Machine Learning, 50, 45–71, 2003

EM, MCMC, and Chain Flipping for Structure from Motion with Unknown Correspondence

FRANK DELLAERT frank@cc.gatech.edu
College of Computing, Georgia Institute of Technology, Atlanta GA, USA

STEVEN M. SEITZ seitz@cs.washington.edu
Department of Computer Science and Engineering, University of Washington, Seattle WA, USA

CHARLES E. THORPE cet@cs.cmu.edu
SEBASTIAN THRUN thrun@cs.cmu.edu
School of Computer Science, Carnegie Mellon University, Pittsburgh PA, USA

Editors: Nando de Freitas, Christophe Andrieu, Arnaud Doucet

Abstract. Learning spatial models from sensor data raises the challenging data association problem of relating model parameters to individual measurements. This paper proposes an EM-based algorithm, which solves the model learning and the data association problem in parallel. The algorithm is developed in the context of the the structure from motion problem, which is the problem of estimating a 3D scene model from a collection of image data. To accommodate the spatial constraints in this domain, we compute *virtual measurements* as sufficient statistics to be used in the M-step. We develop an efficient Markov chain Monte Carlo sampling method called *chain flipping*, to calculate these statistics in the E-step. Experimental results show that we can solve hard data association problems when learning models of 3D scenes, and that we can do so efficiently. We conjecture that this approach can be applied to a broad range of model learning problems from sensor data, such as the robot mapping problem.

Keywords: expectation-maximization, Markov chain Monte Carlo, data association, structure from motion, correspondence problem, efficient sampling, computer vision

1. Introduction

This paper addresses the problem of data association when learning models from data. The *data association problem*, also known as the *correspondence problem*, is the problem of relating sensor measurements to parameters in the model that is being learned. This problem arises in a range of disciplines. In clustering, it is the problem of determining which data point belongs to which cluster (McLachlan & Basford, 1988). In mobile robotics, learning a map of the environment creates the problem of determining the correspondence between individual measurements (e.g., the robot sees a door), and the corresponding features in the world (e.g., door number 17) (Leonard, Cox, & Durrant-Whyte, 1992; Shatkay, 1998; Thrun, Fox, & Burgard, 1998a). A similar problem can be found in computer vision, where it is known as *structure from motion* (SFM). SFM seeks to learn a 3D model from a collection of images, which raises the problem of determining the correspondence between features

in the scene and measurements in image space. In all of these problems, learning a model requires a robust solution to the data association problem which, in the general case, is hard to obtain. Because the problem is hard, many existing algorithms make highly restrictive assumptions, such as the availability of unique landmarks in robotics (Borenstein, Everett, & Feng, 1996), or the existence of reliable feature tracking mechanisms in computer vision (Tomasi & Kanade, 1992; Hartley, 1994).

From a statistical point of view, the data association can be phrased as an incomplete data problem (Tanner, 1996), for which a range of methods exists. One popular approach is *expectation maximization* (EM) (Dempster, Laird, & Rubin, 1977), which has been applied with great success to clustering problems and a range of other estimation problems with incomplete data (McLachlan & Krishnan, 1977). The EM algorithm iterates two estimation steps, called expectation (E-step) and maximization (M-step). The E-step estimates a distribution over the incomplete data using a fixed model. The M-step then calculates the model that maximizes the expected log-likelihood computed in the E-step. It has been shown that iterating these basic steps leads to a model that locally maximizes the likelihood (Dempster, Laird, & Rubin, 1977).

Applying EM to learning spatial models is *not* straightforward, as each domain comes with a set of constraints that are often difficult to incorporate. An example is the work on learning a map of the environment for mobile robots in Shatkay and Kaelbling (1997), Shatkay (1998), and in Burgard et al. (1999), and Thrun, Fox, and Burgard (1998a, 1998b). Both teams have proposed extensions of EM that take into account the geometric constraints of robot environments, and the resulting mapping algorithms have shown to scale up to large environments.

This paper proposes an algorithm that applies EM to a new domain: the structure from motion problem in computer vision. In SFM the model that is being learned is the location of all 3D features, along with the camera poses with 6 DOF. In this paper, we make the commonly made assumption that all 3D features are seen in all images (Tomasi & Kanade, 1992; Hartley, 1994). However, we will discuss at the end of this paper how to extend our method to imaging situations with occlusions and spurious measurements. More importantly, we do not assume any prior knowledge on the camera positions or on the correspondence between image measurements and 3D features the feature identities, giving rise to a hard data association problem.

The majority of literature on SFM considers special situations where the data association problem can be solved easily. Some approaches simply assume that data correspondence is known a priori (Ullman, 1979; Longuet-Higgins, 1981; Tsai & Huang, 1984; Hartley, 1994; Morris & Kanade, 1998). Other approaches consider situations where images are recorded in a sequence, so that features can be tracked from frame to frame (Broida & Chellappa, 1991; Tomasi & Kanade, 1992; Szeliski & Kang, 1993; Poelman & Kanade, 1997). Several authors considered the special case of correct but incomplete correspondence, by interpolating occluded features (Tomasi & Kanade, 1992; Jacobs, 1997; Basri, Grove, & Jacobs, 1998), or expanding a minimal correspondence into a complete correspondence (Seitz & Dyer, 1995). However, these approaches require that a non-degenerate set of correct correspondences be provided a priori. Finally, methods based on the robust recovery of epipolar geometry, e.g. using RANSAC (Beardsley, Torr, & Zisserman, 1996; Torr, Fitzgibbon, & Zisserman, 1998) can cope with larger inter-frame displacements and can be very effective in practice.

However, RANSAC depends crucially on the ability to identify a reliable set of initial correspondences, and this becomes more and more difficult with increasing inter-frame motion.

In the most general case, however, images are taken from widely separated viewpoints. This problem has largely been ignored in the SFM literature, due to the difficulty the data association problem, which has been referred to as the most difficult part of structure recovery (Torr, Fitzgibbon, & Zisserman, 1998). Note that this is particularly challenging in 3D: traditional approaches for establishing correspondence between sets of 2D features (Scott & Longuet-Higgins, 1991; Shapiro & Brady, 1992; Gold et al., 1998) are of limited use in this domain, as the projected 3D structure can look very different in each image.

From a statistical estimation point of view, the SFM problem comes with a unique set of properties, which makes the application of EM non-trivial:

1. *Geometric consistency.* The laws of optical projection constrain the space of valid estimates (models, data associations) in a non-trivial way.
2. *Mutual exclusiveness.* Each feature in the real world occurs *at most once* in each individual camera image—this is an important assumption that severely constrains the data association.
3. *Large parameter spaces.* The number of features in computer vision domains is usually large, giving raise to a large number of local minima.

This paper develops an algorithm based on EM that addresses these challenges. The correspondence (data association) is encoded by an *assignment vector* that assigns individual measurements to specific features in the model. The basic steps of EM are modified to suit the specifics of SFM:

The E-step calculates a posterior over the space of all possible assignments. Unfortunately, the constraints listed above make it impossible to calculate the posterior in closed form. The standard approach for posterior estimation in such situations is Markov chain Monte Carlo (MCMC) (Doucet, de Freitas, & Gordon, 2001; Gilks, Richardson, & Spiegelhalter, 1996; Neal, 1993). In particular, our approach uses the popular Metropolis-Hastings algorithm (Hastings, 1970; Smith & Gelfand, 1992), for approximating the desired posterior summaries. However, the design of efficient Metropolis-Hastings algorithms can be very difficult in high-dimensional spaces (Gilks, Richardson, & Spiegelhalter, 1996). In this paper, we propose a novel, efficient proposal strategy called *chain flipping*, which can quickly jump between globally different assignments. Experimental results show that this approach is much more efficient than approaches that consider only local changes in the MCMC sampling process.

The M-step calculates the location of the features in the scene, along with the camera positions. As pointed out, the SFM literature has developed a number of excellent algorithms for solving this problem under the assumption that the data association problem is solved. However, the E-step generates only probabilistic data associations. To bridge this gap, we introduce the notion of *virtual measurements*. Virtual measurements are generated in the E-step, and have two pleasing properties: first, they make it possible to apply off-the-shelf SFM algorithms for learning the model and the camera positions, and second, they are *sufficient statistics* of the posterior with respect to the problem of learning the model; hence the

M-step is mathematically sound. Independently from us, the concept of virtual measurements had already been used in the tracking literature (Avitzour, 1992; Streit & Luginbuhl, 1994).

From a machine learning point of view, our approach extends EM to an important domain with a set of characteristics for which we previously lacked a sound statistical estimator. From a SFM point of view, our approach adds a method for data association that is statistically sound. Our approach is orthogonal to the vast majority of work on SFM in that it can be combined with virtually any algorithm that assumes known data association. Thus, our approach adds the benefit of solving the data association problem for a large body of literature that previously operated under more narrow assumptions.

2. EM for structure from motion without correspondence

Below we introduce the structure from motion problem and the assumptions we make, and discuss methods to find a maximum-likelihood model for *known* correspondence. We then show how the EM algorithm can be used to learn the model parameters for the case of *unknown* correspondence.

2.1. Problem statement, notation, and assumptions

The SFM problem is this: given a set of images of a scene, learn a model of the 3D scene and recover the camera poses. Several flavors of this problem exist, depending on (a) whether the algorithm works with raw pixel values, or whether a set of discrete measurements is first extracted, (b) whether the images were taken in a continuous sequence or from arbitrary separate locations, or (c) whether the camera's intrinsic parameters are varying or not. In this paper we make the following assumptions:

1. We adopt a feature-based approach, i.e., we assume that the input to the algorithm is a set of discrete *image measurements* $\mathbf{U} = \{\mathbf{u}_{ik} \mid i \in 1..m,\ k \in 1..K_i\}$, where i is the image index. It is assumed that the $\mathbf{u}_{ik}$ correspond to the projection of a set of real world, *3D features* $\mathbf{X} = \{\mathbf{x}_j \mid j \in 1..n\}$, corrupted by additive noise.
2. It is *not* required that the correspondence between measurements in the different images is known. This is exactly the data association problem. To model the correspondence between measurements $\mathbf{u}_{ik}$ and 3D features $\mathbf{x}_j$ we introduce an *assignment vector* $\mathbf{J}$: for each measurement $\mathbf{u}_{ik}$ the vector $\mathbf{J}$ contains an indicator variable j_{ik}, indicating that $\mathbf{u}_{ik}$ is a measurement assigned to the j_{ik}-th feature $\mathbf{x}_{j_{ik}}$. Note that this additional data is unknown or *hidden*.
3. We allow images to be taken from a set of arbitrary *camera poses* $\mathbf{M} = \{\mathbf{m}_i \mid i \in 1..m\}$. This makes the data association problem harder: most existing approaches rely on the temporal continuity of an image stream to *track* features over time (Deriche & Faugeras, 1990; Tomasi & Kanade, 1992; Zhang & Faugeras, 1992; Cox, 1993), or otherwise constrain the data association problem (Beardsley, Torr, & Zisserman, 1996).
4. In this paper, we adopt the commonly used assumption that all features $\mathbf{x}_j$ are seen in all images (Tomasi & Kanade, 1992; Hartley, 1994), i.e. there are no spurious measurements and there is no occlusion. This is a strong assumption: we discuss at the end of this paper

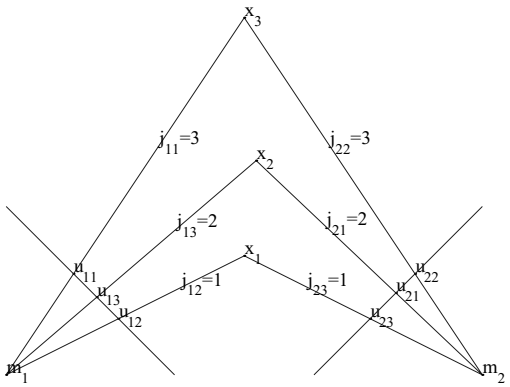


Figure 1. An example with 3 features seen in 2 images. The 6 measurements $\mathbf{u}_{ik}$ are assigned to the individual features $\mathbf{x}_j$ by means of the assignment variables j_{ik}.

how to extend our method to more general imaging situations. Note that this implies that there are exactly n measurements in each image, i.e. $K_i = n$ for all i.

The various variables introduced above are illustrated in figure 1.

2.2. *SFM with known correspondence*

In the case that the assignment vector $\mathbf{J}$ *is* known, i.e., the data association is known, most existing approaches to SFM can be viewed as *maximum likelihood* (ML) methods. The model parameters Θ consist of the 3D feature locations $\mathbf{X}$ and the camera poses $\mathbf{M}$, i.e., $\Theta = (\mathbf{X}, \mathbf{M})$, the *structure* and the *motion*. The data consists of the 2D image measurements $\mathbf{U}$, and the assignment vector $\mathbf{J}$ that assigns measurements $\mathbf{u}_{ik}$ to 3D features $\mathbf{x}_{j_{ik}}$. The *maximum likelihood estimate* Θ^* given the data $\mathbf{U}$ and $\mathbf{J}$ is then given by

$$\Theta^* = \underset{\Theta}{\operatorname{argmax}} \log L(\Theta; \mathbf{U}, \mathbf{J}) \tag{1}$$

where the likelihood $L(\Theta; \mathbf{U}, \mathbf{J})$ is proportional to $P(\mathbf{U}, \mathbf{J} \mid \Theta)$, the conditional density of the data given the model. To evaluate the likelihood, we assume that each measurement $\mathbf{u}_{ik}$ is generated by applying the *measurement function* $\mathbf{h}$ to the model, then corrupted by additive noise $\mathbf{n}$:

$$\mathbf{u}_{ik} = \mathbf{h}(\mathbf{m}_i, \mathbf{x}_{j_{ik}}) + \mathbf{n}$$

A measurement $\mathbf{u}_{ik}$ depends only on the parameters $\mathbf{m}_i$ for the image in which it was observed, and on the 3D feature $\mathbf{x}_{j_{ik}}$ to which it is assigned.

Without loss of generality, let us consider the case in which the features $\mathbf{x}_j$ are 3D points and the measurements $\mathbf{u}_{ik}$ are points in the 2D image. In this case $\mathbf{h}$ can be written as a 3D rigid displacement followed by a projection:

$$\mathbf{h}(\mathbf{m}_i, \mathbf{x}_j) = \Pi_i[\mathbf{R}_i(\mathbf{x}_j - \mathbf{t}_i)] \tag{2}$$

where $\mathbf{R}_i$ and $\mathbf{t}_i$ are the rotation matrix and translation of the i-th camera, respectively, and $\Pi_i : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ is a projection operator which projects a 3D point to the 2D image plane. Various camera models can be defined by specifying the action of this projection operator on a point $\mathbf{x} = (x, y, z)^T$ (Morris, Kanatani, & Kanade, 1999). For example, the projection operators for orthography and calibrated perspective are defined as:

$$\Pi_i^o[\mathbf{x}] = \begin{pmatrix} x \\ y \end{pmatrix}, \quad \Pi_i^p[\mathbf{x}] = \begin{pmatrix} x/z \\ y/z \end{pmatrix}$$

Finally, we need to assume a distribution for the noise $\mathbf{n}$. In the case that $\mathbf{n}$ is i.i.d. zero-mean Gaussian noise with standard deviation σ, the negative log-likelihood is simply a sum of squared reprojection errors:

$$\log L(\Theta; \mathbf{U}, \mathbf{J}) = -\frac{1}{2\sigma^2} \sum_{i=1}^{m} \sum_{k=1}^{n} \left\| \mathbf{u}_{ik} - \mathbf{h}\big(\mathbf{m}_i, \mathbf{x}_{j_{ik}}\big) \right\|^2 \tag{3}$$

The more realistic model for automatic feature detectors, where each measurement can have its own individual covariance matrix $\mathbf{R}_{ik}$, can be accommodated with obvious modifications.

2.3. *Existing methods for structure from motion*

The structure from motion problem has been studied extensively in the computer vision literature over the past three decades. A good survey of techniques can be found in Hartley and Zisserman (2000).

The earliest work focused on reconstruction from two images only (Ullman, 1979; Longuet-Higgins, 1981). Later new methods were developed to handle multiple images, and they can all viewed as minimizing an objective function such as (3), under a variety of different assumptions:

In the case of orthographic projection the maximum likelihood model Θ^* can be found efficiently using a *factorization* approach (Tomasi & Kanade, 1992). Here singular value decomposition (SVD) is first applied to the data $\mathbf{U}$ in order to obtain *affine* structure $\mathbf{X}^a$ and motion $\mathbf{M}^a$. Euclidean structure and motion is then obtained after an additional step that imposes metric constraints on $\mathbf{M}^a$. The factorization method is fast and does not require a good initial estimate to converge. It has been applied to more complex camera models, i.e., weak- and para-perspective models (Poelman & Kanade, 1997), and even to fully perspective cameras (Triggs, 1996). The reader is referred to Tomasi and Kanade (1992), Poelman and Kanade (1997), and Morris and Kanade (1998) for details and additional references.

In the case of full perspective cameras the measurement function $\mathbf{h}(\mathbf{m}_i, \mathbf{x}_j)$ is non-linear, and one needs to resort to non-linear optimization to minimize the re-projection error (3). This procedure is known in photogrammetry and computer vision as *bundle adjustment* (Spetsakis & Aloimonos, 1991; Szeliski & Kang, 1993; Hartley, 1994; Triggs et al., 1999). The advantage with respect to factorization is that it gives the exact ML estimate, if it converges. However, it can easily get stuck in local minima, and thus a good initial estimate

of the solution needs to be available. To obtain this, recursive estimation techniques can be used to process the images as they arrive (Broida & Chellappa, 1991).

2.4. SFM without correspondences

In the case that the correspondences are unknown we cannot directly apply the methods discussed in Section 2.3. Although we can still frame this case as a problem of maximum likelihood estimation, solving it directly is intractable due to the combinatorial nature of the data association problem. By total probability the maximum likelihood estimate $\Theta^* = (\mathbf{X}^*, \mathbf{M}^*)$ of structure and motion given *only* the measurements $\mathbf{U}$ is given by

$$\Theta^* = \operatorname*{argmax}_{\Theta} \log L(\Theta; \mathbf{U}) = \operatorname*{argmax}_{\Theta} \log \sum_{\mathbf{J}} L(\Theta; \mathbf{U}, \mathbf{J}) \tag{4}$$

a sum of likelihood terms of the form (1), with one term for *every possible* assignment vector $\mathbf{J}$. Now, for any realistic number of features n and number of images m, the number of assignments explodes combinatorially. There are $n!$ possible assignment vectors $\mathbf{J}_i$ in each image, yielding a total of $n!^m$ assignments. In summary, $L(\Theta; \mathbf{U})$ is in general hard to obtain explicitly, as it involves summing over a combinatorial number of possible assignments.

2.5. The expectation maximization algorithm

A key insight is that we can use the well-known EM algorithm (Hartley, 1958; Dempster, Laird, & Rubin, 1977; McLachlan & Krishnan, 1977) to attack the data association problem that arises in the context of structure from motion. While a direct approach to computing the total likelihood $L(\Theta; \mathbf{U})$ in (4) is generally intractable, EM provides a practical method for finding its maxima. The EM algorithm starts from an initial guess Θ^0 for structure and motion, and then iterates over the following steps:

1. *E-step*: Calculate the expected log likelihood function $Q^t(\Theta)$:

$$Q^t(\Theta) = \sum_{\mathbf{J}} f^t(\mathbf{J}) \log L(\Theta; \mathbf{U}, \mathbf{J}) \tag{5}$$

   where the expectation is taken with respect to the posterior distribution $f^t(\mathbf{J}) \triangleq P(\mathbf{J} \mid \mathbf{U}, \Theta^t)$ over all possible assignments $\mathbf{J}$ given the data $\mathbf{U}$ and a current guess Θ^t for structure and motion.
2. *M-step*: Find the ML estimate Θ^{t+1} for structure and motion, by maximizing $Q^t(\Theta)$:

$$\Theta^{t+1} = \operatorname*{argmax}_{\Theta} Q^t(\Theta)$$

It is important to note that $Q^t(\Theta)$ is calculated in the E-step by evaluating $f^t(\mathbf{J})$ using the *current guess* Θ^t for structure and motion (hence the superscript t), whereas in the M-step we are optimizing $Q^t(\Theta)$ with respect to the *free variable* Θ to obtain the new estimate

Θ^{t+1}. It can be proven that the EM algorithm converges to a local maximum of $L(\Theta; \mathbf{U})$ (Dempster, Laird, & Rubin, 1977; McLachlan & Krishnan, 1977).

3. The M-step and virtual measurements

In this section we show that the M-step for structure from motion can be implemented in a simple and intuitive way. We show that the expected log-likelihood can be rewritten *such that the M-step amounts to solving a structure from motion problem of the same size as before*, but using as input a newly synthesized set of "virtual measurements", created in the E-step. The concept of using synthetic measurements is not new. It is also used in the tracking literature, where EM is used to perform track smoothing (Avitzour, 1992; Streit & Luginbuhl, 1994).

We first rewrite the expected log-likelihood $Q^t(\Theta)$ in terms of sum of squared errors, which we can do under the assumption of i.i.d. Gaussian noise. By substituting the expression for the log likelihood $\log L(\Theta; \mathbf{U}, \mathbf{J})$ from (3) in Eq. (5), we obtain:

$$Q^t(\Theta) = -\frac{1}{2\sigma^2} \sum_{\mathbf{J}} f^t(\mathbf{J}) \sum_{i=1}^{m} \sum_{k=1}^{n} \left\| \mathbf{u}_{ik} - \mathbf{h}\left(\mathbf{m}_i, \mathbf{x}_{j_{ik}}\right) \right\|^2 \tag{6}$$

The key to the efficiency of EM lies in the fact that the expression above contains many repeated terms, and can be rewritten as

$$Q^t(\Theta) = -\frac{1}{2\sigma^2} \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{n} f_{ijk}^t \|\mathbf{u}_{ik} - \mathbf{h}(\mathbf{m}_i, \mathbf{x}_j)\|^2 \tag{7}$$

where f_{ijk}^t is the *marginal posterior probability* $P(j_{ik} = j \mid \mathbf{U}, \Theta^t)$. Note that this does not depend on the assumption of Gaussian noise, but rather on the conditional independence of image measurements. The marginal probabilities f_{ijk}^t can be calculated by summing $f^t(\mathbf{J})$ over all possible assignments $\mathbf{J}$ where $j_{ik} = j$ (with $\delta(.,.)$ the Kronecker delta function):

$$f_{ijk}^t \triangleq P(j_{ik} = j \mid \mathbf{U}, \Theta^t) = \sum_{\mathbf{J}} \delta(j_{ik}, j) f^t(\mathbf{J}) \tag{8}$$

The main point to be made in this section is this: it can be shown by simple algebraic manipulation that (7) can be written as the sum of a constant that does not depend on Θ, and a new re-projection error of n features in m images

$$Q^t(\Theta) = C - \frac{1}{2\sigma^2} \sum_{i=1}^{m} \sum_{j=1}^{n} \left\| \mathbf{v}_{ij}^t - \mathbf{h}(\mathbf{m}_i, \mathbf{x}_j) \right\|^2 \tag{9}$$

where the *virtual measurements* $\mathbf{v}_{ij}^t$ are defined simply as weighted averages of the original measurements $\mathbf{u}_{ik}$:

$$\mathbf{v}_{ij}^t \triangleq \sum_{k=1}^{n} f_{ijk}^t \mathbf{u}_{ik} \tag{10}$$

The important point is that the M-step objective function (9) above, arrived at by assuming *unknown* correspondence, is of exactly the same form as the objective function (3) for the SFM problem with *known* correspondence. As a consequence, *any of the existing SFM methods can be used to implement the M-step.* This provides an intuitive interpretation for the overall algorithm:

1. *E-step*: Calculate the weights f_{ijk}^t from the distribution over assignments. Then, in each of the m images calculate n virtual measurements $\mathbf{v}_{ij}^t$.
2. *M-step*: Solve a conventional SFM problem using the virtual measurements as input.

In other words, the E-step synthesizes new measurement data, and the M-step is implemented using conventional SFM methods.

4. Markov chain Monte Carlo and the E-step

The previous section showed that, when given the virtual measurements, the M-step can be implemented using any known SFM approach. As a consequence, we need only concern ourselves with the implementation of the E-step. In particular, we need to calculate the marginal probabilities $f_{ijk}^t = P(j_{ik} = j \mid \mathbf{U}, \Theta^t)$ needed to calculate the virtual measurements $\mathbf{v}_{ij}^t$.

Unfortunately, due to the *mutual exclusion* constraint an analytic expression for the sufficient statistics f_{ijk}^t is hard to obtain. Assuming conditional independence of the assignments $\mathbf{J}_i$ in each image, we can factor $f^t(\mathbf{J})$ as:

$$f^t(\mathbf{J}) \triangleq P(\mathbf{J} \mid \mathbf{U}, \Theta^t) = \prod_{i=1}^{m} P(\mathbf{J}_i \mid \mathbf{U}_i, \Theta^t)$$

where $\mathbf{U}_i$ are the measurements in image i. Applying Bayes law, we have

$$P(\mathbf{J}_i \mid \mathbf{U}_i, \Theta^t) \propto P(\mathbf{J}_i \mid \Theta^t) \exp\left[-\frac{1}{2\sigma^2} \sum_{k=1}^{n} \left\| \mathbf{u}_{ik} - \mathbf{h}\left(\mathbf{m}_i^t, \mathbf{x}_j^t\right) \right\|^2\right] \tag{11}$$

The second part of this expression is simple enough. However, the prior probability $P(\mathbf{J}_i \mid \Theta^t)$ of an assignment $\mathbf{J}_i$ encodes the knowledge we have about the structure from motion domain: if a measurement $\mathbf{u}_{ik}$ has been assigned $j_{ik} = j$, then no other measurement in the same image should be assigned the same feature point $\mathbf{x}_j$. While it is easy to evaluate the posterior probability $f_i^t(\mathbf{J}_i)$ for any *given assi*gnment $\mathbf{J}_i$ through (11), a closed form expression for f_{ijk}^t that incorporates this mutual exclusion constraint is not available.

4.1. *Sampling the distribution over assignments* $\mathbf{J}_i$

The solution we employ is to instead *sample* from the posterior probability distribution $f_i^t(\mathbf{J}_i)$ over valid assignments vectors $\mathbf{J}_i$. Formally this can be justified in the context of a

Monte Carlo EM or MCEM, a version of EM where the E-step is executed by a Monte-Carlo process (Tanner, 1996).

To sample from $f_i^t(\mathbf{J}_i)$ we use a Markov chain Monte Carlo (MCMC) sampling method (Neal, 1993; Gilks, Richardson, & Spiegelhalter, 1996; Doucet, de Freitas, & Gordon, 2001). MCMC methods can be used to obtain approximate values for expectations over distributions that defy easy analytical solutions. All MCMC methods work the same way: they generate a sequence of *states*, in our case the assignments vectors $\mathbf{J}_i$ in image i, with the property that the collection of generated assignments $\mathbf{J}_i^r$ approximates a sample from a target distribution, in our case the posterior distribution $f_i^t(\mathbf{J}_i)$. To accomplish this, a Markov chain is defined over the space of assignments $\mathbf{J}_i$, i.e. a transition probability matrix is specified that gives the probability of transitioning from any given assignment $\mathbf{J}_i$ to any other. The transition probabilities are set up in a very specific way, however, such that the stationary distribution of the Markov chain is exactly the target distribution $f_i^t(\mathbf{J}_i)$. This guarantees that, if we run the chain for a sufficiently long time and then start recording states, these states constitute a (correlated) sample from the target distribution.

The Metropolis-Hastings (MH) algorithm (Hastings, 1970) is one way to simulate a Markov chain with the correct stationary distribution, without explicitly building the full transition probability matrix (which would be intractable). In our case, we use it to generate a sequence of R samples $\mathbf{J}_i^r$ from the posterior $f_i^t(\mathbf{J}_i)$. The pseudo-code for the MH algorithm is as follows (adapted from Gilks, Richardson, & Spiegelhalter, 1996):

1. Start with a valid initial assignment $\mathbf{J}_i^0$.
2. Propose a new assignment $\mathbf{J}_i'$ using the *proposal density* $Q(\mathbf{J}_i'; \mathbf{J}_i^r)$
3. Calculate the *acceptance ratio*

$$a = \frac{f_i^t(\mathbf{J}_i')}{f_i^t(\mathbf{J}_i^r)} \frac{Q(\mathbf{J}_i^r; \mathbf{J}_i')}{Q(\mathbf{J}_i'; \mathbf{J}_i^r)} \tag{12}$$

4. **If** $a \geq 1$ then accept $\mathbf{J}_i'$, i.e. we set $\mathbf{J}_i^{r+1} \leftarrow \mathbf{J}_i'$.
   Otherwise, accept $\mathbf{J}_i'$ with probability a. If the proposal is rejected, then we keep the previous sample, i.e. we set $\mathbf{J}_i^{r+1} = \mathbf{J}_i^r$.

Intuitively, step 2 proposes "moves" in state space, generated according to a probability distribution $Q(\mathbf{J}_i'; \mathbf{J}_i^r)$ which is fixed in time but can depend on the current state $\mathbf{J}_i^r$. The calculation of a and the acceptance mechanism in steps 3 and 4 have the effect of modifying the transition probabilities of the chain such that its stationary distribution is exactly f_i^t.

The MH algorithm easily allows incorporating the mutual exclusion constraint: if an an assignment $\mathbf{J}_i'$ is proposed that violates the constraint, the acceptance ratio is simply 0, and the move is not accepted. Alternatively, and this is more efficient, one could take care never to propose such a move.

To compute the virtual measurements in (10), we need to compute the marginal probabilities f_{ijk}^t from the sample $\{\mathbf{J}_i^r\}$. Fortunately, this can be done without explicitly storing the samples, by keeping running counts C_{ijk} of how many times each measurement $\mathbf{u}_{ik}$ is

assigned to feature j, as

$$f_{ijk}^t \approx \frac{1}{R} C_{ijk} \stackrel{\Delta}{=} \frac{1}{R} \sum_{r=1}^{R} \delta\left(j_{ik}^r, j\right) \tag{13}$$

is easily seen to be the Monte Carlo approximation to (8).

Finally, in order to implement the sampler, we need to know how to propose new assignments $\mathbf{J}_i'$, i.e. the proposal density $Q(\mathbf{J}_i^r; \mathbf{J}_i')$, and how to compute the ratio a. Both elements are discussed in detail in Section 7.

5. The algorithm in practice

The pseudo-code for the final algorithm is as follows:

1. Generate an initial structure and motion estimate Θ^0.
2. Given Θ^t and the data $\mathbf{U}$, run the Metropolis-Hastings sampler in each image to obtain approximate values for the weights f_{ijk}^t (Eq. 13).
3. Calculate the virtual measurements $\mathbf{v}_{ij}^t$ using Eq. (10).
4. Find the new estimate Θ^{t+1} for structure and motion using the virtual measurements $\mathbf{v}_{ij}^t$ as data. This can be done using any SFM method discussed in Section 2.3.
5. If not converged, return to step 2.

One significant disadvantage of EM is that is only guaranteed to converge to a *local* maximum of the likelihood function, not to a global maximum. This is especially problematic in the current application, where bad initial estimates for structure and motion can be locked in by incorrect correspondences, and vice versa. In order to avoid this, we employ three different strategies:

1. Gross to fine structure via *annealing*. A well known technique to avoid local mimima is annealing: here we increase the noise parameter σ in early iterations, gradually decreasing it to its correct value. This has two beneficial consequences. First, the posterior distribution $f_i^t(\mathbf{J}_i)$ is less peaked when σ is high, allowing the MCMC sampler to explore the space of assignments $\mathbf{J}_i$ more easily. Second, the expected log-likelihood $Q^t(\Theta)$ is smoother and has fewer local maxima for higher values of σ.
2. Minimizing the influence of local mismatches via *robust optimization*. A typical failure mode of the algorithm in the final stages is due to local mismatches, where measurements generated by two features are correctly assigned in most of the images, but switched in some. Because of the quadratic error function this can severly bias the motion recovery, which in turn locks the incorrect correspondence into place. Fortunately, we found that this can be overcome by employing a robust optimization algorithm in the M-step, e.g. the robust factorization method decsribed in Kurata et al. (1999). Note that the EM mechanism is still crucial to recovering the gross structure and refining the solution: the robust optimization only helps in discarding local mismatches in the final stages. At that

point, the distributions computed in the E-step become really sharp, and can get locked into local minima more easily.
3. Random restarts. It is easy to detect when a local minimum is reached based on the expected value of the residual. If this occurs, the algorithm is restarted with different initial conditions, until eventually succesful.

The combination of these strategies leads to good results in many cases. A more detailed analysis is presented in the section below.

6. Results for SFM without correspondence

Below we show results on two sets of images for which the SFM problem is non-trivial in the absence of correspondence. Many more examples can be found in Dellaert (2001). The input to the algorithm is always a set of manually obtained image measurements. To initialize, the 3D points $\mathbf{x}_j$ were generated randomly in a normally distributed cloud around a depth of 1, whereas the cameras $\mathbf{m}_i$ were all initialized at the origin. In each case, we ran the EM algorithm for 100 iterations, with the annealing parameter σ decreasing linearly from 40 pixels to 1 pixel. For each EM iteration, we ran the sampler in each image for 1000 steps per point. An entire run (of 100 EM iterations) takes on the order of a minute of CPU time on a standard PC.

In practice, the algorithm converges consistently and quickly to an estimate for the structure and motion where the correct assignment is the most probable one, and where all assignments in the different images agree with each other. We illustrate this using the image set shown in figure 2, which was taken under orthographic projection. The typical evolution of the algorithm is illustrated in figure 3, where we have shown a wire-frame model of the recovered structure at successive instants of time. There are two important points to note: (a) *the gross structure is recovered in the very first iteration, starting from random initial structure*, and (b) finer details of the structure are gradually resolved as the parameter σ is decreased. The estimate for the structure after convergence is almost identical to the one found by factorization when given the correct assignment.

To illustrate the EM iterations, consider the set of images in figure 4 taken under perspective projection. In the perspective case, we implement the M-step as para-perspective factorization followed by bundle adjustment. In this example we do not show the recovered structure (which is good), but show the marginal probabilities f_{ijk}^t at three different times

Figure 2. Three out of 11 cube images. Although the images were originally taken as a sequence in time, the ordering of the images is irrelevant to our method.

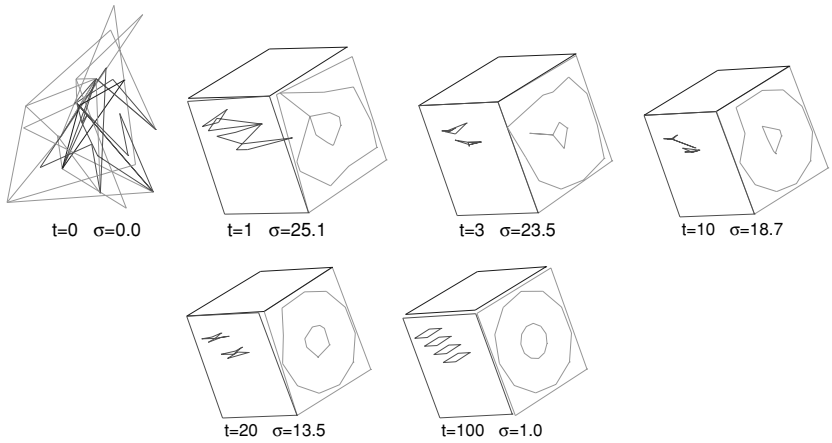


Figure 3. The structure estimate as initialized and at successive iterations t of the algorithm.

Figure 4. 4 out of 5 perspective images of a house.

during the course of the algorithm, in figure 5. In early iterations, σ is high and there is still a lot of ambiguity. Towards the end, the distribution focuses in on one consistent assignment. In the last iteration the marginal probabilities are all consistent with the ground-truth assignment, and the features in the figure are ordered such that this corresponds to a set of 5 stacked identity matrices. The off-diagonal marginals are not exactly zero, simply very close to zero.

Finally, in order to investigate the behavior of the algorithm in terms of local maxima, we have conducted a series of experiments with synthetic data. For different settings of m and n, 5 scenes were generated randomly. The n points were randomly generated on a 2D square

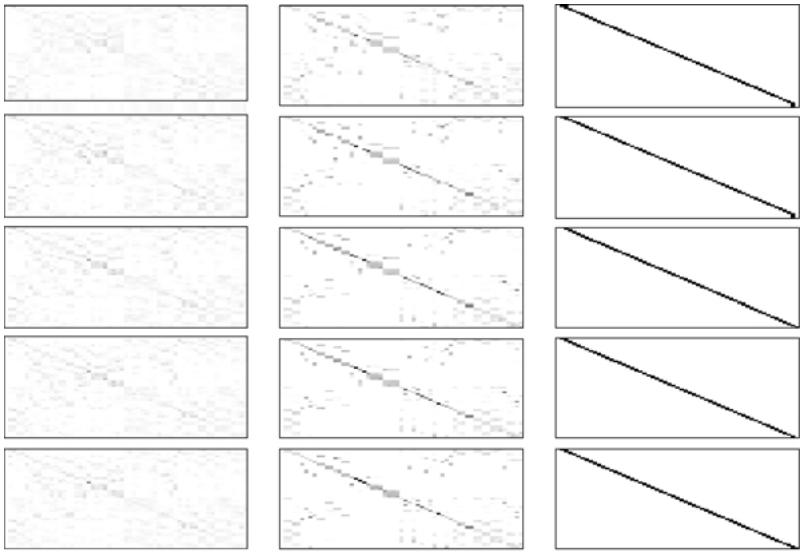


Figure 5. The marginal probabilities f_{ijk}^t at three different iterations, respectively 1, 10, and 100. Each row corresponds to a measurement $\mathbf{u}_{ik}$, grouped according to image index, whereas the columns represent the n features $\mathbf{x}_j$. In this example $n = 58$ and $m = 5$. Black corresponds to a marginal probability of 1. Note that in the final iteration, the correspondence is near-perfect.

with side $S = 4$, then were displaced from the plane according to a normal distribution with standard deviation of $\sigma_d = 0.1$, yielding a random 'plane plus parallax scene. m cameras where then placed randomly on a sphere segment spanning an arc $\alpha = \pi/2$, at a distance of $R = 5$. The measurement model was orthographic, with measurement noise drawn form a Gaussian with $\sigma = 0.005$. For each setting of m and n, 5 scenes were generated in this manner, and the EM algorithm was run 5 times for each scene. The number of times EM converged is summarized in figure 6. For $n = 20$ the majority of trials converged, for all values of m. For more points the possibility of confusing measurements grows, and in three cases (out of 20) none of the 5 trials converged to the global maximum. However, even in these cases the percentage of incorrectly assigned measurements was small, indicating that some local mismatches remained that were unresolved by the robust factorization scheme. This is to be expected as more and more points are introduced in the scene.

7. An efficient sampler

The EM approach for structure and motion without correspondence outlined in the previous sections is a statistically sound way to deal with a difficult data association problem. However, in order for it to scale up to larger problems, it is imperative that it is also *efficient*. In this section we show that the Metropolis-Hastings method can be made to very effectively sample from weighted assignments, yielding an efficient E-step implementation.

scene:	A	B	C	D	E
m=10 n=20	4	5	5	4	5
m=15 n=20	4	5	5	5	5
m=20 n=20	4	5	4	5	3
m=25 n=20	5	4	4	5	5
m=30 n=20	5	4	5	5	5

scene:	A	B	C	D	E
m=10 n=20	4	5	5	4	5
m=10 n=40	3	5	4	4	5
m=10 n=60	4	4	(0.7)	4	3
m=10 n=80	(0.3)	4	3	(1.3)	2

Figure 6. EM was run 5 times for each of 5 randomly generated examples A–E for each different setting of m and n. The tables show the number of times EM converged to the global maximum, out of 5 trials. If none of the trials converged, the percentage of incorrectly assigned measurements is shown in brackets.

The convergence of the Metropolis-Hastings algorithm depends crucially on the proposal density Q. We need a proposal strategy that leads to a rapidly mixing Markov chain, i.e. one that converges quickly to the stationary distribution. Below we discuss three different proposal strategies, each of which induces a Markov chain with increasingly better convergence properties.

7.1. Preliminaries

It is convenient at this time to look at the sampling in each image in isolation, and think of it in terms of *weighted bipartite graph matching.* Consider the bipartite graph $G = (U, V, E)$ in image i where the vertices U correspond to the image measurements, i.e. $u_k \triangleq \mathbf{u}_{ik}$, and the vertices V are identified with the projected features, given the current guess Θ^t for structure and motion, i.e. $v_j \triangleq \mathbf{h}(\mathbf{m}_i^t, \mathbf{x}_j^t)$. Both k and j range from 1 to n, i.e. $|U| = |V| = n$. Finally, the graph is fully connected $E = U \times V$, and we associate the following *edge weight* with each edge $e = (u_k, v_j)$:

$$w(u_k, v_j) \triangleq \frac{1}{2\sigma^2}\|u_k - v_j\|^2 = \frac{1}{2\sigma^2}\left\|\mathbf{u}_{ik} - \mathbf{h}\left(\mathbf{m}_i^t, \mathbf{x}_j^t\right)\right\|^2$$

A *matching* is defined as a subset M of the edges E, such that each vertex is incident to at most one edge. An *assignment* is defined as a perfect matching: a set of n edges such that every vertex is incident to exactly one edge.

Given these definitions, it is easily seen that every assignment vector $\mathbf{J}_i$ corresponds to an assignment in the bipartite graph G, so we use the same symbol to denote both entities. Furthermore, we use the notation $\mathbf{J}_i(u)$ to denote the match of a vertex u, i.e. $\mathbf{J}_i(u_k) = v_j$ iff $j_{ik} = j$. Recalling Eq. (11), it is easily seen that *for valid assignments* $\mathbf{J}_i$, the posterior

probability $f_i^t(\mathbf{J}_i)$ can be expressed in terms of the edge weights as follows:

$$f_i^t(\mathbf{J}_i) \propto \exp\left[-\frac{1}{2\sigma^2}\sum_{k=1}^{n}\left\|\mathbf{u}_{ik} - \mathbf{h}\left(\mathbf{m}_i^t, \mathbf{x}_j^t\right)\right\|^2\right] \propto e^{-w(\mathbf{J}_i)} \tag{14}$$

where the *weight* $w(\mathbf{J}_i)$ of an assignment is defined as

$$w(\mathbf{J}_i) = \sum_{k=1}^{n} w(u_k, \mathbf{J}_i(u_k))$$

Expression (14) has the form of a Gibbs distribution, where $w(\mathbf{J}_i)$ plays the role of an energy term: assignments with higher weight (energy) are less likely, assignments with lower weight (energy) are more likely.

Thus, the problem of sampling from the assignment vectors $\mathbf{J}_i$ *in the structure and motion problem is equivalent to sampling from weighted assignments in the bipartite graph G, where the target distribution is given by the Gibbs distribution* (14). Below we drop the image index i, and think solely in terms of the weighted assignment problem.

7.2. Flip proposals

The simplest way to propose a new assignment J' from a current assignment J is simply to swap the assignment of two randomly chosen vertices u:

1. Pick two matched edges (u_1, v_1) and (u_2, v_2) at random.
2. Swap their assignments, i.e. set $J'(u_1) \leftarrow v_2$ and $J'(u_2) \leftarrow v_1$

To calculate the ratio a, note that the proposal ratio $\frac{Q(J;J')}{Q(J';J)} = 1$. Thus, the acceptance ratio a is equal to the probability ratio, given by

$$a = \frac{P(J')}{P(J)} = \exp[w(u_1, v_1) + w(u_2, v_2) - w(u_1, v_2) - w(u_2, v_1)]$$

Even though this "flip proposal" strategy is attractive from a computational point of view, it has the severe disadvantage of leading to slowly mixing chains in many instances. To see this, consider the arrangement with $n = 3$ in figure 7. There is no way to move from the most likely configurations (a) to (f) via flip proposals without passing through one of the unlikely states (b–e). An MCMC sampler that proposes only such moves can stay stuck in the modes (a) or (f) for a long time.

7.3. Augmenting paths and alternating cycles

In order to improve the convergence properties of the chain, we use the idea of randomly generating an *augmenting path*, a construct that plays a central role in deterministic algorithms to find the optimal weighted assignment (Bertsekas, 1991; Cook et al., 1998; Papadimitriou & Steiglitz, 1982). The intuition behind an augmenting path is simple: it is a way to resolve conflicts when proposing a new assignment for some random vertex in U.

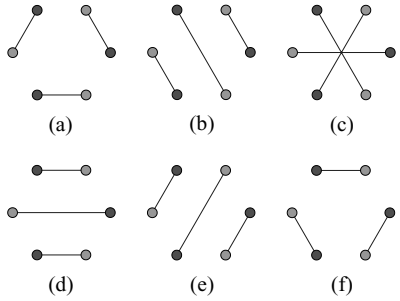


Figure 7. An ambiguous assignment problem with $n = 3$. The regular arrangement of the vertices yields two optimal assignments, (a) and (f), whereas (b–e) are much less likely. The figure illustrates a major problem with "flip proposals": there is no way to move from (a) to (e) via flip proposals without passing through one of the unlikely states (b–e).

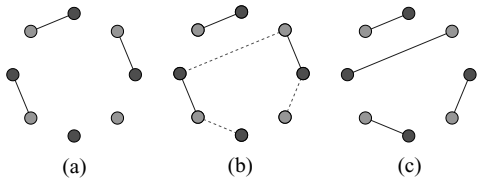


Figure 8. Augmenting paths. (a) Original, partial matching. (b) An augmenting path, alternating between free and matched edges. (c) The resulting matching after augmenting the matching in (a) with the path in (b).

When sampling, an idea for a proposal density is to randomly pick a vertex u and change its assignment, but as this can lead to a conflict, we propose to use a similar mechanism to resolve the conflict recursively.

We now explain augmenting paths following (Kozen, 1991). Assume we have a partial matching M. An example is given in figure 8(a). Now pick an unmatched vertex u, and propose to match it up with v. We indicate this by traversing the free edge (u, v). If v is free, we can simply add this edge to the matching M. However, if v is not free we cancel its current assignment by traversing the *matched* edge (v, u'). We then recurse, until a free vertex in V is reached, tracing out the *augmenting path* p. One such a path is shown in figure 8(b). Now the matching can be *augmented* to M' by swapping the matched and the free edges in p. This *augmentation* operation is written as $M' = M \oplus p$, where $\oplus$ is the symmetric difference operator on sets

$$A \oplus B = (A \cup B) - (A \cap B) = (A - B) \cup (B - A)$$

For the example, the resulting matching is shown in figure 8(c). Algorithms to find optimal matchings start with an empty matching, and then perform a series of augmentations until a maximal matching is obtained.

For sampling purposes alternating *cycles* are of interest, because they implement k-swaps. An example is shown for $n = 4$ in figure 9. In contrast to the optimal algorithms, when sampling we start out with a perfect matching (an assignment), and want to propose a

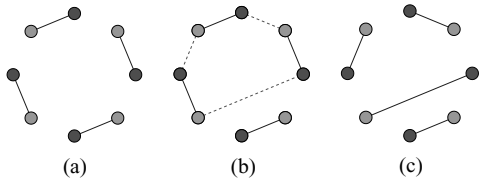


Figure 9. (a) Original assignment. (b) An alternating cycle implementing a k-swap, with $k = 3$ in this example. (c) Newly obtained assignment.

move to a different -also perfect- matching. We can do this by proposing the matching $J' = J \oplus C$, where C is an alternating cycle, which has the effect of permuting a subset of the assignments. Such permutations that leave no element untouched are also called *derangements.*

7.4. Proposing moves by "chain flipping"

Recall that the goal is to sample from assignments **J** using the Metropolis-Hastings algorithm. We now advance a new strategy to generate proposed moves, through an algorithm that we call "chain flipping" (CF). The algorithm is based on randomly generating an alternating cycle according to the following algorithm:

1. Pick a random vertex u in U
2. Choose a match v in V by traversing the edge $e = (u, v)$ according to the transition probabilities

$$q(u, v) \triangleq \frac{\exp(-w(u, v))}{\sum_v \exp(-w(u, v))} \tag{15}$$

   which accords higher probability to edges $e = (u, v)$ with lower weight.
3. Traverse the matched edge (v, u') to undo the former match.
4. Continue with 2 until a cycle is formed.
5. Erase the transient part to get an alternating cycle C.

This algorithm simulates a Markov chain MC defined on the bipartite graph G and terminates the simulation when a cycle is detected. The resulting alternating cycle C is used to propose a new assignment $J' = J \oplus C$, i.e. we "flip" the assignments on the alternating cycle or "chain" of alternating edges.

We also need to calculate the acceptance ratio a. As it happens, we have

$$a_{CF} = \frac{P(J')}{P(J)} \frac{Q(J; J')}{Q(J'; J)} = 1 \tag{16}$$

To prove this, note that by (14) and (15) the probability ratio is given by

$$\frac{P(J')}{P(J)} = \frac{e^{-w(J')}}{e^{-w(J)}} = \prod_{u \in C} \frac{q(u, J'(u))}{q(u, J(u))} \tag{17}$$

The proposal density $Q(J'; J)$ is equal to the probability of proposing a cycle C that yields J' from J, which is given by:

$$Q(J'; J) = \left(\prod_{(u,v)\in p} q(u, J'(u)) \right) \sum_{T} P_{MC}(T) \tag{18}$$

where the sum is over all transient paths T that end on the cycle C, and $P_{MC}(T)$ is the probability of one such transient. The probability $Q(J; J')$ of proposing J starting from J' is similarly obtained, and substituting both together with (17) into (16) yields the surprising result $a = 1$.

A distinct advantage of the CF algorithm is that, as with the Gibbs sampler (Gilks, Richardson, & Spiegelhalter, 1996), every proposed move is always accepted. The n^2 transition probabilities $q(u, v)$ are also fixed and can be easily pre-computed. A major disadvantage, however, is that many of the generated paths do not actually change the current assignment, making the chain slower than it could be. This is because in step 2 there is nothing that prevents us from choosing a matched edge, leading to a trivial cycle, and in steady state matched edges are exactly those with high transition probabilities.

7.5. "Smart chain flipping"

An obvious modification to the CF algorithm, and one that leads to very effective sampling, is to make it impossible to traverse through a matched edge when generating the proposal paths. This ensures that every proposed move does indeed change the assignment, *if* it is accepted. However, now the ratio a can be less than 1, causing some moves to be rejected.

Forcing the chosen edges to be free can be accomplished by modifying the transition probabilities $q(u, v)$. We denote the new transition probabilities as $q^{\mathbf{J}}(u, v)$, as they depend on the current assignment J, and define them as follows:

$$q^{\mathbf{J}}(u, v) \triangleq \begin{cases} \dfrac{\exp(-w(u, v))}{\sum_{v\neq J(u)} \exp(-w(u, v))} & \text{if } v \neq J(u) \\ 0 & \text{if } v = J(u) \end{cases}$$

i.e. we disallow the transition through a matched edge. We can rewrite this in terms of the transition probabilities $q(u, v)$ defined earlier in (15), as follows

$$q^{\mathbf{J}}(u, v) = \begin{cases} \dfrac{q(u, v)}{1 - q(u, J(u))} & \text{if } v \neq J(u) \\ 0 & \text{if } v = J(u) \end{cases}$$

Note that *these depend on the current assignment* J, but in an implementation their explicit calculation can be avoided by appropriately modifying the cumulative distribution function of q at run-time.

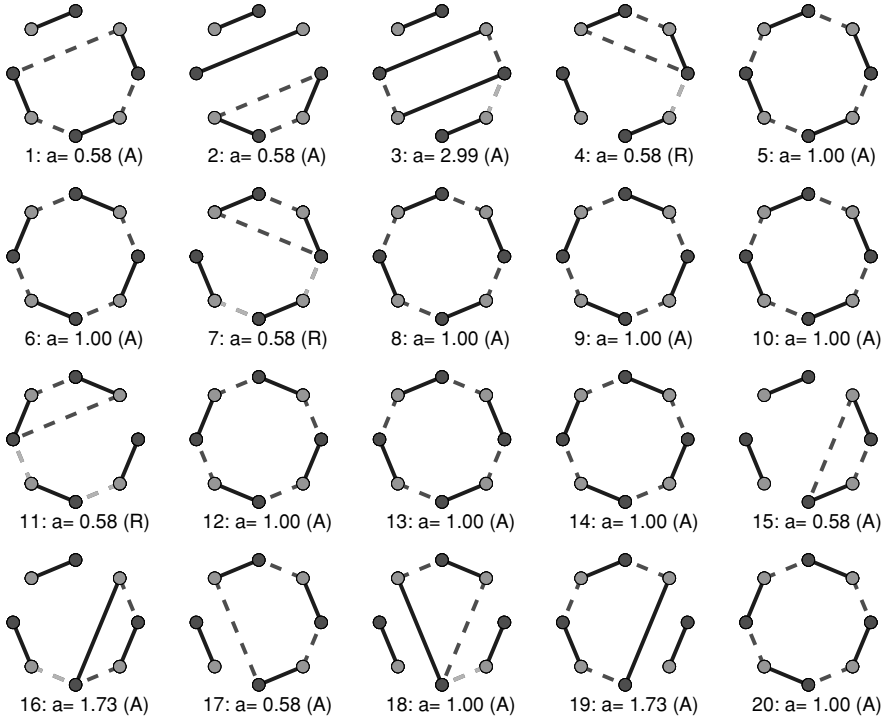


Figure 10. 20 iterations of an MCMC sampler with the "smart chain flipping" proposals. For each iteration we show a and whether the move was accepted (A) or rejected (R).

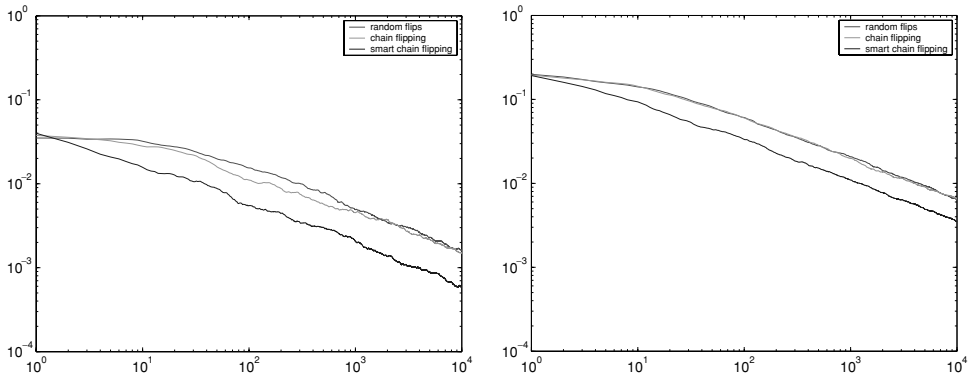
This proposal strategy, which we call "smart chain flipping" (SMART), generates more exploratory moves than the CF algorithm, but at the expense of rejecting some of the moves. It can be easily verified that we now have

$$a_{SMART} = \prod_{u \in C} \frac{1 - q(u, J(u))}{1 - q(u, J'(u))}$$

In figure 10 we have shown 20 iterations of a Metropolis-Hastings sampler using the SMART proposals, and also show the value of a and whether the move was accepted (A) or rejected (R).

8. Results for efficient sampling

Experimental results support the intuition that "smart chain flipping" leads to more rapidly mixing chains. In order to assess the relative performance of the three different samplers we have discussed above, we have generated 1000 synthetic examples with $n = 5$, and



(a) Low Temperature (b) High Temperature

Figure 11. Log-log plot comparing the mean absolute error (y-axis) versus number of samples (x-axis) for the 3 different proposal distributions: random flips, chain flipping, and smart chain flipping. (a) For a 'sharp' distribution with low annealing parameter $\sigma = 0.2$, and (b) for a high value of $\sigma = 0.6$.

ran each sampler for 10000 iterations on each example. There was no need to wait until the stationary distribution was reached, as the initial assignment was drawn from the exact distribution to start with, which is possible for examples with n low. We then generated a log-log plot of the average absolute error (averaged over all examples) for one of the marginal statistics (13) as compared to the true value (8). This was done for two different values of the annealing parameters σ, which determines the smoothness of the distribution.

As can be seen in figure 11, the "smart chain flipping" proposal is an order of magnitude better than the two other samplers, i.e. it reaches the same level of accuracy in far fewer iterations. For lower temperatures, i.e. sharper distributions, the difference is more pronounced. For higher temperatures, the errors are larger on average (as the sampler needs to explore a larger typical set), and the difference is less pronounced. It can also be seen that the difference between the random flip and (non-smart) chain flipping proposals is negligible.

9. Related work

In recent years, EM has become a popular algorithm for estimating models from incomplete data. As outlined in the introduction, the issue of incomplete data and the data association problem are closely related, though not identical.

The structure from motion problem has been studied extensively in the computer vision literature over the past decades, as we have discussed in detail in Section 2.3. In the introduction we discussed the shortcomings of the existing methods for data-association in the SFM literature.

Also in computer vision, EM has been used to determine membership of pixels to discrete image layers, to compute so-called "layered representations" of the scene (see Torr, Szeliski, & Anandan, 1999, and the references therein). In the latter paper, integrating out the disparity of the pixels in any given layer finesses the correspondence problem in an interesting way.

Thus, it could be viewed as an image-based version of what is attempted in this paper. On the other hand, the motion parameters are assumed to be known by other means in their approach, whereas motion estimation is an integral part of our method.

The SFM problem is similar, and in some cases equivalent, to the map learning problem in robotics. Here a mobile robot is given a sequence of sensor measurements (e.g., range measurements) along with odometry readings, and seeks to construct a map of its environment. In the case of bearing measurements on discrete features, this *concurrent mapping and localization* (*CML*) (Leonard & Durrant-Whyte, 1992), is mathematically identical to a SFM problem. One of the dominant families of algorithms relied on recursive estimation of model features and robot poses by a variable dimension Kalman filter (Castellanos et al., 1999; Castellanos & Tardos, 2000; Leonard & Durrant-Whyte, 1992; Leonard, Cox, & Durrant-Whyte, 1992).

The classical target tracking literature provides a number of methods for data-association (Bar-Shalom & Fortmann, 1988; Popoli & Blackman, 1999) that are used in computer vision (Cox, 1993) and CML (Cox & Leonard, 1994; Feder, Leonard, & Smith, 1999), such as the track splitting filter (Zhang & Faugeras, 1992), the Joint Probabilistic Data Association Filter (JPDAF) (Rasmussen & Hager, 1998), and the multiple hypothesis tracker (MHT) (Reid, 1979; Cox & Leonard, 1994; Cox & Hingorani, 1994). Unfortunately the latter, more powerful methods have exponential complexity so suboptimal approximations are used in practice. However, the strategies for hypothesis pruning are based on assumptions such as motion continuity that are often violated in practice (Seitz & Dyer, 1995). Thus, they are not directly applicable to the SFM or CML problem when the measurements do not arrive in a temporally continuous fashion.

Thus, both vision and map learning approaches assume that the data association problem is solved, either through uniquely identifiable features in the environment of a robot, or through sensor streams that make it possible to track individual features. Of particular difficulty, thus, is the problem of mapping cyclic environments (Gutmann & Konolige, 2000), where features cannot be tracked and the data association problem arises naturally. Recently, an alternative class of algorithms has been proposed that addresses the data association problem (Burgard et al., 1999; Shatkay & Kaelbling, 1997; Shatkay, 1998; Thrun, Fox, & Burgard, 1998a, 1998b). Like ours, these algorithms are based on EM, and they have been demonstrated to accommodate ambiguities and large odometric errors. These algorithms are similar in spirit to the one proposed here in that they formulate the mapping problem as estimation problem from incomplete data, and use the E-step of EM to estimate expectations over those missing data. There are essential differences, though. In particular, these algorithms consider the camera positions as missing data, whereas ours regard the camera poses as model parameters, and instead the correspondence matrix is being estimated in the E-step.

Recently, EM has also been proposed in the target-tracking literature to perform smoothing of tracks, leading to the Probabilistic Multi-Hypothesis Tracker (PMHT) (Avitzour, 1992; Streit & Luginbuhl, 1994; Gauvrit, Le Cadre, & Jauffret, 1997). In the PMHT, the same conditional independence assumptions are made as in this paper, and identical expressions are obtained for the virtual measurements in the M-step. However, the PMHT makes the same motion continuity assumptions as the classical JPDAF and MHT algorithms, which

we do not assume. Moreover, in our work we optimize for structure (targets) *and motion*, a considerably more difficult problem. Most importantly, however, the PMHT altogether abandons the mutual exclusion constraint in the interest of computational efficiency. In contrast, in our work we have shown that the correct distribution in the E-step can be efficiently approximated by Markov chain Monte Carlo sampling. Nevertheless, the PMHT is a very elegant algorithm, and we conjecture that combining the PMHT with our efficient sampler in the E-step could lead to a novel, approximately optimal tracker and/or smoother of interest to the target tracking community.

The new proposal strategies we propose for efficient sampling of assignments bear an interesting relation to research in the field of computational complexity theory. The "chain flipping" proposal is related in terms of mechanism, if not description, to the Broder chain, an MCMC type method to generate (unweighted) assignments at random (Broder, 1986). However, our method is specifically geared towards sampling from *weighted* assignments, and uses the weights to bias proposals towards more likely assignments.

10. Discussion

In this paper we have presented a novel tool, which enables us to learn models from data in the presence of non-trivial data association problems. We have applied it successfully to the structure from motion problem *with unknown correspondence*, significantly extending the applicability of these methods to new imaging situations. In particular, our method can cope with images given in arbitrary order and taken from widely separate viewpoints, obviating the temporal continuity assumption needed to track features over time.

The final algorithm is simple and easy to implement. As summarized in Section 5, at each iteration one only needs to obtain a sample of probable assignments, compute the virtual measurements, and solve a synthetic SFM problem using known methods. In addition, we have developed a novel sampling strategy, called "smart chain flipping", to calculate these virtual measurements efficiently using the Metropolis-Hastings algorithm.

There is plenty of opportunity for future work. In this paper, we make the commonly made assumption that all 3D features are seen in all images (Tomasi & Kanade, 1992; Hartley, 1994; Mclauchlan & Murray, 1995). Our approach does not depend on this assumption, however. In Dellaert (2001) the approach was extended to deal with spurious measurements, by the introduction of a NULL feature (as in Gold et al. (1998)), and occlusion, through the development of a more sophisticated prior on assignments. In addition, (Dellaert, 2001) also shows how appearance measurements can be easily integrated within the EM framework.

Allowing occlusion introduces the thorny issue of model selection, which we have as yet not addressed. In the presence of occlusion it is not commonly known a priori how many features actually exist in the world. This problem of model selection has been addressed successfully before in the context of vision (Ayer & Sawhney, 1995; Torr, 1997), and it is hoped that the lessons learned there can equally apply in the current context.

As argued in the introduction to this paper, the data association problem arises in many problems of learning models from data. While the current work has been phrased in the context of the structure from motion problem in computer vision, we conjecture that the general approach is more widely applicable. For example, as discussed above, the robot

mapping problem shares a similar set of constraints, making the chain flipping proposal distribution directly applicable. Thus, just as we employed off-the-shelf techniques for solving the SFM problem with *known* correspondences, EM and our new MCMC techniques can be stipulated to the rich literature on concurrent mapping and localization (CML) with known correspondences. Such an approach would "bootstrap" these techniques to cases with *unknown* correspondence, which has great practical importance, particularly in the area of multi-robot mapping. As a second example, we suspect that our MCMC chain flipping approach is also applicable to visual object identification from distributed sensors, where others (Pasula et al., 1999) have already successfully applied EM and MCMC to solve the data association problem. Data association problems occur in a wide range of learning models from data. The application of our approach to other data association problems is subject of future research.

Acknowledgments

This work was performed when all authors were at Carnegie Mellon University. It was partially supported by grants from Intel Corporation, Siebel Systems, SAIC, the National Science Foundation under grants IIS-9876136, IIS-9877033, and IIS-9984672, by DARPA-ATO via TACOM (contract number DAAE07-98-C-L032), and by a subcontract from the DARPA MARS program through SAIC, which is gratefully acknowledged. The views and conclusions contained in this document are those of the author and should not be interpreted as necessarily representing official policies or endorsements, either expressed or implied, of the United States Government or any of the sponsoring institutions.

References

Avitzour, D. (1992). A maximum likelihood approach to data association. *IEEE Trans. on Aerospace and Electronic Systems, 28:2*, 560–566.

Ayer, S., & Sawhney, H. S. (1995). Layered representation of motion video using robust maximum-likelihood estimation of mixture models and MDL encoding. In *Int. Conf. on Computer Vision (ICCV)* (pp. 777–784).

Bar-Shalom, Y., & Fortmann, T. E. (1988). *Tracking and data association*. New York: Academic Press.

Basri, R., Grove, A. J., & Jacobs, D. W. (1998). Efficient determination of shape from multiple images containing partial information. *Pattern Recognition, 31:11*, 1691–1703.

Beardsley, P. A., Torr, P. H. S., & Zisserman, A. (1996). 3D model acquisition from extended image sequences. In *Eur. Conf. on Computer Vision (ECCV)* (pp. II:683–695).

Bertsekas, D. P. (1991). *Linear network optimization: Algorithms and codes*. Cambridge, MA: The MIT Press.

Borenstein, J., Everett, B., & Feng, L. (1996). *Navigating mobile robots: Systems and techniques*. Wellesley, MA: A. K. Peters, Ltd.

Broder, A. Z. (1986). How hard is to marry at random? (On the approximation of the permanent). In *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing* (pp. 50–58). Berkeley, California.

Broida, T., & Chellappa, R. (1991). Estimating the kinematics and structure of a rigid object from a sequence of monocular images. *IEEE Trans. on Pattern Analysis and Machine Intelligence, 13:6*, 497–513.

Burgard, W., Fox, D., Jans, H., Matenar, C., & Thrun, S. (1999). Sonar-based mapping of large-scale mobile robot environments using EM. In *Proceedings of the International Conference on Machine Learning*, Bled, Slovenia.

Castellanos, J. A., Montiel, J. M. M., Neira, J., & Tardos, J. D. (1999). The SPmap: A probabilistic framework for simultaneous localization and map building. *IEEE Trans. on Robotics and Automation, 15:5*, 948–953.

Castellanos, J. A., & Tardos, J. D. (2000). *Mobile robot localization and map building: A multisensor fusion approach*. Boston, MA: Kluwer Academic Publishers.

Cook, W. J., Cunningham, W. H., Pulleyblank, W. R., & Schrijver, A. (1998). *Combinatorial optimization*. New York, NY: John Wiley & Sons.

Cox, I. J. (1993). A review of statistical data association techniques for motion correspondence. *Int. J. of Computer Vision, 10:1*, 53–66.

Cox, I. J., & Hingorani, S. L. (1994). An efficient implementation and evaluation of Reid's multiple hypothesis tracking algorithm for visual tracking. In *Int. Conf. on Pattern Recognition (ICPR),* (Vol. 1, pp. 437–442). Jerusalem, Israel .

Cox, I. J., & Leonard, J. J. (1994). Modeling a dynamic environment using a Bayesian multiple hypothesis approach. *Artificial Intelligence, 66:2*, 311–344.

Dellaert, F. (2001). Monte Carlo EM for data association and its applications in computer vision. PhD thesis, School of Computer Science, Carnegie Mellon. Also available as Technical Report CMU-CS-01-153.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B, 39:1*, 1–38.

Deriche, R., & Faugeras, O. D. (1990). Tracking line segments. *Image and Vision Computing, 8*, 261–270.

Doucet, A., de Freitas, N., & Gordon, N. (Eds.). (2001). *Sequential Monte Carlo methods in practice*. New York: Springer-Verlag.

Feder, H. J. S., Leonard, J. J., & Smith, C. M. (1999). Adaptive mobile robot navigation and mapping. *International Journal of Robotics Research, Special Issue on Field and Service Robotics, 18:7*, 650–668.

Gauvrit, H., Le Cadre, J. P., & Jauffret, C. (1997). A formulation of multitarget tracking as an incomplete data problem. *IEEE Trans. on Aerospace and Electronic Systems, 33:4*, 1242–1257.

Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (Eds.) (1996). *Markov chain Monte Carlo in practice*. Chapman and Hall.

Gold, S., Rangarajan, A., Lu, C., Pappu, S., & Mjolsness, E. (1998). New algorithms for 2D and 3D point matching. *Pattern Recognition, 31:8*, 1019–1031.

Gutmann, J.-S., & Konolige, K. (2000). Incremental mapping of large cyclic environments. In *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*.

Hartley, H. O. (1958). Maximum likelihood estimation from incomplete data. *Biometrics, 14*, 174–194.

Hartley, R. I. (1994). Euclidean reconstruction from uncalibrated views. In *Application of Invariance in Computer Vision*, 237–256.

Hartley, R., & Zisserman, A. (2000). *Multiple view geometry in computer vision*. Cambridge: Cambridge University Press.

Hastings, W. K. (1970). Monte carlo sampling methods using markov chains and their applications. *Biometrika, 57*, 97–109.

Jacobs, D. W. (1997). Linear fitting with missing data: Applications to structure from motion and to characterizing intensity images. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (pp. 206–212).

Kozen, D. C. (1991). *The design and analysis of algorithms*. Berlin: Springer-Verlag.

Kurata, T., Fujiki, J., Kourogi, M., & Sakaue, K. (1999). A robust recursive factorization method for recovering structure and motion from live video frames. In *1999 ICCV Workshop on Frame Rate Processing*, Corfu, Greece.

Leonard, J. J., Cox, I. J., & Durrant-Whyte, H. F. (1992). Dynamic mmap building for an autonomous mobile robot. *Int. J. Robotics Research, 11:4*, 286–289.

Leonard, J. J. & Durrant-Whyte, H. F. (1992). *Directed sonar sensing for mobile robot navigation*. Boston: Kluwer Academic.

Longuet-Higgins, H. C. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature, 293*, 133–135.

McLachlan, G. J., & Basford, K. E. (1988). *Mixture models: Inference and applications to clustering*. New York: Marcel Dekker.

McLachlan, G. J., & Krishnan, T. (1997). *The EM algorithm and extensions*. Wiley series in probability and statistics. John Wiley & Sons.

McLauchlan, P., & Murray, D. (1995). A unifying framework for structure and motion recovery from image sequences. In *Int. Conf. on Computer Vision (ICCV)* (pp. 314–320).

Morris, D. D., & Kanade, T. (1998). A unified factorization algorithm for points, line segments and planes with uncertainty models. In *Int. Conf. on Computer Vision (ICCV)* (pp. 696–702).

Morris, D. D., Kanatani, K., & Kanade, T. (1999). Uncertainty modeling for optimal structure from motion. In *ICCV Workshop on Vision Algorithms: Theory and Practice*.

Neal, R. M. (1993). Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG-TR-93-1, Dept. of Computer Science, University of Toronto.

Papadimitriou, C. H., & Steiglitz, K. (1982). *Combinatorial optimization: Algorithms and complexity*. Englewood Cliff, NJ: Prentice-Hall.

Pasula, H., Russell, S., Ostland, M., & Ritov, Y. (1999). Tracking many objects with many sensors. In *Int. Joint Conf. on Artificial Intelligence (IJCAI)*, Stockholm.

Poelman, C., & Kanade, T. (1997). A paraperspective factorization method for shape and motion recovery. *IEEE Trans. on Pattern Analysis and Machine Intelligence, 19:3*, 206–218.

Popoli, R., & Blackman, S. S. (1999). *Design and analysis of modern tracking systems*. Artech House Radar Library.

Rasmussen, C., & Hager, G. D. (1998). Joint probabilistic techniques for tracking objects using multiple vision clues. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)* (pp. 191–196).

Reid, D. B. (1979). An algorithm for tracking multiple targets. *IEEE Trans. on Automation and Control, AC-24:6*, 84–90.

Scott, G. L., & Longuet-Higgins, H. C. (1991). An algorithm for associating the features of two images. *Proceedings of Royal Society of London, B-244*, 21–26.

Seitz, S. M., & Dyer, C. R. (1995). Complete structure from four point correspondences. In *Int. Conf. on Computer Vision (ICCV)* (pp. 330–337).

Shapiro, L. S., & Brady, J. M. (1992). Feature-based correspondence: An eigenvector approach. *Image and Vision Computing, 10:5*, 283–288.

Shatkay, H. (1998). *Learning models for robot navigation*. Ph.D. thesis, Computer Science Department, Brown University, Providence, RI.

Shatkay, H., & Kaelbling, L. (1997). Learning topological maps with weak local odometric information. In *Proceedings of IJCAI-97*, IJCAI, Inc.

Smith, A. F. M., & Gelfand, A. E. (1992). Bayesian statistics without tears: A sampling-resampling perspective. *American Statistician, 46:2*, 84–88.

Spetsakis, M., & Aloimonos, Y. (1991). A multi-frame approach to visual motion perception. *Int. J. of Computer Vision, 6:3*, 245–255.

Streit, R., & Luginbuhl, T. (1994). Maximum likelihood method for probabilistic multi-hypothesis tracking. In *Proc. SPIE* (Vol. 2335, pp. 394–405).

Szeliski, R., & Kang, S. B. (1993). Recovering 3D shape and motion from image streams using non-linear least squares. Technical Report CRL 93/3, DEC Cambridge Research Lab.

Tanner, M. A. (1996). *Tools for statistical inference* (3rd edn). New York: Springer Verlag.

Thrun, S., Fox, D., & Burgard, W. (1998a). A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning, 31*, 29–53. also appeared in *Autonomous Robots, 5*, 253–271.

Thrun, S., Fox, D., & Burgard, W. (1998). Probabilistic mapping of an environment by a mobile robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

Tomasi, C., & Kanade, T. (1992). Shape and motion from image streams under orthography: A factorization method. *Int. J. of Computer Vision, 9:2*, 137–154.

Torr, P. H. S. (1997). An assessment of information criteria for motion model selection. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (pp. 47–53).

Torr, P., Fitzgibbon, A., & Zisserman, A. (1998). Maintaining multiple motion model hypotheses over many views to recover matching and structure. In *Int. Conf. on Computer Vision (ICCV)* (pp. 485–491).

Torr, P. H. S., Szeliski, R., & Anandan, P. (1999). An integrated bayesian approach to layer extraction from image sequences. In *Int. Conf. on Computer Vision (ICCV)* (pp. 983–990).

Triggs, B. (1996). Factorization methods for projective structure and motion. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* (pp. 845–851).

Triggs, B., McLauchlan, P., Hartley, R., & Fitzgibbon, A. (1999). Bundle adjustment—a modern synthesis. In *Vision Algorithms 99*, Corfu, Greece.

Tsai, R. Y., & Huang, T. S. (1984). Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence, 6:1*, 13–27.

Ullman, S. (1979). *The interpretation of visual motion*. Cambridge, MA: MIT Press.

Zhang, Z., & Faugeras, O. D. (1992). Three-dimensional motion computation and object segmentation in a long sequence of stereo frames. *Int. J. of Computer Vision, 7:3*, 211–241.

Received July 25, 2000
Revised June 1, 2001
Accepted September 20, 2001
Final manuscript January 19, 2002