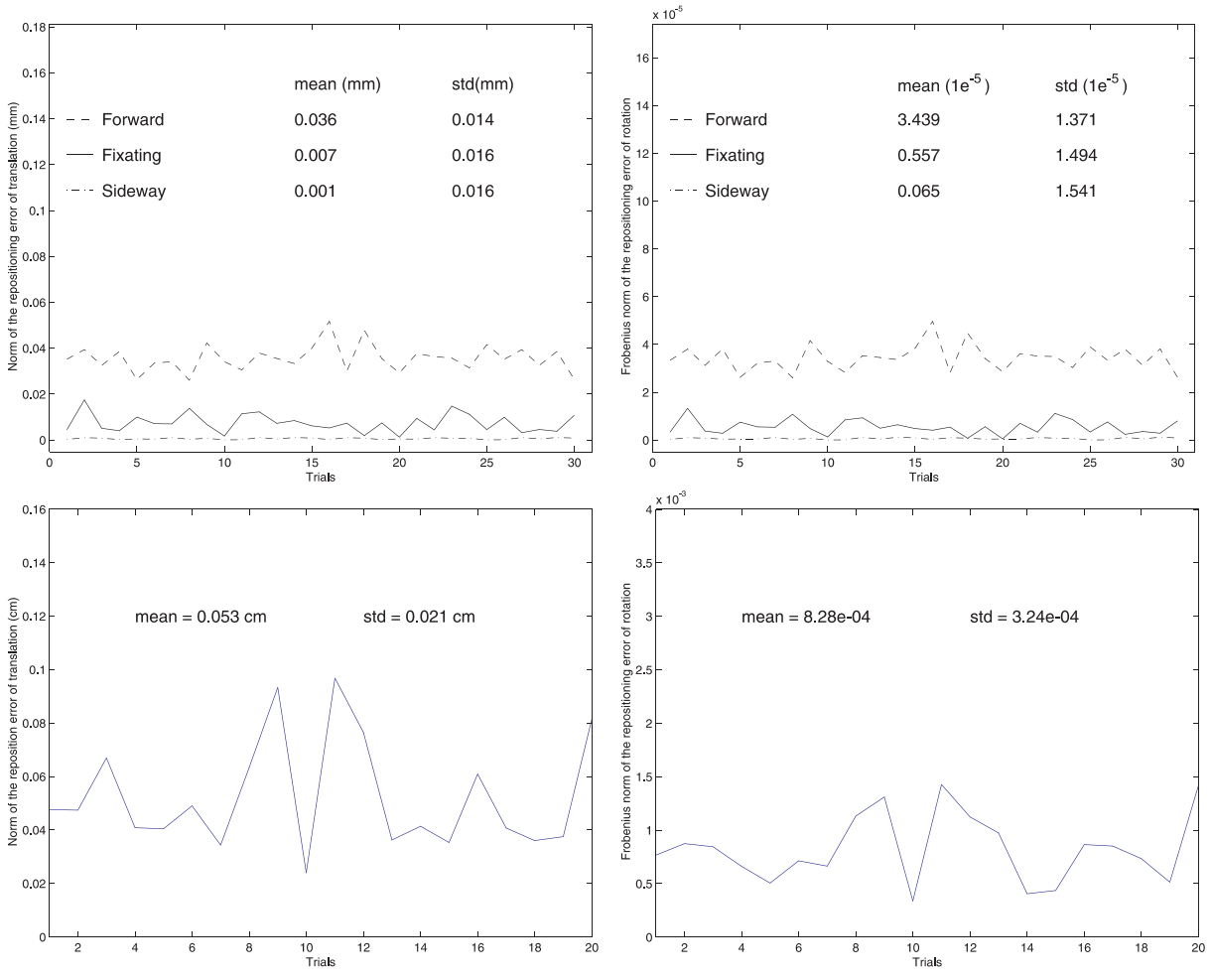

Fig. 2. Motion error. Synthetic data (*top*): the three types of motion we consider are periodic in time. The motion estimation error is thus defined as the repositioning error of the camera after a number of complete cycles. We show the error for the three types of motion with 30 trials. **Motion error. Real data** (*bottom*): the same conditions simulated in the experiments reported in the *top plots* have been recreated on a real scene. A set of objects are placed on a turntable and 20 sequences of periodic fixating motions are recorded. The camera is positioned about 1 m away from the turntable center. The repositioning error of the camera motion after a complete cycle is shown for 20 trials

class of geometric and photometric models for the scene. Image-region tracking and three-dimensional motion estimation are then combined into a closed loop. We then cast the problem of structure from motion in the framework of nonlinear filtering. The unknown structure and motion are estimated by reconstructing the state of a nonlinear dynamical system via an extended Kalman filter. Furthermore, we have shown that the dynamical system is observable under the assumptions that the scene contains at least two planar patches with different normal directions and sufficiently exciting texture, and the translational velocity is nonzero. The recursive nature of our algorithm makes it suitable for real-time implementation. Our algorithm also returns an estimate of the appearance of the scene as seen from an arbitrary pose, and could therefore be used for on-line construction of three-dimensional image mosaics. We also use

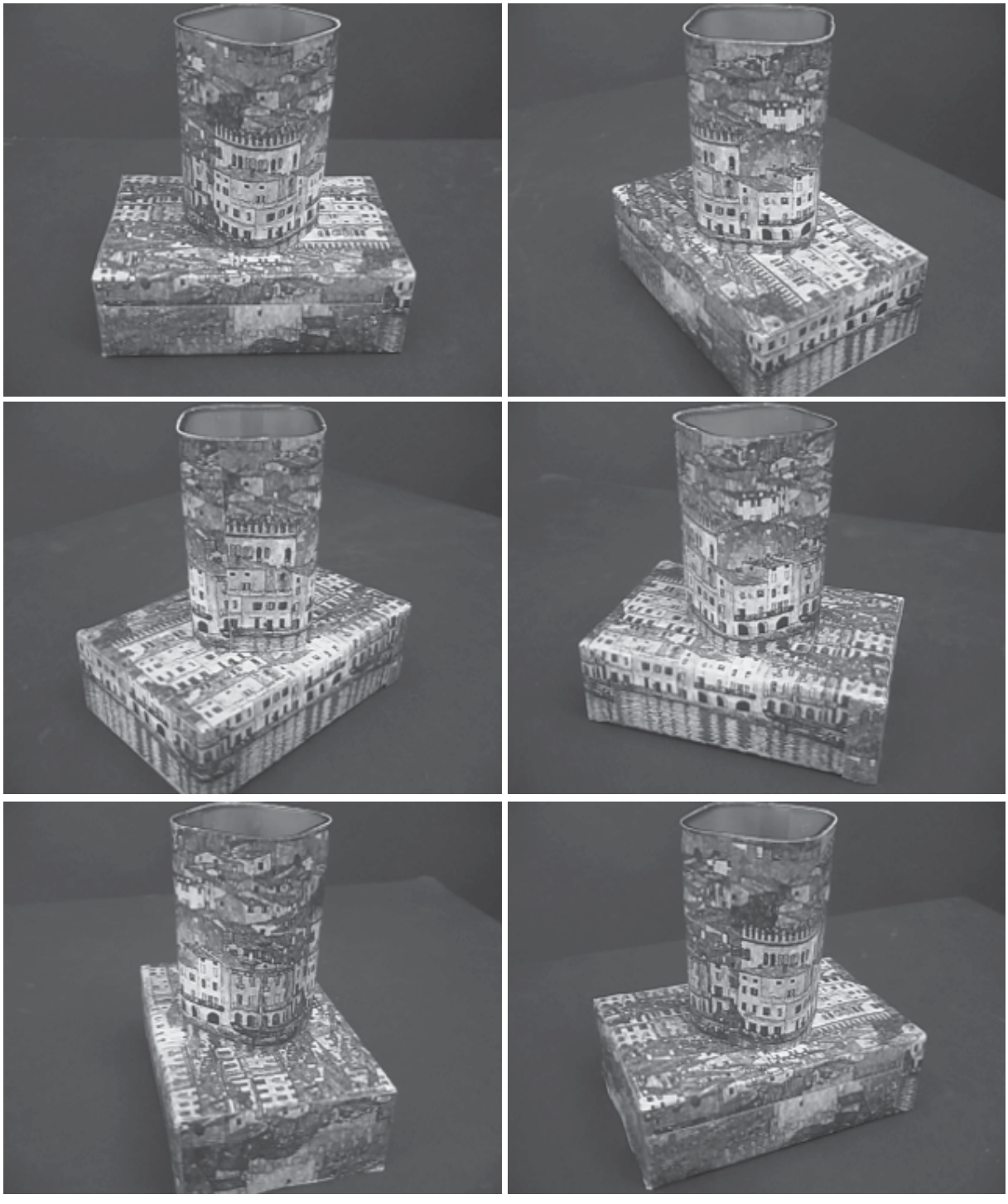

Fig. 3. Original data-set: the camera moves around the object so that no feature remains visible throughout the course of the sequence. The sequence is 800-frames long

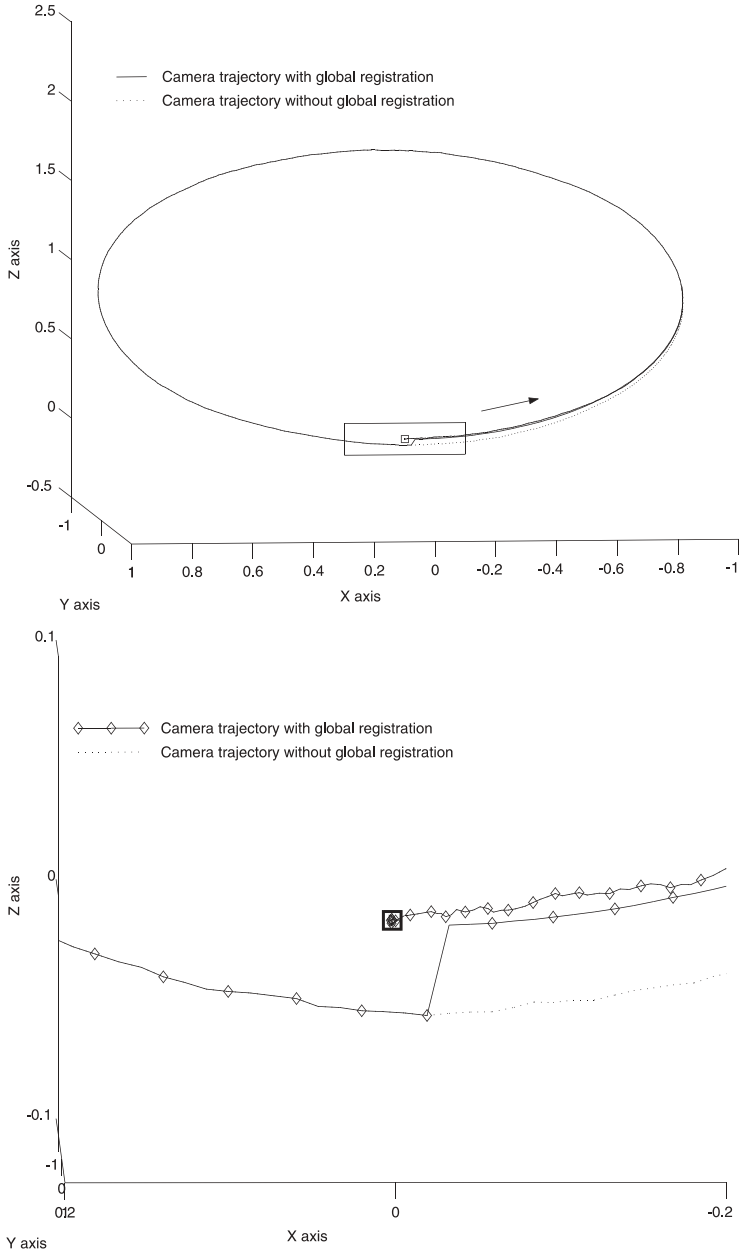

Fig. 4. Causally estimated spatial trajectory for a sequence of images (samples of which are shown in Fig. 5). The trajectory of the camera surrounds the object so that no features survive from the beginning to the end of the experiment. Despite the fact that the camera goes back to the original configuration, the estimated trajectory does not reach the origin (*top*). This can be seen in the detail image (*bottom*). This is unavoidable since no visual features are present from the beginning to the end of the sequence. However, starting from frame 524, several features that were visible at some point become visible again. Our filter stores both the pose and the orientation of the planar patches that become occluded, as well as the texture patch that they support. Matching the current field of view with stored features allows us to globally register the trajectory and effectively eliminate the drift. Not imposing global registration results in a drift, shown in the *dotted line*, during a second pass around the object

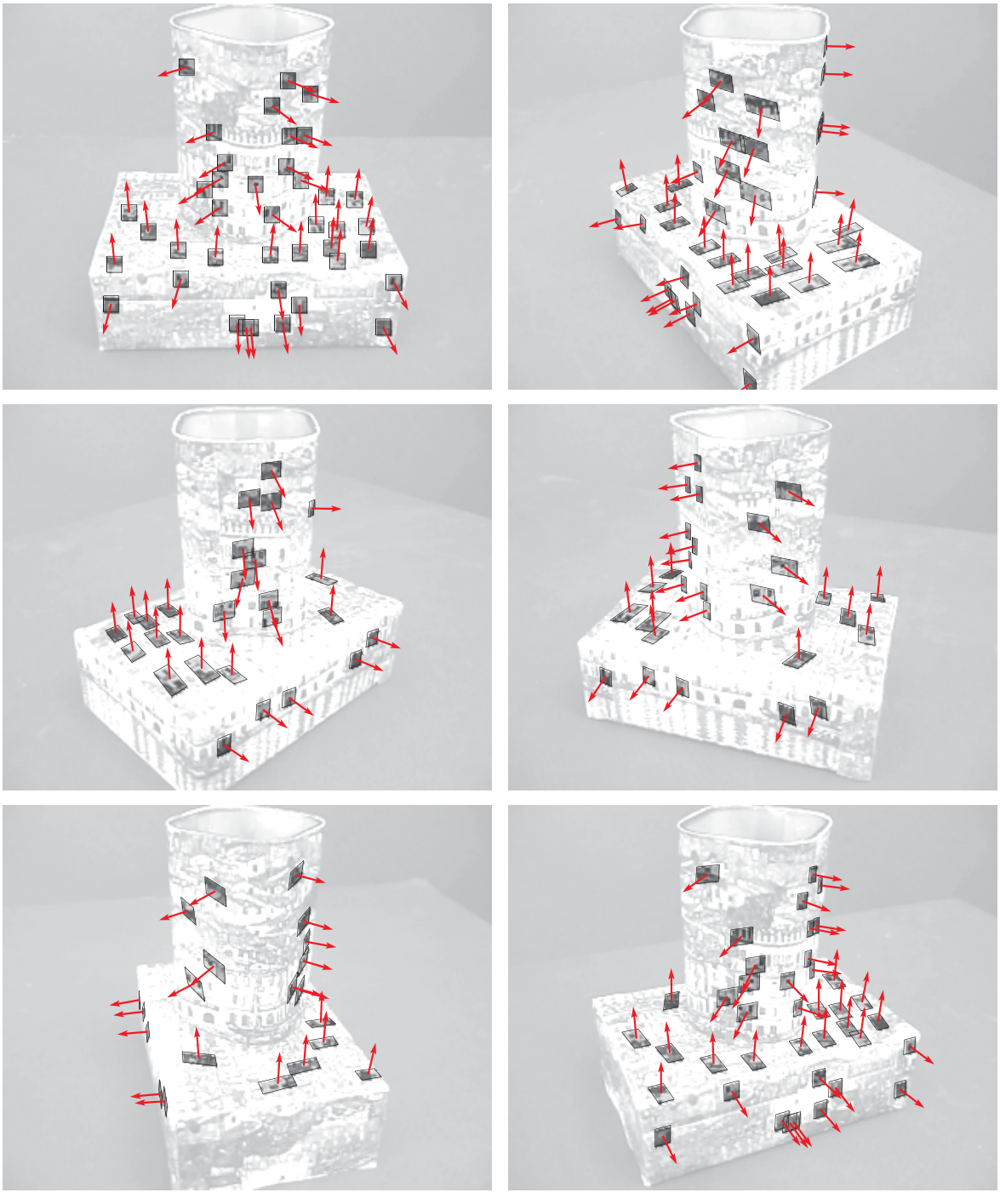

Fig. 3. Estimated representation of the scene: each feature corresponds to a planar patch represented by a point and a normal vector. The proposed filter estimates the geometric parameters and stores the texture patch that is supported on the planar feature. A few views of the reconstructed geometry (normal vectors) and texture (texture patches registered to the estimated pose of the corresponding planes) are superimposed to contrast-reduced views of the original scene to show that the texture patches capture the local appearance of the object

this estimate to globally align the motion estimates in long sequences, since the appearance of features once seen can be used to match the same features at the current time in similar position and orientation. This allows for compensating the drift in the motion estimates. To improve the computational efficiency, we develop some heuristic strategies to avoid matching features that are not visible.

Acknowledgements. This research is supported in part by NSF grant IIS-9876145, ARO grant DAAD19-99-1-0139, ONR grant N00014-02-1-0720, and Intel grant 8029.

Appendix

To prove Proposition 1, we need first to introduce the following lemma:

Lemma 1. *Given the set* $\{\rho^1, \rho^2, U, V, \nu^1, \nu^2\}$, $M^i \doteq \rho^i(U + V\nu^{i^T})$, $i = 1, 2$, *where* $\rho^i \in \mathbb{R}$, $\rho^i \neq 0$, $i = 1, 2$, $U \in SO(3)$, $\nu^1, \nu^2, V \in \mathbb{R}^3$, $\nu^1, \nu^2 \neq 0$, $V \neq 0$, $\nu^1 \neq \nu^2$, *and* $\nu_3^1 = 1$, *then the set* $\{(\bar\rho^1, \bar\rho^2, \bar U, \bar V, \bar\nu^1, \bar\nu^2) \,|\, M^i = \rho^i(U + V\nu^{i^T}) = \bar\rho^i(\bar U + \bar V\bar\nu^{i^T}),\ i = 1, 2,\ \bar\rho^1, \bar\rho^2 \in \mathbb{R}, \bar U \in SO(3), \bar V, \bar\nu^1, \bar\nu^2 \in \mathbb{R}^3, \bar\nu_3^1 = 1\}$ $= \left\{(\rho^1, \rho^2, U, V, \nu^1, \nu^2), \left(-\rho^1, -\rho^2, \left(\frac{2VV^T}{\|V\|^2} - I_d\right)U, \alpha V, -\frac{1}{\alpha}\left(\nu^1 + \frac{2U^TV}{\|V\|^2}\right), -\frac{1}{\alpha}\left(\nu^2 + \frac{2U^TV}{\|V\|^2}\right)\right)\right\}$, *where* α *is chosen such that the* z *coordinate of* $-\frac{1}{\alpha}\left(\nu^1 + \frac{2U^TV}{\|V\|^2}\right)$ *is* 1.

Proof. First, we will show that $|\bar\rho^i| = |\rho^i|$, $i = 1, 2$. For $i = 1$, we have

$$\rho^1(U + V\nu^{1^T}) = \bar\rho^1(\bar U + \bar V\bar\nu^{1^T}).$$

Multiplying both sides from the right by $\nu_\perp$, a vector such that $\nu_\perp \perp \nu^1$, $\nu_\perp \perp \bar\nu^1$, and $\nu_\perp \neq 0$, and then taking the norm of both sides, we have

$$|\rho^1|\|U\nu_\perp\| = |\bar\rho^1|\|\bar U\nu_\perp\|,$$

which yields $|\bar\rho^1| = |\rho^1|$. Second, we will show that $\bar\rho^1 = \rho^1$ and $\bar\rho^2 = \rho^2$, or $\bar\rho^1 = -\rho^1$ and $\bar\rho^2 = -\rho^2$. We will prove by contradiction that the other two choices are not feasible. Without loss of generality, we consider the case $\bar\rho^1 = \rho^1$ and $\bar\rho^2 = -\rho^2$, where we have

$$\rho^1(U + V\nu^{1^T}) = \rho^1(\bar U + \bar V\bar\nu^{1^T}),$$
$$\rho^2(U + V\nu^{2^T}) = -\rho^2(\bar U + \bar V\bar\nu^{2^T}).$$

Multiplying both equations from the left by $V_\perp^T$ ($V_\perp \perp V$, $V_\perp \perp \bar V$, and $V_\perp \neq 0$), we have:

$$V_\perp^T U = V_\perp^T \bar U \quad \text{and} \quad V_\perp^T U = -V_\perp^T \bar U,$$

which is a contradiction. Now we will show that for $\bar\rho^1$ and $\bar\rho^2$ fixed, the set $\{\bar\rho^1, \bar\rho^2, \bar U, \bar V, \bar\nu^1, \bar\nu^2\}$ is unique. Assume that there is another set $\{\bar\rho^1, \bar\rho^2, \tilde U, \tilde V, \tilde\nu^1, \tilde\nu^2\}$ that satisfies the following identities:

$$M^1 = \bar\rho^1(\bar U + \bar V\bar\nu^{1^T}) = \bar\rho^1(\tilde U + \tilde V\tilde\nu^{1^T}),$$
$$M^2 = \bar\rho^2(\bar U + \bar V\bar\nu^{2^T}) = \bar\rho^2(\tilde U + \tilde V\tilde\nu^{2^T}).$$

Eliminating $\bar U$ and $\tilde U$ we have:

$$\bar V(\bar\nu^{1^T} - \bar\nu^{2^T}) = \tilde V(\tilde\nu^{1^T} - \tilde\nu^{2^T}).$$

Since $\bar\nu^1 \neq \bar\nu^2$ and $\bar V \neq 0$, this implies $\exists\alpha \in \mathbb{R}, \alpha \neq 0 : \tilde V = \alpha\bar V$, and it follows that $\tilde U = \bar U$, $\tilde\nu^1 = \frac{1}{\alpha}\bar\nu^1$, and $\tilde\nu^2 = \frac{1}{\alpha}\bar\nu^2$. Recalling that the z coordinate of both $\tilde\nu^1$ and $\bar\nu^1$ has to be 1, we have $\alpha = 1$. Finally, it is easy to verify that the following choice:

$$\begin{aligned} \tilde\rho^i &= -\rho^i \quad i = 1, 2 \\ \tilde U &= \left(\tfrac{2VV^T}{\|V\|^2} - I_d\right)U \\ \tilde V &= \alpha V \\ \tilde\nu^i &= -\tfrac{1}{\alpha}\left(\nu^i + \tfrac{2U^TV}{\|V\|^2}\right) \quad i = 1, 2 \end{aligned} \tag{27}$$

where α is such that $\tilde\nu_3^1 = 1$, is valid with respect to the statement, which concludes the proof. □

Remark 1. *The previous lemma says that the factorization of two matrices* $\{M^1, M^2\}$ *into* $\{\rho^1, \rho^2, U, V, \nu^1, \nu^2\}$ *has only two solutions. However, it is easy to show that one of the two solutions corresponds to having the structure behind the camera (i.e. it is not visible). Thus, Lemma 1 suggests that, to uniquely reconstruct the structure and motion from two homographies, we must require that the scene is in front of the viewer.*

However, we shall show that in our filtering framework it is not necessary to impose such a constraint, as the model Eq. (8) is already observable.

Proof of Proposition 1

Proof. As far as observability is concerned, we set $\alpha_\lambda = 0$, $\alpha_\delta = 0$, $n(t) = 0$, and $\omega(t) = 0$. Consider

a patch at any time instant t. If the texture is sufficiently exciting, we know, by the definition, that we can determine the correspondence of at least four points between time 0 and time t. This, in turn, can be used to establish a unique homography M [28].
Consider two initial conditions $\{0,\ I_d,\ \omega,\ V,\ \nu^1,\ \nu^2\}$ and $\{0,\ I_d,\ \bar{\omega},\ \bar{V},\ \bar{\nu}^1,\ \bar{\nu}^2\}$. For them to be indistinguishable, we must have at any time t:

$$M^i(t) = \rho^i(t)(R(t) + T(t)\nu^{i^T}) = \bar{\rho}^i(t)(\bar{R}(t) + \bar{T}(t)\bar{\nu}^{i^T}). \quad (28)$$

In particular, when $t = 1$, $R = U = \exp(\widehat{\omega})$, $T(1) = V$, $\bar{R} = \bar{U} = \exp(\widehat{\bar{\omega}})$, and $\bar{T}(1) = \bar{V}$:

$$M^1(1) = \rho_1^1(U + V\nu^{1^T}) = \bar{\rho}_1^1(\bar{U} + \bar{V}\bar{\nu}^{1^T}), \quad (29)$$

$$M^2(1) = \rho_1^2(U + V\nu^{2^T}) = \bar{\rho}_1^2(\bar{U} + \bar{V}\bar{\nu}^{2^T}). \quad (30)$$

By assumption $\nu^1 \neq \nu^2$ and $V \neq 0$; then from Lemma 1 we know that there is only one set of $\{\bar{\rho}_1^1, \bar{\rho}_1^2, \bar{U}, \bar{V}, \bar{\nu}^1, \bar{\nu}^2\}$ with $\bar{\nu}^i \neq \nu^i, i = 1, 2$ satisfying Eq. (29) and Eq. (30). In particular, $\exists \alpha_1 \in \mathbb{R}$ and $\alpha_1 \neq 0$ such that:

$$\bar{\nu}^i = -\frac{1}{\alpha_1}\left(\nu^i + \frac{2U^T V}{\|V\|^2}\right) \quad i = 1, 2.$$

Therefore, we have:

$$\nu^{1^T} + \alpha_1\bar{\nu}^{1^T} = \nu^{2^T} + \alpha_1\bar{\nu}^{2^T}. \quad (31)$$

We will show that this will lead to a contradiction.
Consider the time $t = 2$: $R(2) = U^2$ and $T(2) = UV + V$. The indistinguishability condition is as follows:

$$M^1(2) = \rho_2^1(U^2 + (UV + V)\nu^{1^T}) = \bar{\rho}_2^1(\bar{U}^2 + (\bar{U}\bar{V} + \bar{V})\bar{\nu}^{1^T}), \quad (32)$$

$$M^2(2) = \rho_2^2(U^2 + (UV + V)\nu^{2^T}) = \bar{\rho}_2^2(\bar{U}^2 + (\bar{U}\bar{V} + \bar{V})\bar{\nu}^{2^T}). \quad (33)$$

If $UV + V \neq 0$, we can apply again Lemma 1 at the second step, and we have $\exists \alpha_2 \in \mathbb{R}$ and $\alpha_2 \neq 0$, such that:

$$\bar{\nu}^i = -\frac{1}{\alpha_2}\left(\nu^i + \frac{2(U^T)^2(UV + V)}{\|UV + V\|^2}\right) \quad i = 1, 2.$$

Therefore, we arrive at:

$$\nu^{1^T} + \alpha_2\bar{\nu}^{1^T} = \nu^{2^T} + \alpha_2\bar{\nu}^{2^T}. \quad (34)$$

Considering both equations Eqs. (31) and (34) we conclude immediately that $\alpha_1 = \alpha_2$. Multiplying Eq. (29) on the left by U and subtracting (32), we have:

$$U\bar{U} - \bar{U}^2 = \frac{-2VV^T U}{\|V\|^2}. \quad (35)$$

The right-hand side of Eq. (35) is a rank-one matrix. This conflicts with the fact that the difference of two rotation matrices cannot have rank one.
If $UV + V = 0$, then $U^2V + UV + V \neq 0$, since $V \neq 0$. We can apply Lemma 1 on the time $t = 3$ and will reach a contradiction in a similar way. This concludes the proof. □

References

1. Adiv G (1998) Determining 3-d motion and structure from optical flow generated by several moving objects. IEEE Trans Pattern Anal Mach Intell 7(4):384–401
2. Alon J, Sclaroff S (2000) Recursive estimation of motion and planar structure. IEEE Comput Vision Pattern Recogn II:550–556
3. Azarbayejani A, Pentland AP (1995) Recursive estimation of motion, structure, and focal length. IEEE Trans Pattern Anal Mach Intell 17(6):562–575
4. Bartlett MS (1956) An introduction to stochastic processes. Cambridge University Press
5. Broida TJ, Chellappa R (1986) Estimation of object motion parameters from noisy images. IEEE Trans Pattern Anal Mach Intell 8(1):90–99
6. Chiuso A, Brockett R, Soatto S (2000) Optimal structure from motion: local ambiguities and global estimates. Int J Comput Vision 39(3):195–228
7. Dellaert F, Seitz S, Thorpe C, Thrun S (2000) Structure from motion without correspondence. Proc IEEE Comput Vision Pattern Recogn II:557–564
8. Dickmanns ED, Graefe V (1988) Applications of dynamic monocular machine vision. Mach Vision Appl 1:241–261
9. Hanna KJ (1991) Direct multi-resolution estimation of ego-motion and structure from motion. Workshop on Visual Motion, pp 156–162
10. Jin H, Favaro P, Soatto S (2000) Real-time 3-d motion and structure of point features: front-end system for vision-based control and interaction. Proc IEEE Comput Vision Pattern Recogn II:778–779
11. Jin H, Favaro P, Soatto S (2001) Real-time feature tracking and outlier rejection with changes in illumination. In: Proceedings International Conference on Computer Vision I, pp 684–689
12. Matthies LH, Szeliski R, Kanade T (1989) Kalman filter-based algorithms for estimating depth from image sequences. Int J Comput Vision 3(3):209–238
13. McLauchlan PF (1999) Gauge invariance in projective 3d reconstruction. Workshop on Multi-View Modeling and Analysis of Visual Scenes, pp 37–44

14. McLauchlan PF (2000) A batch/recursive algorithm for 3d scene reconstruction. IEEE Comput Vision Pattern Recogn II:738–743
15. Oliensis J (2000) A new structure-from-motion ambiguity. IEEE Trans Pattern Anal Mach Intell 22(7):685–700
16. Philip J (1991) Estimation of three-dimensional motion of rigid objects from noisy observations. IEEE Trans Pattern Anal Mach Intell 13(1):61–66
17. Poelman CJ, Kanade T (1997) A paraperspective factorization for shape and motion recovery. IEEE Trans Pattern Anal Mach Intell 19(3):206–218
18. Rahimi A, Morency LP, Darrell T (2001) Reducing drift in parametric motion tracking. In Proceedings International Conference on Computer Vision I, pp 315–322
19. Sawhney HS (1994) Simplifying motion and structure analysis using planar parallax and image warping. International Conference on Pattern Recognition A, pp 403–408
20. Soatto S (1994) Observability/identifiability of rigid motion under perspective projection. CDC, pp 3235–3240
21. Soatto S (1997) 3-d structure from visual motion: modeling, representation and observability. Automatica 33:1287–1312
22. Soatto S, Perona P (1998) Reducing structure-from-motion: a general framework for dynamic vision part 1: modeling. IEEE Trans Pattern Anal Mach Intell 20(9):933–942
23. Spetsakis M, Aloimonos JY (1988) Optimal computing of structure from motion using point correspondences in two frames. International Conference on Computer Vision, pp 449–453
24. Sturm PF (2000) Algorithms for plane-based pose estimation. IEEE Comput Vision Pattern Recogn I:706–711
25. Szeliski R, Kang SB (1995) Direct methods for visual scene reconstruction. IEEE Workshop on Representation of Visual Scenes, pp 26–33
26. Thomas JI, Oliensis J (1992) Recursive multi-frame structure from motion incorporating motion error. Image Understanding Workshop, pp 507–513
27. Tomasi C, Kanade T (1992) Shape and motion from image streams under orthography: a factorization method. Int J Comput Vision 9(2):137–154
28. Weng J, Ahuja N, Huang TS (1991) Motion and structure from point correspondences with error estimation: planar surfaces. IEEE Trans Signal Process 39(12):2691–2717
29. Weng J, Ahuja N, Huang TS (1993) Optimal motion and structure estimation. IEEE Trans Pattern Anal Mach Intell 15(9):864–884
30. Xu G, Terai JI, Shum HY (2000) A linear algorithm for camera self-calibration, motion and structure recovery for multi-planar scenes from two perspective images. IEEE Comput Vision Pattern Recogn II:474–479

Photographs of the authors and their biographies are given on the next page.

HAILIN JIN received the Bachelor of Science degree in Automation from Tshingua University, Beijing, China in 1998 and the Master of Science degree in Electrical Engineering from Washington University, Saint Louis in 2000. He is currently a PhD student in the Department of Electrical Engineering, Washington University. Since July 2000 he has been a visiting scholar in the Computer Science Department, UCLA. His research interests are in Computer Vision. His research topics include multi-view stereo, nonlinear stochastic filtering, and geometric partial differential equations for image processing. He is a student member of the IEEE Computer Society.

PAOLO FAVARO received the DIng degree from the University of Padova, Italy in 1999 and the MSc degree in Electrical Engineering from Washington University, Saint Louis in 2002. He is currently working towards his PhD degree in Electrical Engineering at Washington University. His research interests are in computer vision and inverse problems in imaging. He is a student member of the IEEE Society.

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. 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 with the University of California at Los Angeles, where he is currently Associate Professor in the Computer Science Department. His research interests are in computer vision and in nonlinear systems and control theory. Dr. Soatto is the recipient 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.