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

Frank Dellaert Steven M. Seitz Charles E. Thorpe
Sebastian Thrun

July 20, 2000

CMU-CS-00-144

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

We gratefully acknowledge the support of the following sponsoring institutions: SAIC corporation; KIRIN brewery company; SONY corporation; DOT and NHTSA; NSF through grants IIS-9984672 and IIS-9876136; DARPA-ATO via TACOM (contract number DAAE07-98-C-L032) and DARPA-ISO via Rome Labs (contract number F30602-98-2-0137); and CS and Robotics at CMU

Keywords: Computer Vision, 3D Scene Analysis, Structure from Motion, Correspondence, Data Association, Probabilistic Algorithms, Expectation-Maximization, Markov Processes, Markov chain Monte Carlo

Abstract

Learning spatial models from sensor data often raises a challenging data association problem of relating parameters in the model to individual measurements. This paper proposes an algorithm based on EM, which simultaneously solves the model learning and the data association problem. The algorithm is developed in the context of the the structure from motion problem, which is the problem of learning a 3D scene model from a collection of image data. To accommodate the spatial constraints in this domain, we introduce the notion of *virtual measurements* as sufficient statistics to be used in the M-step, and 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.

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 & Durrant-Whyte, 1992; Leonard et al., 1992; Shatkay, 1998; Thrun et al., 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 physical world 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 et al., 1996), or the existence of a 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 et al., 1977), which has been applied with great success to clustering problems and a range of other estimation problems with incomplete data (McLachlan & Krishnan, 1997). 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 maximum likelihood model for the distributions computed in the E-step. It has been shown that EM performs hill-climbing in likelihood space, and iterating both steps leads to a model that locally maximizes the likelihood (Dempster et al., 1977; Neal & Hinton, 1998).

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, by Shatkay and Kaelbling (Shatkay & Kaelbling, 1997; Shatkay, 1998), and also by Thrun, Burgard, and Fox (Burgard et al., 1999; Thrun et al., 1998b, 1998a). 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 different domain: the structure from motion problem in computer vision. As is commonly the case in SFM, the model that is being learned is the location of all features in 3D, along with the cameras positions in 6D. 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). However, the method we propose does not depend crucially on this assumption, and we will discuss at the end of this paper how to extend it to more general 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 vast majority of literature on SFM considers special situations where the data association problem can be solved trivially. Some approaches simply assume that data correspondence is known *a priori* (Ullman, 1979; Longuet-Higgins, 1981; Tsai & Huang, 1984; Hartley, 1994; McLauchlan & Murray, 1995; Morris & Kanade, 1998). Other approaches only 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; Azarbayejani & Pentland, 1995; Poelman & Kanade, 1997). Several authors considered the special case of correct but incomplete correspondence, by interpolating occluded features (Tomasi & Kanade, 1992; Jacobs, 1997; Basri et al., 1998), or expanding a minimal correspondence into a complete correspondence (Seitz & Dyer, 1995). However, these approaches require that a sufficient and non-degenerate set of initial correspondences be provided a priori which is assumed to be correct. A few authors have proposed methods for using geometric constraints to facilitate the correspondence problem in uncalibrated images. In particular, Irani (Irani, 1999) described how geometric rank constraints can be used to facilitate optical flow computation over closely-spaced views. Finally, methods based on the robust recovery of epipolar geometry, e.g. using RANSAC (Beardsley et al., 1996; Torr et al., 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 reasonably 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 of interleaving model learning and data association. It has been referred to as the most difficult part of structure recovery (Torr et al., 1998), and it is also the problem addressed in this paper. The reader should notice that it 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 make 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 huge number of local minima in the space of data associations.

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 Monte Carlo Markov Chain (MCMC) (Doucet et al., 2001; Gilks et al., 1996; Neal, 1993). In particular, our approach uses the popular Metropolis-Hastings algorithm (Hastings, 1970; Metropolis et al., 1953; Smith & Gelfand, 1992), for approximating the desired posterior summaries. However, Metropolis-Hastings can be extremely inefficient in large spaces (Gilks et al., 1996). To remedy this problem, we propose a new proposal distribution based on *chain flipping*, which is specifically suited for data association problems. Chain flipping is a method that can quickly jump across globally different assignments. Experimental results show that this approach is orders of magnitude more efficient than commonly used approaches that consider only local changes in the MCMC sampling process (e.g., Gibbs sampling (Geman & Geman, 1984; Smith & Gelfand, 1992)).

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, most notably in (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. In fact, our approach is orthogonal to the vast majority of work in the SFM field in that is can be combined with virtually any optimization 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.

The remainder of this paper is structured as follows: in the next section, Section 2, we introduce the structure from motion problem, the assumptions we make, and show how EM can be used to solve the associated data association problem. In Section 4 we introduce the notion of virtual measurements, whereas Section 5 discusses how they can be computed using MCMC methods. Sections 6 and 7 provide an overview of the main algorithm, and present results for several non-trivial computer vision datasets with challenging data association problems. In Section 8 we then present the "chain flipping" proposal distribution, which makes the E-step computationally efficient. This is illustrated with additional results on the convergence speed of the different sampling

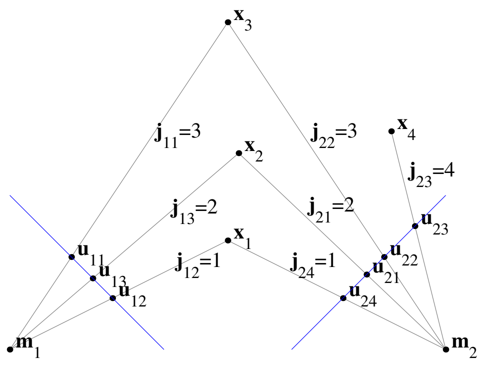

Figure 1: An example with 4 features seen in 2 images. The 7 measurements $\mathbf{u}_{ik}$ are assigned to the individual features $\mathbf{x}_j$ by means of the assignment variables j_{ik}.

methods in Section 9.

2 EM for Structure from Motion without Correspondence

In this section we introduce the structure from motion problem and the assumptions we make, and discuss existing SFM methods to solve the associated maximum-likelihood problem for known correspondence. For the case of *unknown* correspondence*,* we show how the expectation-maximization (EM) algorithm can be used to learn the model parameters, even if we do not know which measurements correspond to which scene elements.

2.1 Problem Statement, Notation, and Assumptions

The *structure from motion* (SFM) problem is this: given a set of images of a scene, taken from different viewpoints, 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 the pixel values of the images themselves, or whether a set of image 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. As most other approaches to SFM (Tomasi & Kanade, 1992; Szeliski & Kang, 1993; Hartley, 1994; McLauchlan & Murray, 1995; Poelman & Kanade, 1997; Morris & Kanade, 1998), we adopt a feature-based approach, i.e., we assume that the input to the algorithm is a set of discrete *image measurements* $\mathbf{U} = \{\mathbf{u}_{ik} | i \in 1..m,\ k \in 1..K_i\}$, where i is the image index, and K_i is the number of

measurements in image i. It is assumed that these measurements $\mathbf{u}_{ik}$ correspond to the projection of a set of real world, *3D features* $\mathbf{X} = \{\mathbf{x}_j | j \in 1..n\}$. In any real imaging situation there is some uncertainty in the measurement, and so we model the measurement process as one of projection followed by additive noise.

2. In contrast to the feature-based methods shortcited above, it is *not* required that the correspondence between the measurements in the different images is known. This is exactly the data association problem we solve using EM. To model the correspondence between measurements $\mathbf{u}_{ik}$ and 3D feature $\mathbf{x}_j$ 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, i.e. it is hidden data.

3. We also allow images to be taken from a set of arbitrary *camera poses* $\mathbf{M} = \{\mathbf{m}_i | i \in 1..m\}$. 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; Cox & Hingorani, 1994), or otherwise constrain the data association problem (Beardsley et al., 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 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 solved, most existing approaches to SFM can be viewed as *maximum likelihood* (ML) methods. ML methods attempt to find those model parameters Θ that are most likely to have generated the data. In our case we have

1. 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*.

2. 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 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 $P(\mathbf{U}, \mathbf{J}|\Theta)$, the conditional density of the data given the model. To evaluate the likelihood, we need to assume a generative model. In particular, we assume that each measurement $\mathbf{u}_{ik}$ is generated by applying the *measurement function* $\mathbf{h}$ to the model, after which it is corrupted by additive noise $\mathbf{n}$:

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

The above expression also makes explicit that a given measurement $\mathbf{u}_{ik}$ depends only on the camera 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 the measurement function 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 et al., 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 (or learn) a distribution for the noise $\mathbf{n}$. In the case that the noise $\mathbf{n}$ on the measurements is i.i.d. zero-mean Gaussian noise with standard deviation σ, the negative log-likelihood is simply a sum of squared re-projection errors:

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

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.

3 Existing Methods for Structure from Motion

The structure from motion problem has been studied extensively in the computer vision literature over the past decade. A good survey of techniques can be found in Hartley and Zisserman's recent book on multiple view geometry (Hartley & Zisserman, 2000).

The earliest work focused on reconstruction from two images only (Ullman, 1979; Longuet-Higgins, 1981; Tsai & Huang, 1984). 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, i.e., the projection is orthogonal to the image plane and has its focus at infinity, the estimate Θ^* for the model parameters that minimize (3) can be found efficiently using using a *factorization* approach (Tomasi & Kanade, 1992). Using this technique, singular value decomposition (SVD) is first applied to the data $\mathbf{U}$ in order to obtain *affine* structure $\mathbf{X}^a$ and motion $\mathbf{M}^a$, so called because they are only defined up to a 3D affine transformation. To do this, the assignment information $\mathbf{J}$ is needed to re-arrange the data $\mathbf{U}$ in the correct order needed for SVD. To get true Euclidean structure and motion, an additional step is needed that imposes metric constraints on $\mathbf{M}^a$. The factorization method has the advantages that it is fast and does not need a good initial estimate of structure and motion 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). These are well developed techniques, and the reader is referred to (Tomasi & Kanade, 1992; Poelman & Kanade, 1997; Morris & 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, as the projection involves a division by the depth of a feature point relative to the camera. In this case 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; Weng et al., 1993; Hartley, 1994; Cooper & Robson, 1996; TRiggs et al., 1999). A method that works with line measurements is (Taylor & Kriegman, 1995). The advantage with respect to factorization is that it gives the exact ML estimate, when it converges. The disadvantage, however, is that it can get stuck in local minima, and thus a good initial estimate for structure and motion needs to be available. To alleviate this, recursive estimation techniques can be used to process the images as they arrive (Broida & Chellappa, 1991; Azarbayejani & Pentland, 1995).

3.1 SFM without Correspondences

In the case that the correspondences are unknown we cannot directly apply the methods discussed in Section 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. To see this, note that the maximum likelihood estimate $\Theta^* = (\mathbf{X}^*, \mathbf{M}^*)$ of structure and motion given *only* the measurements $\mathbf{U}$ is given by:

$$\Theta^* = \underset{\Theta}{\operatorname{argmax}} \ \log L(\Theta; \mathbf{U}) \tag{4}$$

Although this might seem counterintuitive at first, the above states that *we can find the ML structure and motion without explicitly reasoning about which assignment might be correct*. We 'only' need to maximize the likelihood $L(\Theta; \mathbf{U})$, which does directly not depend on $\mathbf{J}$. To calculate $L(\Theta; \mathbf{U})$, note that by total probability we can write it as a sum of likelihood terms of the form (1), with one term for *every possible* assignment vector $\mathbf{J}$:

$$L(\Theta; \mathbf{U}) = \sum_{\mathbf{J}} L(\Theta; \mathbf{U}, \mathbf{J}) \tag{5}$$

Now, for any realistic number of features n and number of images m, the number of possible assignments explodes combinatorially. Given our assumptions, there are $n!$

possible assignment vectors $\mathbf{J}_i$ in each image, yielding a total of $n!^m$ assignments $\mathbf{J}$. Under mild assumptions we can factor the likelihood $L(\Theta; \mathbf{U}, \mathbf{J})$ over the different images, but even then the number of assignments in each image remains $n!$. In summary, $L(\Theta; \mathbf{U})$ is in general hard to obtain explicitly, as it involves summing over a combinatorial number of possible assignments.

3.2 The Expectation Maximization Algorithm

A key insight is that we can use the well-known EM algorithm (Hartley, 1958; Dempster et al., 1977; McLachlan & Krishnan, 1997) 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 (5) 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 $Q^t(\Theta)$:

$$Q^t(\Theta) = \sum_{\mathbf{J}} f^t(\mathbf{J}) \log L(\Theta; \mathbf{U}, \mathbf{J}) \tag{6}$$

   where the expectation is taken with respect to the posterior distribution $f^t(\mathbf{J}) \triangleq P(\mathbf{J}|\mathbf{U}, \Theta^t)$ 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 $Q^t(\Theta)$:

$$\Theta^{t+1} = \underset{\Theta}{\text{argmax}} \;\; Q^t(\Theta)$$

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 et al., 1977; McLachlan & Krishnan, 1997).

4 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. Below 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). To start with the derivation, 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 equation (6), we obtain:

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

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

$$Q^t(\Theta) = -\frac{1}{2\sigma^2} \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{n} f^t_{ijk} \|\mathbf{u}_{ik} - \mathbf{h}(\mathbf{m}_i, \mathbf{x}_j)\|^2 \tag{8}$$

where f^t_{ijk} is the *marginal posterior probability* $P(j_{ik} = j|\mathbf{U}, \Theta^t)$. Note that this does not depend on the assumption of Gaussian noise, but rather on the conditional independence of image measurements. Note also that a similar trick cannot be applied to the total likelihood (5), as the latter is a sum of likelihoods, not *log*-likelihoods. The marginal probabilities f^t_{ijk} can be calculated by summing $f^t(\mathbf{J})$ over all possible assignments $\mathbf{J}$ where $j_{ik} = j$:

$$f^t_{ijk} \triangleq P(j_{ik} = j|\mathbf{U}, \Theta^t) = \sum_{\mathbf{J}} \delta(j_{ik}, j) f^t(\mathbf{J}) \tag{9}$$

with $\delta(.,.)$ is the Kronecker delta function.

The main point to be made in this section is this: it can be shown by simple algebraic manipulation that (8) 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

$$Q^t(\Theta) = C - \frac{1}{2\sigma^2} \sum_{i=1}^{m} \sum_{j=1}^{n} \|\mathbf{v}^t_{ij} - \mathbf{h}(\mathbf{m}_i, \mathbf{x}_j)\|^2 \tag{10}$$

where the *virtual measurements* $\mathbf{v}^t_{ij}$ are defined as

$$\mathbf{v}^t_{ij} \triangleq \sum_{k=1}^{n} f^t_{ijk} \mathbf{u}_{ik} \tag{11}$$

Each virtual measurement $\mathbf{v}^t_{ij}$ is simply a weighted average of the original measurements $\mathbf{u}_{ik}$ in the i-th image, and the weights are the marginal probabilities f^t_{ijk}.

The important point is that the M-step objective function (10) 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, of which many are discussed in Section 3, can be used to implement the M-step.* This provides an intuitive interpretation for the overall algorithm:

1. **E-step:** Calculate the weights f^t_{ijk} from the distribution over assignments. Then, in each of the m images calculate n virtual measurements $\mathbf{v}^t_{ij}$.

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. What is left is to show how the E-step can be implemented.

5 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 $f^t_{ijk} = P(j_{ik} = j|\mathbf{U}, \Theta^t)$ needed to calculate the virtual measurements $\mathbf{v}^t_{ij}$.

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

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

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

$$P(\mathbf{J}_i|\mathbf{U}_i, \Theta^t) \propto P(\mathbf{J}_i|\Theta^t) \exp\left[-\frac{1}{2\sigma^2}\sum_{k=1}^{n} \|\mathbf{u}_{ik} - \mathbf{h}(\mathbf{m}^t_i, \mathbf{x}^t_j)\|^2\right] \tag{12}$$

The second part of this expression is simple enough. However, the prior probability $P(\mathbf{J}_i|\Theta^t)$ 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 $\mathbf{x}_j$. In other words, if we assume that *valid* assignments are all equally likely, the prior probability of an assignment is

$$P(\mathbf{J}_i|\Theta^t) = \begin{cases} \frac{1}{n!} & \text{if } \mathbf{J}_i \text{ is a valid assignment} \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

While it is easy to evaluate the posterior probability $f^t_i(\mathbf{J}_i)$ for any *given assi*gnment $\mathbf{J}_i$ through (12), the mutual exclusion constraint makes it difficult to analytically express the weights f^t_{ijk} by substituting those expressions in Equation 9. We know of no efficient closed form expression for f^t_{ijk}.

5.1 Sampling the Distribution over Assignments $\mathbf{J}_i$

The solution we employ is to instead *sample* from the posterior probability distribution $f^t_i(\mathbf{J}_i)$ 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 the EM algorithm where the E-step is executed by a Monte-Carlo process (Tanner, 1996; McLachlan & Krishnan, 1997).

To sample from $f_i^t(\mathbf{J}_i)$ we use the Metropolis-Hastings algorithm, a Markov Chain Monte Carlo method (MCMC) (Neal, 1993; Gilks et al., 1996; Doucet et al., 2001). MCMC methods can be used to obtain approximate values for expectations over distributions that defy easy analytical solutions. In our case, the *target distribution* is the posterior distribution $f_i^t(\mathbf{J}_i)$ over assignment vectors $\mathbf{J}_i$ in image i. All MCMC methods work the same way: they generate a sequence of *states*, in our case the assignments $\mathbf{J}_i$, with the property that the collection of generated assignments $\mathbf{J}_i^r$ approximates a sample from the target 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; Metropolis et al., 1953) is one way to simulate a Markov chain with the correct stationary distribution, without explicitly building the full transition probability matrix (which would be an 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 et al., 1996)) :

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

$$a = \frac{f_i^t(\mathbf{J}_i')}{f_i^t(\mathbf{J}_i^r)} \frac{Q(\mathbf{J}_i^r; \mathbf{J}_i')}{Q(\mathbf{J}_i'; \mathbf{J}_i^r)} \tag{14}$$

   where $f_i^t(\mathbf{J}_i) = P(\mathbf{J}_i|\mathbf{U}_i, \Theta^t)$ is the *target distribution*.

4. **If** $a >= 1$ then accept $\mathbf{J}_i'$, i.e. we set $\mathbf{J}_i^{r+1} = \mathbf{J}_i'$.
   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 (11), we need to compute the marginal probabilities f^t_{ijk} from the sample $\{\mathbf{J}^r_i\}$. Fortunately, this can be done without explicitly storing the samples, by keeping running counts of how many times each measurement $\mathbf{u}_{ik}$ is assigned to feature j, as

$$f^t_{ijk} \approx \frac{1}{R}\sum_{r=1}^{R}\delta(j^r_{ik}, j) \tag{15}$$

is easily seen to be the Monte Carlo approximation to (9).

Finally, in order to implement the sampler, we need to know how to propose new assignments $\mathbf{J}'_i$ (the proposal density $Q(\mathbf{J}^r_i;\mathbf{J}'_i)$, and how to compute the ratio a. Both elements are discussed in detail in Section 8.

6 Summary of the Algorithm

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^t_{ijk} (equation 15).
3. Calculate the virtual measurements $\mathbf{v}^t_{ij}$ with (11).
4. Find the new estimate Θ^{t+1} for structure and motion using the virtual measurements $\mathbf{v}^t_{ij}$ as data. This can be done using any SFM method discussed in Section 3.
5. If not converged, return to step 2.

In practice it is important to add *annealing* to this basic EM scheme, to avoid getting stuck in local minima. In simulated annealing we artificially increase the noise parameter σ in early iterations, gradually decreasing it to its correct value. This has two beneficial consequences. First, the posterior distribution $f^t_i(\mathbf{J}_i)$ 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.

7 Results for SFM without Correspondence

In this section we show results on three sets of images for which the SFM problem is non-trivial in the absence of correspondence. For each set we highlight a particular property of our method. For all the results we present, the input to the algorithm was 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

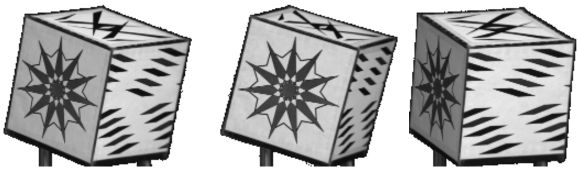

Figure 2: Three out of 11 cube images. Although the images were originally taken as a sequence in time, the ordering of the images is irrelevant to our method.

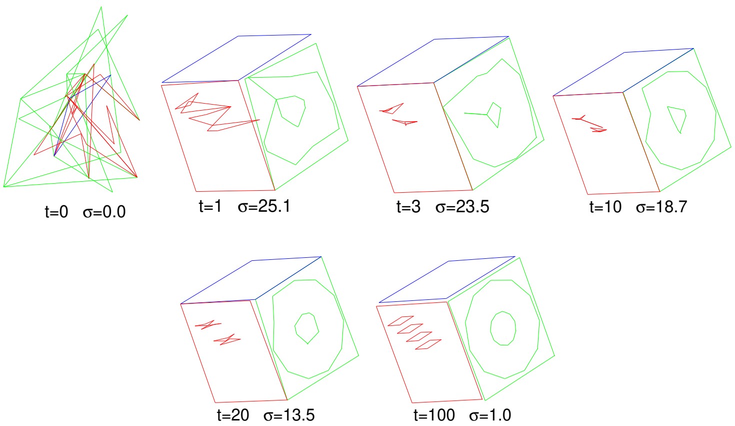

Figure 3: The structure estimate as initialized and at successive iterations t of the algorithm.

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

100 iterations, with the annealing parameter σ decreasing exponentially from 25 pixels to 1 pixel. For each EM iteration, we ran the sampler in each image for 10000 steps. An entire run takes about a minute of CPU time on a standard PC. As is typical for EM, the algorithm can sometimes get stuck in local minima, in which case we restart it manually.

In practice, the algorithm converges consistently and fast to an estimate for the structure and motion where the correct assignment is the most probable one, and where most if not 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. Incidentally, we found the algorithm converges less often when we replace the random initialization by a 'good' initial estimate where all the points in some image are projected onto a plane of constant depth.

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^t_{ijk} at two different times 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. If all the probability were concentrated in one consistent assignment over all images, the large f^t_{ijk} matrix would be a set of identical permutation matrices stacked one upon the other.

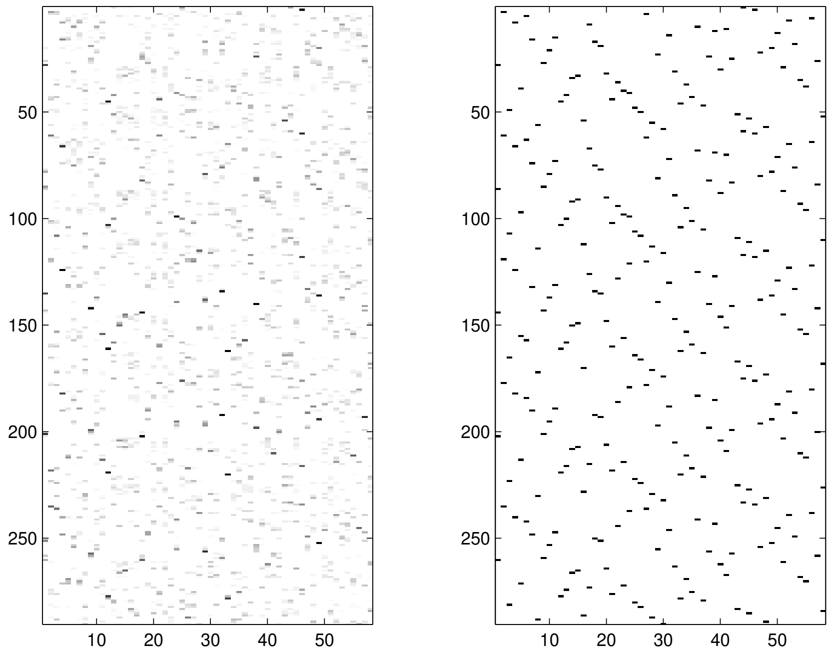

Figure 5: The marginal probabilities f^t_{ijk} at an early and at a later iteration, respectively. Each row corresponds to a measurement $\mathbf{u}_{ik}$, grouped according to image index, whereas the columns represent the n features $\mathbf{x}_j$. In this example $n = 58$ and $m = 5$. Black corresponds to a marginal probability of 1.

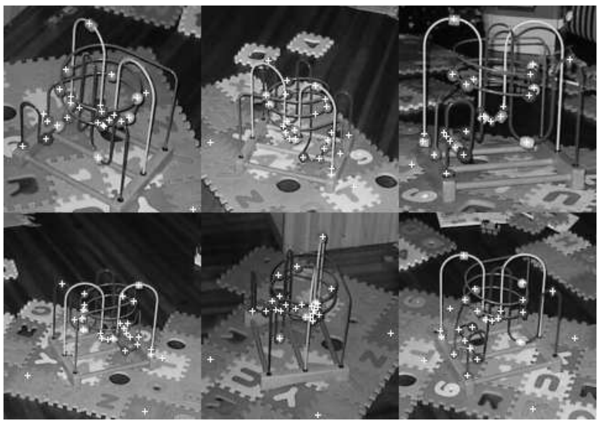

Figure 6: 6 out of 8 images of a wire-frame toy, taken from widely different viewpoints.

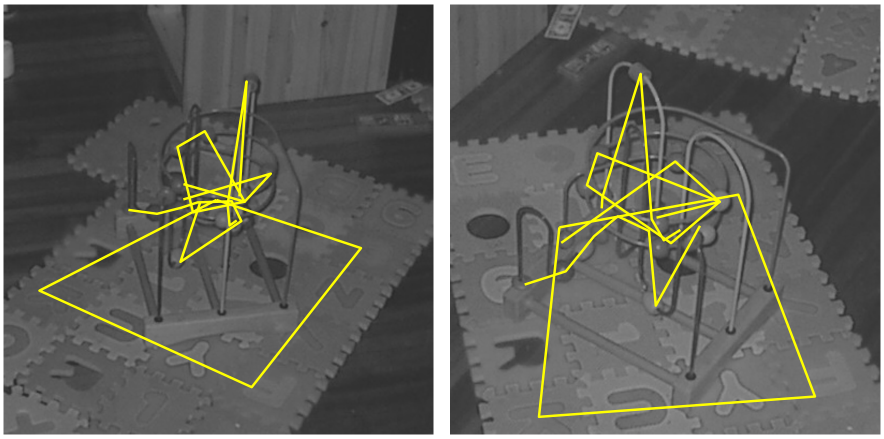

Figure 7: Recovered structure for wire-frame toy re-projected in 2 images.

The algorithm also deals with situations where the images are taken from widely separate viewpoints, as is the case for the images in Figure 6. In this sequence, the image features used were the colored beads on the wire-frame toy in the image, plus four points on the ground plane. Images were taken from both sides of the object. Because of the 'see-through' nature of the object, there is also a lot of potential confusion between image measurements. Figure 7 shows the wire-frame model obtained by our method, where each of the wires corresponds to one of the wires on the toy. Although in the final iteration there is still disagreement between images about the most likely feature assignment, the overall structure of the model is recovered despite the arbitrary configuration of the cameras.

8 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*. Especially the reliance on a Monte Carlo estimate for the expectations calculated in the E-step *might* seem troublesome in this regard. 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.

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.

8.1 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 $e = (u_k, v_j)$:

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

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 equation (12), 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:

$$f_i^t(\mathbf{J}_i) \propto \exp\left[-\frac{1}{2\sigma^2}\sum_{k=1}^{n}\|\mathbf{u}_{ik} - \mathbf{h}(\mathbf{m}_i^t, \mathbf{x}_j^t)\|^2\right] \propto e^{-w(\mathbf{J}_i)} \tag{16}$$

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 (16) 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 (16).* Below we drop the image index i, and think solely in terms of the weighted assignment problem.

8.2 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.

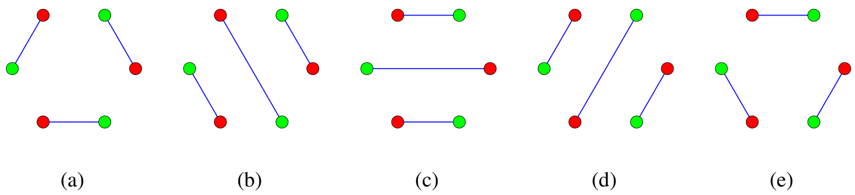

Figure 8: An ambiguous assignment problem with $n = 3$. The regular arrangement of the vertices yields two optimal assignments, (a) and (e), whereas (b-d) are much less likely. The figure illustrates a major problem with “flip proposals”: there is no way to move from (a) to (e) via flip proposals without passing through one of the unlikely states (b-d).

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

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

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 8. There is no way to move from the most likely configurations (a) to (e) via flip proposals without passing through one of the unlikely states (b-d). An MCMC sampler that proposes only such moves can stay stuck in the modes (a) or (e) for a long time.

8.3 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. 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 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 9 (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*

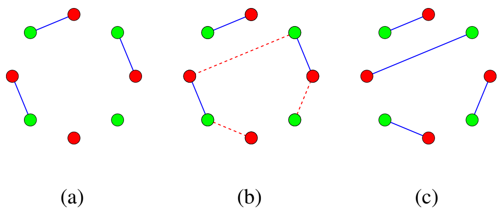

Figure 9: Augmenting paths. (a) Original, partial matching. (b) An augmenting path, alternating between free and matched edges. (c) The resulting matching after augmenting the matching in (a) with the path in (b) .

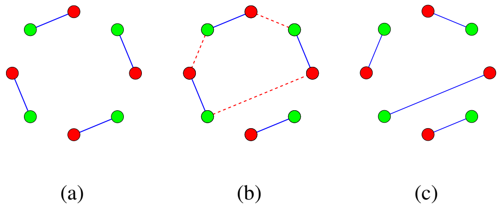

Figure 10: (a) Original assignment. (b) An alternating cycle implementing a k-swap, with k=3 in this example. (c) Newly obtained assignment.

p. One such a path is shown in Figure 9 (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 9 (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 10. In contrast to the optimal algorithms, when sampling we start out with a perfect matching (an assignment), and want to propose a 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.*

8.4 Proposing Moves by "Chain Flipping"

Recall that the goal is to sample from assignments $\mathbf{J}$ 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_0 in U
2. Choose a match v in V by traversing the edge $e = (u, v)$ according to the transition probabilities

$$q(u, v) \triangleq \frac{\exp(-w(u, v))}{\sum_v \exp(-w(u, v))} \tag{17}$$

   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

$$a_{CF} = \frac{P(J')}{P(J)} \frac{Q(J; J')}{Q(J'; J)} = 1 \tag{18}$$

To prove this, note that by (16) and (17) the probability ratio is given by

$$\frac{P(J')}{P(J)} = \frac{e^{-w(J')}}{e^{-w(J)}} = \prod_{u \in C} \frac{q(u, J'(u))}{q(u, J(u))} \tag{19}$$

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:

$$Q(J'; J) = \left(\prod_{(u,v) \in p} q(u, J'(u)) \right) \sum_T P_{MC}(T) \tag{20}$$

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 (19) into (18) yields the surprising result $a = 1$.

A distinct advantage of the CF algorithm is that, as with the Gibbs sampler (Gilks et al., 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.

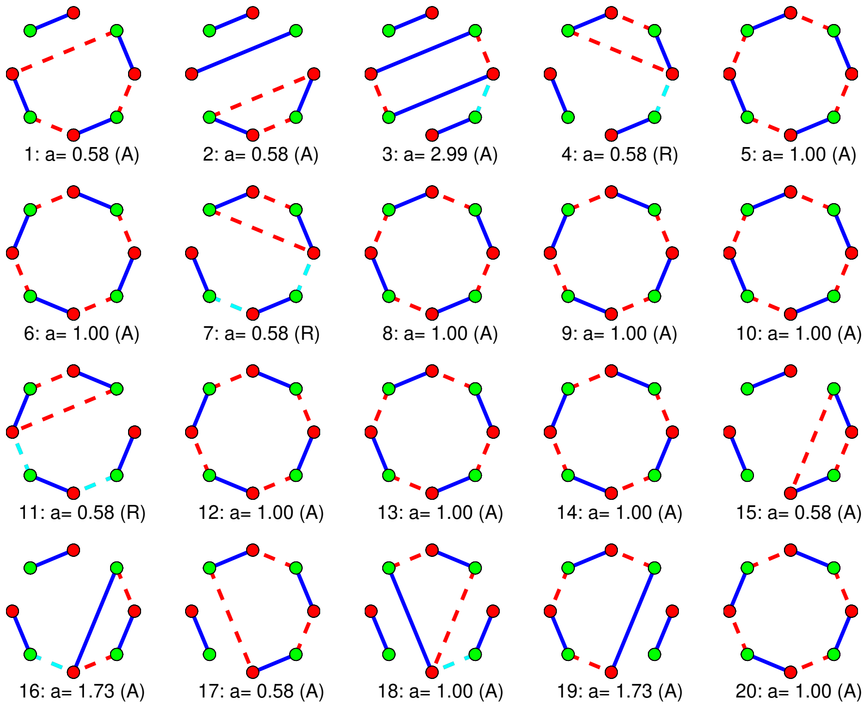

Figure 11: 20 iterations of an MCMC sampler with the "smart chain flipping" proposals. For each iteration we show a and whether the move was accepted (A) or rejected (R).

8.5 "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:

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

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 (17), as follows

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

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.

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 11 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).

9 Results for Efficient Sampling

In this section we show experimental results supporting the intuition that "smart chain flipping" leads to rapidly mixing chains. To assess the convergence of the sampler under different conditions, we use the approach discussed in (Gelman, 1996): we graph the time series for a single summary statistic in multiple, concurrently run MCMC simulations. Convergence can be assumed if all time series converge to the same value for the statistic. Displays such as this also give a qualitative understanding of the behavior of the different strategies, as we discuss in more detail below.

For Figure 12, we sample from a distribution over assignments with $n = 4$, for the configuration of features and observations as shown in Figure 11. It is clear from the latter figure that there are two globally optimal assignments, leading to a strongly bimodal distribution. In Figure 12 we show the convergence of each of the three proposal strategies discussed above, respectively from top to bottom: "flip proposals", "chain flipping", and "smart chain flipping". For each strategy, we show the results for a relatively smooth distribution ($\sigma = 0.9R$, shown at left), and a relatively peaked distribution ($\sigma = 0.5R$, shown at right). The summary statistic used is the proportion of samples that assigns observation 1 to feature 1, estimated by the average

$$\hat{J}_{11} \triangleq \frac{1}{T} \sum_{t} \delta(J^t(1), 1)$$

In the case of the low value for σ, this value is expected to be equal to 0.5, and smaller for higher values of σ. In all cases, the sampler was run for 1100 iterations, the first 100 of which were discarded as a transient.

We draw the following inferences from these figures:

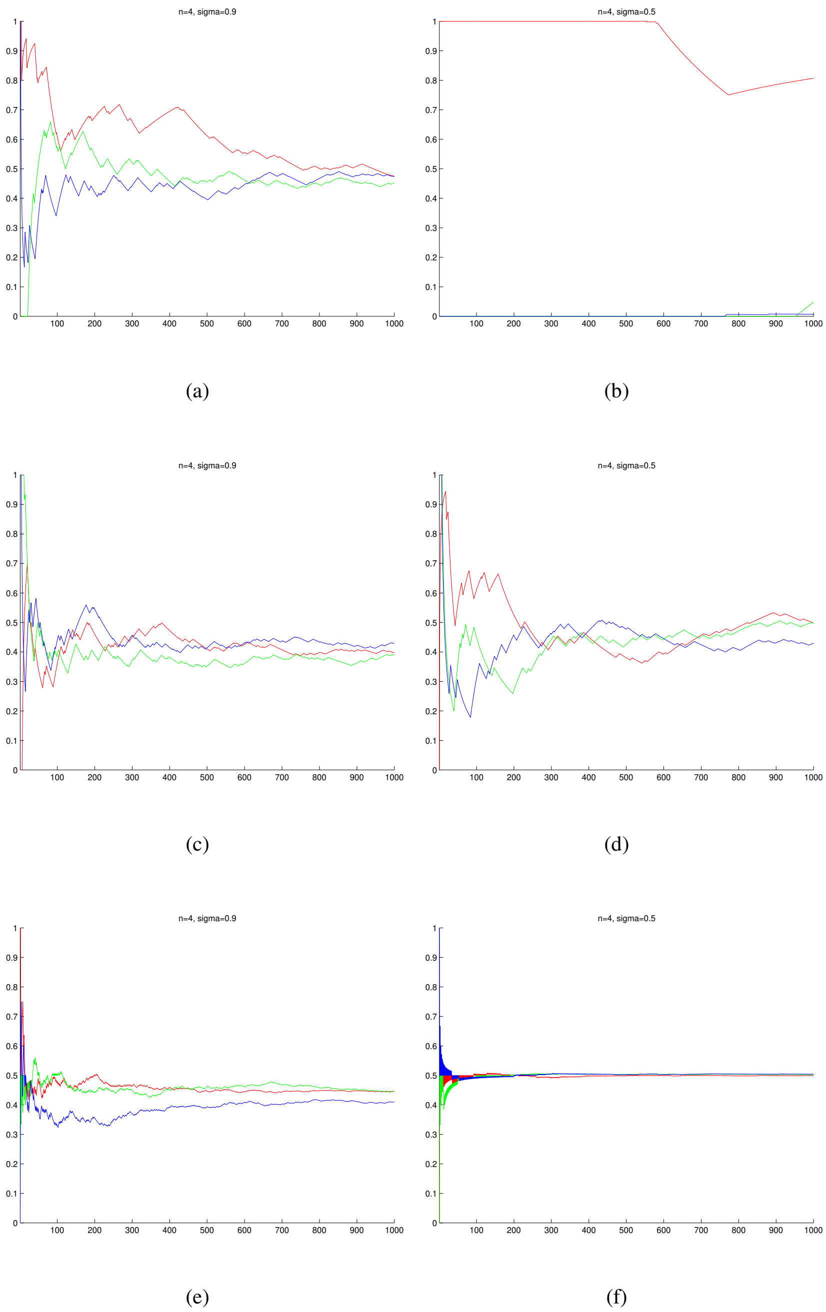

Figure 12: Assessing the convergence for the different proposal strategies. When converged, the time series for the three parallel runs in each figure should converge on the same value. See text for more explanation. (top) "Flip proposals", (middle) "chain flipping", (bottom) "Smart chain flipping". On the left, $\sigma = 0.9R$, on the right $\sigma = 0.5R$. The configuration that is being sampled over is the same as in Figure 11.

- “Flip proposals” are very slowly mixing and get stuck on high probability assignments, especially for peaked distributions (low σ). This is evident from Figure 12 (b).
- “chain flipping” leads to better mixing, but from the Figure 12 (c) and (d) it is clear that there are long stretches where the assignment is not changed much if at all.
- Dramatically better performance is obtained using “smart chain flipping”, especially for the peaked distribution on the right. The convergence to the bimodal distribution is almost immediate when compared to the other strategies. Convergence is somewhat slower for a high value of σ, as there are many more probable states that take some time to be visited often enough.

10 Related Work

In recent years, EM has become a popular algorithm for estimating models of various sorts from incomplete data. As outlined in the introduction, the issue of incomplete data and the data association problem are closely related, though not identical. Other applications, such as the Baum-Welch algorithm for learning hidden Markov model (Rabiner & Juang, 1986), do not assume a one-to-one correspondence between measurements and model parameters.

The structure from motion problem has been studied extensively in the computer vision literature over the past decade, as we have discussed in detail in Section 3. In the introduction and in Section 2 we discussed the shortcomings of the existing methods for data-association in the SFM literature.

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 & Feder, 1999), 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 & Tardós, 2000; Leonard & Durrant-Whyte, 1992; Leonard et al., 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 et al., 1999), such as nearest neighbor tracking (Deriche & Faugeras, 1990), 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, as we have assumed.

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 et al., 1998b, 1998a). 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 ML or MAP smoothing of tracks, leading to the Probabilistic Multi-Hypothesis Tracker (PMHT) (Avitzour, 1992; Streit & Luginbuhl, 1994; Gauvrit et al., 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. This is one of the main causes leading to performance that is generally less optimal than the JPDAF or MHT (Willett et al., 1999). 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 commnunity.

The new proposal strategies we propose for efficient sampling of assignments bear an interesting relation to research in the field of computational complexity theory. In particular, 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. Recently, polynomial approximation bounds have been proven for Broder-type algorithms (Jerrum & Sinclair, 1997), and there is hope that these proofs could be modified to prove hard properties for our method, as well.

11 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.

Despite the space we have devoted to explaining the rationale behind it, the final algorithm is simple and easy to implement. As summarized in Section 6, 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.

However, 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). The development of our approach does not depend on this assumption, however. We are currently extending and evaluating the approach to deal with spurious measurements, by the introduction of a NULL feature, as e.g. in (Gold et al., 1998), and with occlusion, through the development of a more sophisticated prior on assignments. The process of occlusion, however, is a systematic phenomenon which is not accurately modeled by assuming a simple independent probability of occlusion for each 3D feature. In order to deal with this in a principled manner, we are investigating the use of a Markov random field prior (Li, 1995; Winkler, 1995) on occlusion, similar in spirit to (MacCormick & Blake, 1998). It also introduces the thorny issue of model selection, as 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 of-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, Russell, Ostland, & Ritov, 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.

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. (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.

Azarbayejani, A., & Pentland, A. P. (1995). Recursive estimation of motion, structure, and focal length. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, *17*(6), 562.

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

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

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

Bertsekas, D. (1991). *Linear Network Optimization: Algorithms and Codes*. The MIT press, Cambridge, MA.

Borenstein, J., Everett, B., & Feng, L. (1996). *Navigating Mobile Robots: Systems and Techniques*. A. K. Peters, Ltd., Wellesley, MA.

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

Castellanos, J., & Tardós, J. (2000). *Mobile Robot Localization and Map Building: A Multisensor Fusion Approach*. Kluwer Academic Publishers, Boston, MA.

Cook, W., Cunningham, W., Pulleyblank, W., & Schrijver, A. (1998). *Combinatorial Optimization*. John Wiley & Sons, New York, NY.

Cooper, M., & Robson, S. (1996). Theory of close range photogrammetry. In Atkinson, K. (Ed.), *Close range photogrammetry and machine vision*, chap. 1, pp. 9–51. Whittles Publishing.

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

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

Dempster, A., Laird, N., & Rubin, D. (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. (1990). Tracking line segments. *Image and Vision Computing*, *8*, 261–270.

Doucet, A., Gordon, N., & de Freitas, J. (Eds.). (2001). *Sequential Monte Carlo Methods In Practice*. Springer-Verlag, Ney York.

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

Gelman, A. (1996). Inference and monitoring convergence. In Gilks et al. (Gilks et al., 1996), pp. 131–140.

Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, *6*(6), 721–741.

Gilks, W., Richardson, S., & Spiegelhalter, D. (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. (1958). Maximum likelihood estimation from incomplete data. *Biometrics*, *14*, 174–194.

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

Hartley, R., & Zisserman, A. (2000). *Multiple View Geometry in Computer Vision*. Cambridge University Press.

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

Irani, M. (1999). Multi-frame optical flow estimation using subspace constraints. In *Int. Conf. on Computer Vision (ICCV)*, pp. 626–633.

Jacobs, D. (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.

Jerrum, M., & Sinclair, A. (1997). The Markov chain Monte Carlo method: An approach to approximate counting and integration. In Hochbaum, D. (Ed.), *Approximation Algorithms for NP-hard Problems*, chap. 12. PWS Publishing, Boston, MA.

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

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

Leonard, J., & Durrant-Whyte, H. (1992). *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic, Boston.

Leonard, J., & Feder, H. (1999). A computationally efficient method for large-scale concurrent mapping and localization. In Hollerbach, J., & Koditschek, D. (Eds.), *Proceedings of the Ninth International Symposium on Robotics Research* Salt Lake City, Utah.

Li, S. (1995). *Markov Random Field Modeling in Computer Vision*. Springer.

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

MacCormick, J., & Blake, A. (1998). Spatial dependence in the observation of visual contours. In *ECCV*, pp. 765–781.

McLachlan, G., & Basford, K. (1988). *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, New York.

McLachlan, G., & 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.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., & Teller, E. (1953). Equations of state calculations by fast computing machine. *Journal of Chemical Physics*, *21*, 1087–1091.

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

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

Neal, R., & Hinton, G. (1998). A view of the EM algorithm that justifies incremental, sparse, and other variants. In Jordan, M. (Ed.), *Learning in Graphical Models*. Kluwer Academic Press.

Papadimitriou, C., & Steiglitz, K. (1982). *Combinatorial Optimization: Algorithms and Complexity*. 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.

Rabiner, L., & Juang, B. (1986). An introduction to hidden markov models. In *IEEE ASSP Magazine*.

Rasmussen, C., & Hager, G. (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. (1979). An algorithm for tracking multiple targets. *IEEE Trans. on Automation and Control*, *AC-24*(6), 84–90.

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

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

Shapiro, L., & Brady, J. (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., & Gelfand, A. (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. (1993). Recovering 3D shape and motion from image streams using non-linear least squares. Tech. rep. CRL 93/3, DEC Cambridge Research Lab.

Tanner, M. (1996). *Tools for Statistical Inference*. Springer Verlag, New York. Third Edition.

Taylor, C., & Kriegman, D. (1995). Structure and motion from line segments in multiple images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, *17*(11), 1021–1032.

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. (1998b). 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., 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. (1997). An assessment of information criteria for motion model selection. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 47–53.

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., & Huang, T. (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*. MIT Press, Cambridge, MA.

Weng, J., Ahuja, N., & Huang, T. (1993). Optimal motion and structure estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, *15*, 864–884.

Willett, P., Ruan, Y., & Streit, R. (1999). Making the probabilistic multi-hypothesis tracker the tracker of choice. In *Proceedings 1999 IEEE Aerospace Conference*, Vol. 4, pp. 387 –399.

Winkler, G. (1995). *Image analysis, random fields and dynamic Monte Carlo methods*. Springer Verlag.

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