Structure from Motion without Correspondence

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

December 99

CMU-RI-TR-99-44

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

We gratefully acknowledge the support of the following sponsoring institutions: the National Science Foundation (CAREER grant number IIS-9876136 and regular grant number IIS-9877033); DARPA-ATO via TACOM (contract number DAAE07-98-C-L032); DARPA-ISO via Rome Labs (contract number F30602-98-2-0137); the Federal Highway Administration via the program ”Sensor-Friendly Roadways and Vehicles”; NHTSA via the program ”Run-Off-Road Collision Countermeasures”; PennDOT via the program ”Bus Side Collision Countermeasures”; SAIC corporation; KIRIN brewery company; SONY corporation; and the Robotics Institute. The views and conclusions contained in this document are those of the authors 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.

Abstract

A method is presented to recover 3D scene structure and camera motion from multiple images without the need for correspondence information. The problem is framed as finding the maximum likelihood structure and motion given only the 2D measurements, integrating over all possible assignments of 3D features to 2D measurements. This goal is achieved by means of an algorithm which iteratively refines a probability distribution over the set of all correspondence assignments. At each iteration a new structure from motion problem is solved, using as input a set of ’virtual measurements’ derived from this probability distribution. It is shown that the distribution needed can be efficiently obtained by Monte Carlo Markov Chain sampling. The approach is cast within the framework of Expectation-Maximization, which guarantees convergence to a local maximizer of the likelihood. The algorithm works well in practice, as will be demonstrated using results on several real image sequences.

1 Introduction

A primary objective of computer vision is to enable reconstructing 3D scene geometry and camera motion from a set of images of a static scene. The current state of the art provides solutions that apply only under special conditions. Specifically, existing techniques generally assume one or more of the following:

- Known correspondence: given a set of image feature trajectories over time, solve for their 3D positions and camera motion. This classical formulation has been studied in the context of structure from motion [25, 20, 18] and, more recently, self-calibration [8, 21].
- Known cameras: given calibrated images from known camera viewpoints, solve for 3D scene shape. Stereo correspondence [6] and volumetric methods [7, 14] fit within this category.
- Known shape: given one or more images and a 3D model of the scene, determine the camera viewpoint corresponding to each image [12, 15].

The applicability of each of these methods is limited by the need for accurate correspondence, camera calibration, or shape information as input. Reliable pixel correspondence is difficult to obtain, especially over a long sequence of images. Feature-tracking techniques often fail to produce correct matches due to large motions, occlusions, or ambiguities. Furthermore, errors in one frame are likely to propagate to all subsequent frames of the sequence. Outlier rejection techniques [3, 27, 9] can ameliorate these problems, but at the cost of eliminating valid features from the reconstruction, resulting in an incomplete model that does not take into account all available image measurements. A priori knowledge of camera parameters or epipolar geometry can simplify the correspondence problem [6]. However, obtaining accurate calibrated image sequences is difficult even in controlled laboratory environments. While recent progress in self-calibration techniques [8, 21] promises to ameliorate these difficulties, these techniques require point correspondence as input and therefore are sensitive to errors due to incorrectly-tracked features. In short, existing shape recovery techniques are strongly limited by their reliance on error-prone correspondence techniques.

In this paper, we address the structure from motion problem (SFM) *without* prior knowledge of point correspondence or camera viewpoints. We frame the problem as finding the maximum likelihood estimate of structure and motion given only the measurements, integrating over all possible assignments of 3D features to 2D measurements. While the full computation of this likelihood function is generally intractable, we propose to use the Expectation-Maximization algorithm (EM) [10, 5] as a practical method for finding its maxima. We will show that EM has a simple and intuitive interpretation in this context.

The broad outline of our method is as follows: instead of solving for structure and motion given the original image measurements, we solve a new SFM problem using newly synthesized 'virtual' measurements, computed using our current knowledge about the correspondences. This knowledge comes in the form of a probability distribution, computed using the actual image data and an initial guess for the structure and motion. By solving this new SFM problem we obtain a better estimate for the structure and motion. This basic step is iterated until convergence. The virtual measurements play the role of sufficient statistics, summarizing everything we know from before about the correspondences. A key step in our method is in computing the probability distribution over correspondences, which is hard to obtain analytically. To circumvent this, we propose to use Monte Carlo Markov Chain methods to sample from this distribution, which can be done efficiently. In this respect, our approach resembles that of Forsyth et al. [9] who also applied MCMC for structure from motion, assuming known correspondence. A key difference, however, is that we solve for correspondence, structure, and motion simultaneously–a much more difficult problem.

The problem of computing structure and motion from a set of images *without* correspondence information remains largely unaddressed in the literature. Several authors considered the special case of correct but *incomplete* correspondence, by hallucinating occluded features [25, 2], or expanding a minimal correspondence into a complete correspondence [22]. 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 [13] described how geometric rank constraints can be used at a low level to facilitate optical flow computation over closely-spaced views. Beardsley et al. [3] proposed a two-phase approach for robustly computing feature correspondences in an image sequence by processing images triplets. In the first phase, a minimal point correspondence is computed from a set of candidate matches using a RANSAC-based algorithm. The results from this phase are used to compute the trifocal tensor which in turn constrain the search for other feature correspondences. Although we adopt a very different approach and do not require closely-spaced views, we follow their lead in coupling the estimation of

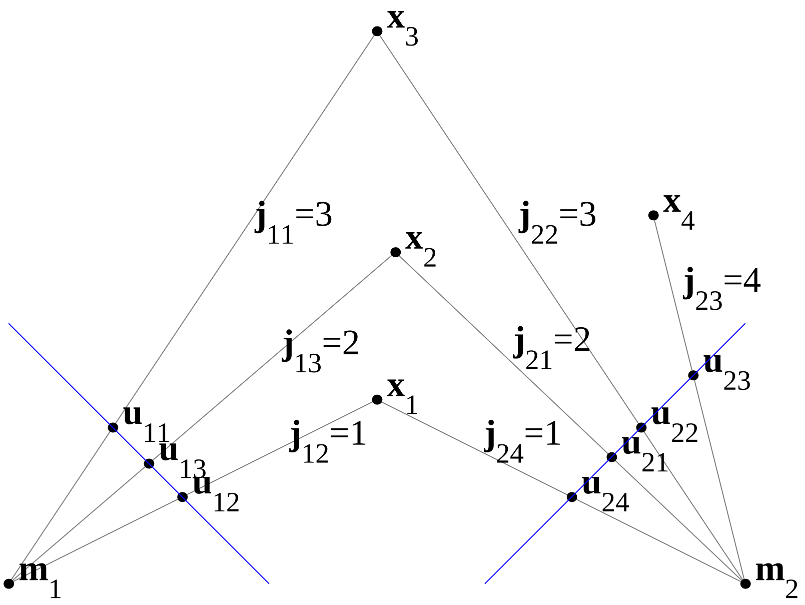


Figure 1: An example with 4 features seen in 2 images. The 7 measurements $\mathbf{u}_{ik}$ are assigned to the individual features x_j by means of the assignment variables $\mathbf{j}_{ik}$.

correspondence and structure. Rather than consider a small set of images or features at time, however, our strategy is to simultaneously optimize over all features in all images.

The remainder of this paper is structured as follows: in Section 2 we state the problem, introduce our notation, and sketch the outline of our approach. Section 3 provides the intuitive interpretation in terms of virtual measurements. In Section 4, we discuss the use of MCMC sampling to implement the E-step. Section 5 presents the results.

2 SFM without Correspondences

2.1 Problem Statement and Notation

The *structure from motion* (SFM) problem is this: given a set of images of a scene, taken from different viewpoints, recover the 3D structure of the scene along with the camera parameters. In the feature-based approach to SFM, we consider the situation in which a set of n 3D features $\mathbf{x}_j$ is viewed by a set of m cameras $\mathbf{m}_i$. As input data we are given the set of 2D measurements $\mathbf{u}_{ik}$, where $k \in \{1..K_i\}$ and K_i is the number of measurements in the i-th image. To model correspondence information, we introduce for each measurement $\mathbf{u}_{ik}$ the indicator variable $\mathbf{j}_{ik}$, indicating that $\mathbf{u}_{ik}$ is a measurement of the $\mathbf{j}_{ik}$-th feature $\mathbf{x}_{\mathbf{j}_{ik}}$. Our notation is illustrated in Figure 1.

The choice of feature type and camera model defines the *measurement function* $\mathbf{h}(\mathbf{m}_i, \mathbf{x}_j)$, predicting the measurement $\mathbf{u}_{ij}$ given $\mathbf{m}_i$ and $\mathbf{x}_j$:

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

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

$$\mathbf{h}(\mathbf{m}_i, \mathbf{x}_{\mathbf{j}_{ik}}) = \Pi_i[\mathbf{R}_i(\mathbf{x}_{\mathbf{j}_{ik}} - \mathbf{t}_i)] \tag{1}$$

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 point in 3D to the 2D image plane. Various camera models can be defined by specifying the action of this projection operator on a point $\mathbf{x} = (x, y, z)^T$ [19]. For example, the projection operators for orthography and calibrated perspective are defined as:

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

2.2 SFM with Known Correspondences

To set the stage for the rest of the paper, it is convenient to view SFM as a *maximum likelihood* (ML) estimation problem. Let us denote the set of 3D points as $\mathbf{X}$, the set of cameras as $\mathbf{M}$, the set of measurements as $\mathbf{U}$, and a set of assignments as $\mathbf{J}$. Furthermore, define $\Theta \overset{\Delta}{=} (\mathbf{X}, \mathbf{M})$. The maximum likelihood estimate $\Theta^* = (\mathbf{M}^*, \mathbf{X}^*)$ of structure and motion given the measurements $\mathbf{U}$ is given by

$$\Theta^* = \underset{\Theta}{\text{argmax}} \ \log L(\Theta; \mathbf{U}, \mathbf{J}) \tag{2}$$

where the likelihood $L(\Theta; \mathbf{U})$ of Θ given $\mathbf{U}$ is defined as any function proportional to $P(\mathbf{U}|\Theta)$ [24].

If we are given the correspondence information $\mathbf{J}$, $\log L(\Theta; \mathbf{U}, \mathbf{J})$ is easy to evaluate. 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 reprojection errors:

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

In the case of orthographic, weak- and para-perspective camera models, we find the estimate Θ^* that minimizes (3) using the *factorization* approach. Using this technique, affine structure $\mathbf{X}^a$ and motion $\mathbf{M}^a$ are first obtained from the measurements $\mathbf{U}$ by means of singular value decomposition. They are then upgraded to Euclidean structure and motion by imposing metric constraints on $\mathbf{M}^a$. This is a well developed technique, and the reader is referred to [25, 20, 18] for details and additional references.

In the case of fully perspective cameras the measurement function $\mathbf{h}(\mathbf{m}_i, \mathbf{x}_j)$ is non-linear, and we resort to non-linear optimization to minimize the re-projection error. This procedure is known in photogrammetry and computer vision as *bundle adjustment*, and details can be found in [23, 11, 4]. We use factorization to obtain an initial estimate and then use the Levenberg-Marquardt optimization method to find Θ^*. Sparse matrix techniques as discussed in [11] can be used to significantly reduce the computational cost.

2.3 SFM without Correspondences

In the case that the correspondences are unknown, the maximum likelihood estimate $\Theta^* = (\mathbf{M}^*, \mathbf{X}^*)$ of structure and motion given *only* the measurements $\mathbf{U}$ is given by:

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

Although this might seem counterintuitive at first, the above states that *we can find the ML structure and motion without explicitly reasoning about which correspondence assignment might be correct.* We ’only’ need to maximize the likelihood $L(\Theta; \mathbf{U})$, which does not depend on $\mathbf{J}$.

To gain a better understanding for what the function $L(\Theta; \mathbf{U})$ looks like, consider the example of Figure 2, where two features $\mathbf{x}_1$ and $\mathbf{x}_2$ are seen in a 1D

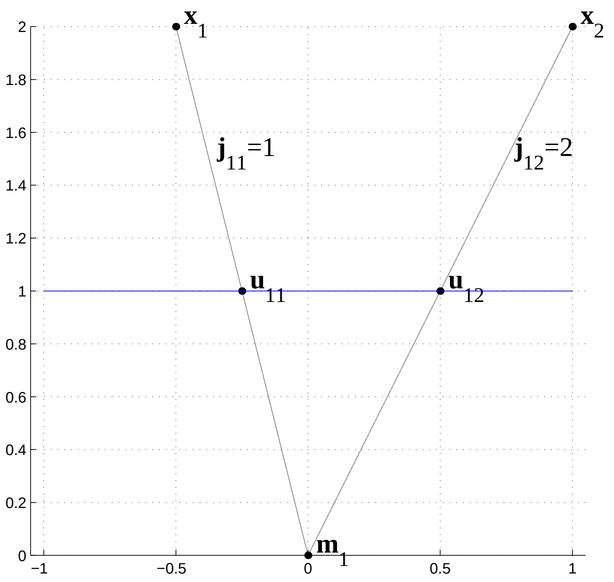


Figure 2: Example where 2 features x_1 and x_2 are seen in one image. The features are constrained to lie on the $y = 2$ line. The associated likelihoods are shown in Figure 3.

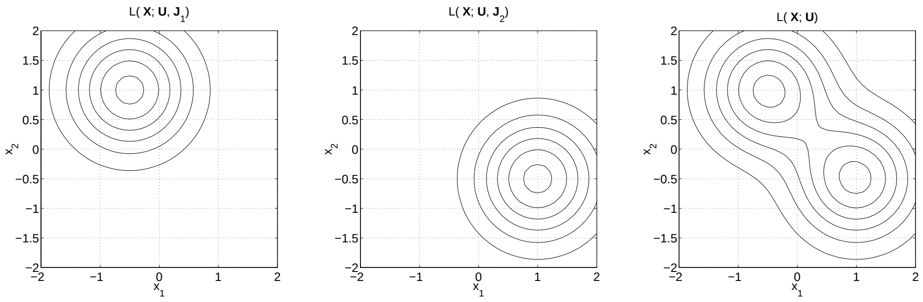


Figure 3: The joint likelihood of $\mathbf{x}_1$ and $\mathbf{x}_2$ from Figure 2 in the three cases: (left) given $\mathbf{U}$ and the 'obvious' assignment $\mathbf{J}_1$, (middle) given $\mathbf{U}$ and the 'reverse' assignment $\mathbf{J}_2$, and (right) given $\mathbf{U}$ only, which is their sum .

camera. In this case there are two measurements $\mathbf{u}_{11}$ and $\mathbf{u}_{22}$, and two possible assignments: $\mathbf{J}_1$ (shown) assigns $\mathbf{u}_{11}$ to $\mathbf{x}_1$ and $\mathbf{u}_{12}$ to $\mathbf{x}_2$, and the opposite assignment $\mathbf{J}_2$. Suppose that the camera $\mathbf{m}_1$ is known, and that the features are constrained to lie on the line $y = 2$, such that they have only one free parameter each. To calculate $L(\Theta; \mathbf{U})$, note that we can write it as a sum of likelihood terms of the form (2), with one term for every possible correspondence assignment $\mathbf{J}$:

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

The computation of $L(\Theta; \mathbf{U})$ for this example is illustrated in Figure 3: for each of the two possible assignments the likelihood is a unimodal distribution, but the *total* likelihood function $L(\Theta; \mathbf{U})$ is bimodal. This agrees with the intuition that either one of the assignments $\mathbf{J}_1$ or $\mathbf{J}_2$ is equally likely.

2.4 Maximizing the Likelihood Using EM

While the full computation of $L(\Theta; \mathbf{U})$ is generally intractable, the Expectation-Maximization [10, 5] algorithm provides a practical method for finding its maxima. In general, $L(\Theta; \mathbf{U})$ is hard to obtain explicitly, as it involves summing over a combinatorial number of possible assignments. However, it can be proven that the EM algorithm converges to a local maximum of $L(\Theta; \mathbf{U})$.

The idea of EM is to maximize the *expected* log likelihood function

$$Q^t(\Theta) \triangleq E_{f^t}\{\log L(\Theta; \mathbf{U}, \mathbf{J})\}$$

where the expectation is taken with respect to the posterior distribution $f^t \triangleq P(\mathbf{J}|\mathbf{U}, \Theta^t)$ over all possible assignments **J** given the data **U** and a current guess Θ^t for structure and motion. The EM algorithm then iterates over [24]:

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

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

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 guess Θ^{t+1}.

3 SFM with Virtual Measurements

In this section we show that the EM algorithm outlined above can be interpreted 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 similar SFM problem, but using as input a newly synthesized set of virtual measurements, created in the E-step.

3.1 Virtual Measurements

In the context of SFM, we substitute the expression for the log likelihood $\log L(.)$ from (3) in equation (5), and obtain the following expression for $Q^t(\Theta)$:

$$\frac{1}{2\sigma^2} \sum_{\mathbf{J}} f^t(\mathbf{J}) \sum_{i=1}^{m} \sum_{k=1}^{K_i} \|\mathbf{u}_{ik} - \mathbf{h}(\mathbf{m}_i, \mathbf{x}_{\mathbf{j}_{ik}})\|^2 \tag{6}$$

It is clear that a direct evaluation of (6) is infeasible, as the number of possible assignments is combinatorial in n. An efficient implementation is nevertheless possible.

To see this, let us first calculate the probability f^t_{ijk} that a measurement $\mathbf{u}_{ik}$ in image i is assigned to a feature $\mathbf{x}_j$, regardless of how the other measurements are assigned. In other words, f^t_{ijk} is the *marginal posterior probability* $P(\mathbf{j}_{ik} = j|\mathbf{U}, \Theta^t)$, and it can be calculated by summing $f^t(\mathbf{J})$ over all possible assignments $\mathbf{J}$ where $\mathbf{j}_{ik} = j$:

$$f^t_{ijk} \stackrel{\Delta}{=} P(\mathbf{j}_{ik} = j|\mathbf{U}, \Theta^t) = \sum_{\mathbf{J}} \delta(\mathbf{j}_{ik}, j) f^t(\mathbf{J}) \tag{7}$$

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

Equation (7) allows us to rewrite the expected log-likelihood $Q^t(\Theta)$ from (6) in a form that only depends indirectly on the assignment variables $\mathbf{j}_{ik}$:

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

Now we state the main result in this section: 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

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

where the *virtual measurements* $\mathbf{v}^t_{ij}$ and *virtual measurement variance* $(\sigma^t_{ij})^2$ are defined as

$$\mathbf{v}^t_{ij} \stackrel{\Delta}{=} \frac{\sum_{k=1}^{K_i} f^t_{ijk} \mathbf{u}_{ik}}{\sum_{k=1}^{K_i} f^t_{ijk}}, \quad (\sigma^t_{ij})^2 \stackrel{\Delta}{=} \frac{\sigma^2}{\sum_{k=1}^{K_i} f^t_{ijk}} \tag{10}$$

Each virtual measurements $\mathbf{v}^t_{ij}$ 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}. If there is no occlusion and all features are seen in all images, then $\sum_{k=1}^{K_i} f^t_{ijk} = 1$ and the expressions further simplify.

3.2 Summary and Implementation Outline

Writing $Q^t(\Theta)$ as a re-projection error with respect to virtual measurements as in equation (9) 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:** Find the structure and motion estimate Θ^{t+1} that minimizes the (weighted) re-projection error given the virtual measurements:

$$\Theta^{t+1} = \underset{\Theta}{\text{argmin}} \sum_{i=1}^{m}\sum_{j=1}^{n} \frac{1}{2(\sigma^t_{ij})^2} \|\mathbf{v}^t_{ij} - \mathbf{h}(\mathbf{m}_i, \mathbf{x}_j)\|^2$$

In other words, the E-step synthesizes new measurement data, and *the M-step is a conventional SFM problem of the same size as before. This means that we can use any known SFM algorithm at our disposal as is,* from straightforward orthographic factorization to the more recent algorithms that work with uncalibrated images. What is left is to show how the E-step can be implemented.

4 Implementing the E-step

Since the M-step can be implemented using known SFM approaches, we need only concern ourselves with the implementation of the E-step. In particular, we need to calculate the marginal probabilities $f^t_{ijk} = P(\mathbf{j}_{ik} = j|\mathbf{U}, \Theta^t)$.

4.1 Conditional Independence vs. the Mutual Exclusion Constraint

If we assume that the feature assignments $\mathbf{j}_{ik}$ *are conditionally independent given* $\mathbf{U}$ *and* Θ^t, we can write the probability of each assignment $\mathbf{J}$ as the product of the probabilities of the individual assignments $\mathbf{j}_{ik}$:

$$P(\mathbf{J}|\mathbf{U}, \Theta^t) = \prod_{i=1}^{m}\prod_{k=1}^{K_i} P(\mathbf{j}_{ik}|\mathbf{u}_{ik}, \Theta^t) \tag{11}$$

If this were an good approximation it would lead to an efficient implementation. It can be shown that in that case the marginal probabilities f^t_{ijk} only depend on the distance from the measurement $\mathbf{u}_{ik}$ to the projected feature point:

$$f^t_{ijk} = C^t_{ik} \exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{u}_{ik} - \mathbf{h}(\mathbf{m}_i, \mathbf{x}_j)\|^2\right\} \tag{12}$$

with C^t_{ik} a normalization constant.

However, *in reality the assignments are not independent*: if a measurement $\mathbf{u}_{ik}$ has been assigned $\mathbf{j}_{ik} = j$, then no other measurement in the same image should be assigned the same feature point $\mathbf{x}_j$. The probability of such a double assignment is zero, which cannot be modeled by the expression above. In other words, it does not take into account the important global constraint of mutual exclusion, which is crucial in practice to obtain good results.

Imposing the mutual exclusion constraint, however, makes it difficult to analytically express the weights f^t_{ijk}. Conditional independence no longer holds, while the simple expression (12) for the weights f^t_{ijk} relies crucially on this assumption. We know of no efficient closed form expression for f^t_{ijk} that allows only permutations.

The solution we propose is to instead *sample* from the posterior probability distribution $f^t(\mathbf{J})$ over valid assignments $\mathbf{J}$, to obtain approximate values for the weights f^t_{ijk}. 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 [24, 17]. The sample can be efficiently obtained using the Metropolis algorithm, as will be described below.

4.2 Sampling the Correspondence Distribution

We would like to obtain a sample from the distribution f^t over *valid* assignments $\mathbf{J}$. First, an important fact to note is that we can do this for each image individually, as the assignment $\mathbf{J}_i$ within each image i is conditionally independent of the assignments in other images. This means that f^t can be factored as:

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

where $\mathbf{U}_i$ are the measurements in image i only.

To sample from $P(\mathbf{J}_i|\mathbf{U}_i, \Theta^t)$ we use the Metropolis algorithm, an instance of the Monte-Carlo Markov-Chain methods (abbreviated MCMC), which involve a Markov chain in which a sequence of samples is generated [16, 9]. If we set up the transition probabilities correctly, the equilibrium distribution of the Markov chain will be equal to the posterior distribution we would like to sample from. In our case, we would like to generate a sequence of samples $\mathbf{J}^r_i$ from the posterior $P(\mathbf{J}_i|\mathbf{U}_i, \Theta^t)$, and the Metropolis algorithm can be formulated in the current context as follows (adapted from the general description in [16]):

1. Start with a valid initial assignment $\mathbf{J}_i^0$.

2. Propose a new valid assignment $\mathbf{J}_i'$, which is probabilistically generated from $\mathbf{J}_i^r$.

3. Compute the ratio

$$a = \frac{P(\mathbf{J}_i' | \mathbf{U}_i, \Theta^t)}{P(\mathbf{J}_i^r | \mathbf{U}_i, \Theta^t)} \tag{13}$$

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 $\mathbf{J}_i^{r+1} = \mathbf{J}_i^r$.

To actually implement this scheme, we need to specify three elements: (a) define what a 'valid' assignment is, (b) a way to probabilistically perturb them, and (c) an explicit expression for a. Below we do this for the case when there is no occlusion, no spurious features, and all features are seen in all images. In this case, the only *valid assignments* $\mathbf{j}_{ik}$ are permutations of the feature indices $1..n$. The *proposal step* can be implemented by swapping the assignment variables $\mathbf{j}_{ik}$ of two randomly chosen measurements $\mathbf{u}_{ik}$, which conserves the permutation property. Finally, the posterior ratio a can be evaluated very efficiently, as it can be shown to depend only on the dot product of two vectors related to the swap (proof omitted):

$$a = \exp(\frac{1}{\sigma^2}(\mathbf{u}_1 - \mathbf{u}_2)^T(\mathbf{h}_2 - \mathbf{h}_1))$$

where $\mathbf{u}_1$ and $\mathbf{u}_2$ the measurements whose assignments will be swapped, and $\mathbf{h}_1$ and $\mathbf{h}_2$ are the projections of the features originally assigned to them .

To conclude the E-step and compute the virtual measurements in (10), the only thing left to do is 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 of how many times each measurement $\mathbf{u}_{ik}$ is assigned to feature j, and use that to compute f_{ijk}^t. If we define C_{ijk}^t to be this count, we have:

$$f_{ijk}^t \approx \frac{1}{R} C_{ijk}^t \tag{14}$$

4.3 Implementation 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 sampler in each image to obtain approximate values for the weights f^t_{ijk}, using equation (14).

3. Calculate the virtual measurements $\mathbf{v}^t_{ij}$ with (10).

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 compatible with the projection model assumed.

5. If not converged, return to step 2.

To avoid getting stuck in local minima, it is important in practice to add *annealing* to this basic scheme. In annealing we artificially increase the noise parameter σ for the early iterations, gradually decreasing it to its correct value. This has two beneficial consequences. First, the posterior distribution f^t will be less peaked when σ is high, so that the Metropolis sampler will explore the space of assignments more easily, and avoid getting stuck on islands of high probability. Second, the expected log likelihood $Q^t(\Theta)$ is smoother and has less local maxima at higher values for σ. We use a logarithmically decreasing annealing scheme, but have found that the algorithm is not sensitive to the exact scheme used.

5 Results

In this section we show the results obtained with three different sets of images. 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. We ran the EM algorithm for 100 iterations each time, with the annealing parameter σ decreasing logarithmically from 25 pixels to 1 pixel. For each EM iteration, we ran the sampler in each image for 10000 steps. For the image sets below it takes about a minute to run 100 iterations on a standard PC.

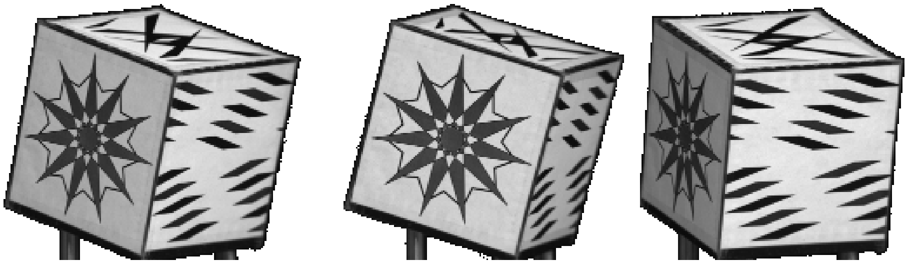
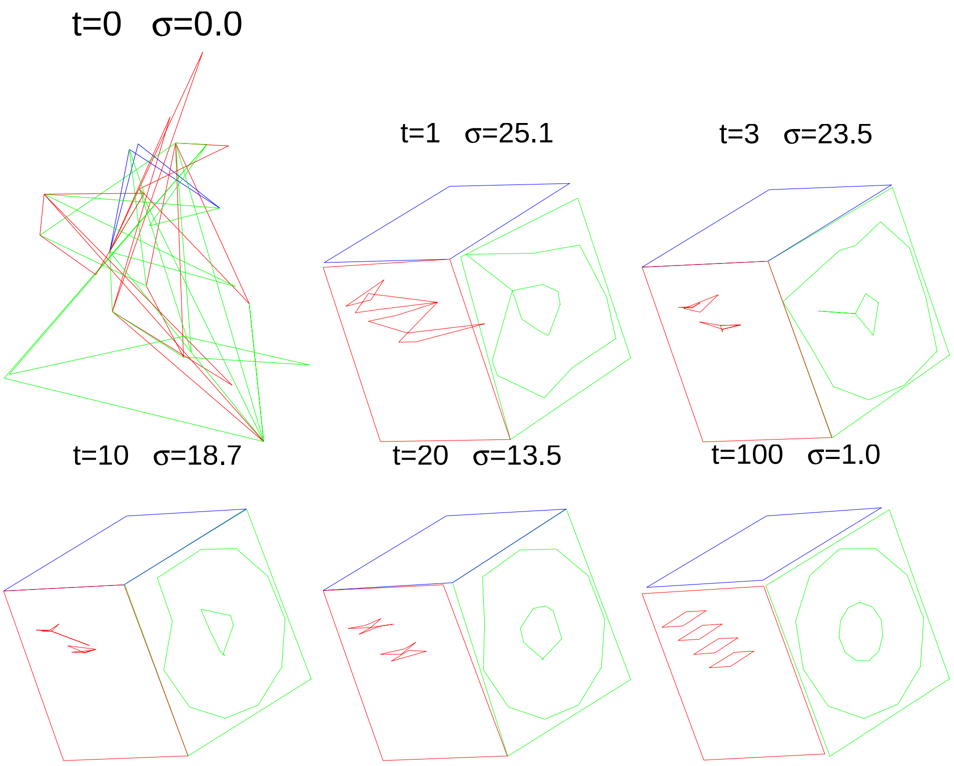


Figure 4: Three out of 11 *cube* images. Although the images were originally taken as a sequence in time, the ordering of the images is irrelevant to our method.



t=0 σ=0.0

t=1 σ=25.1

t=3 σ=23.5

t=10 σ=18.7

t=20 σ=13.5

t=100 σ=1.0

Figure 5: The structure estimate a s initialized and at successive iterations t of the algorithm.

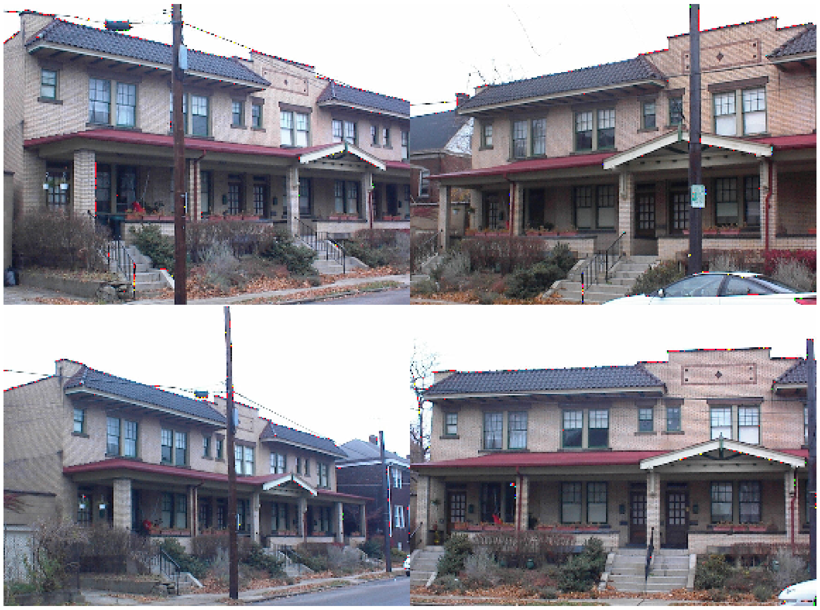


Figure 6: 4 out of 5 perspective images of a house.

In practice, the algorithm converges consistently and fast to an estimate for the structure and motion where the correct correspondence 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 4, which was taken under orthographic projection. The typical evolution of the algorithm is illustrated in Figure 5, where we have shown a wireframe 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 correspondence. 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 6 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

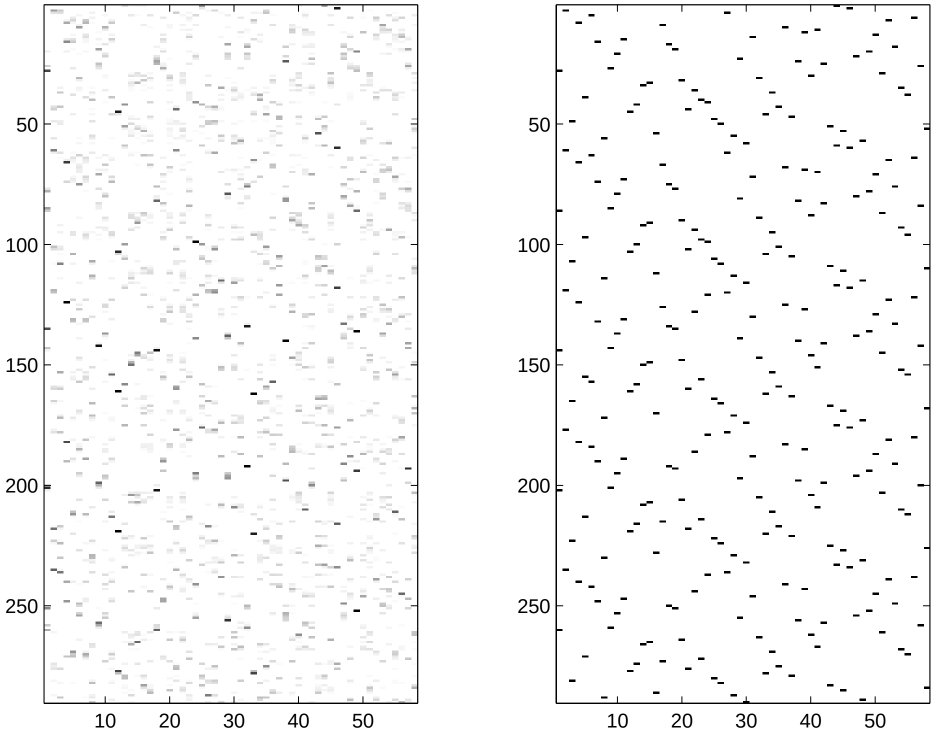


Figure 7: The marginal probabilities f_{ijk}^{t} at an early and at a later iteration, respectively. Each row corresponds to a measurement $\mathbf{u}_{ik}$, grouped according to image index, whereas the columns represent the n features $\mathbf{x}_j$. In this example $n = 58$ and $m = 5$. Black corresponds to a marginal probability of 1.

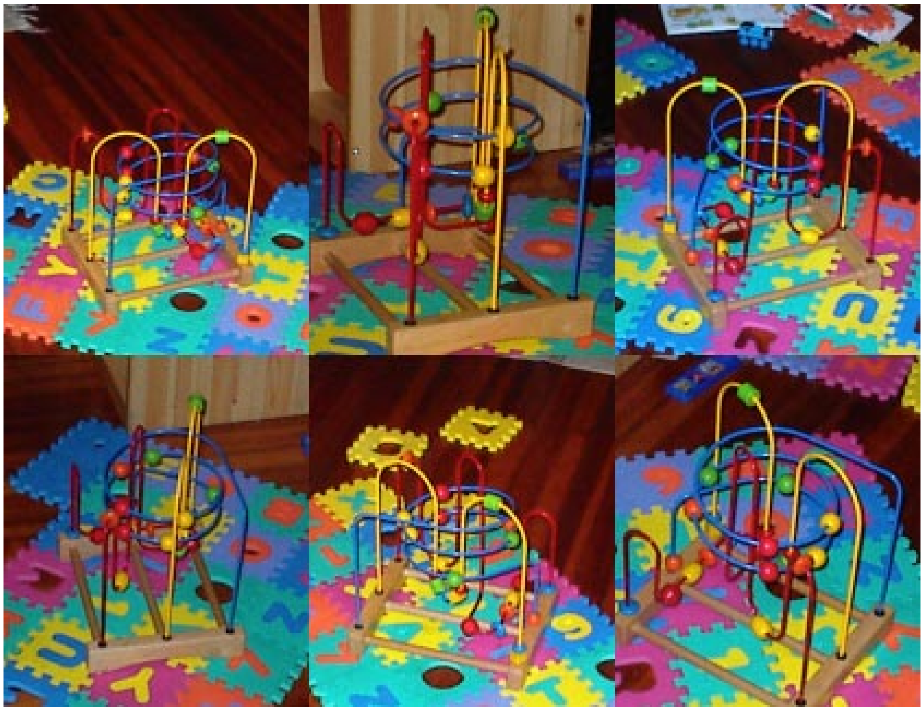


Figure 8: 6 out of 8 images of a wireframe toy, taken from widely different viewpoints.

probabilities f_{ijk}^t at two different times during the course of the algorithm, in Figure 7. 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_{ijk}^t matrix would be a set of identical permutation matrices stacked one upon the other.

The algorithm also deals with situations where the images are taken from widely separate viewpoints, as is the case for the images in Figure 8. In this sequence, the image features used were the colored beads on the wireframe 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 9 shows the wireframe 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

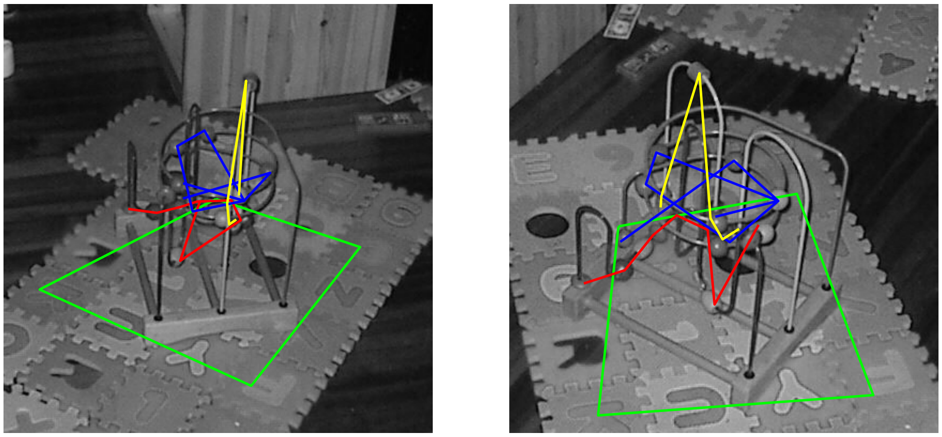


Figure 9: Recovered structure for wireframe toy reprojected in 2 images.

between images about the most likely feature assignment, the overall structure of the model is recovered despite the arbitrary configuration of the cameras.

6 Conclusions and Future Directions

In this paper we have presented a novel tool, which enables us to solve the structure from motion problem *without* a priori correspondence information. In addition, it can cope with images given in arbitrary order and taken from widely separate viewpoints.

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 4.3, 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, it is fast: the Metropolis sampler, which is the main computational bottleneck, can be implemented very efficiently due to the incremental computation of the posterior ratios, and the fact that we do not need to store the samples.

However, there is plenty of opportunity for future work. Although the general algorithm can in principle handle occlusions and spurious features, this needs to be implemented and experimentally verified. Furthermore, this introduces the issue of how many features need to be instantiated, if this is not known a priori. This issue of model selection has been addressed successfully before in [1, 26], and it is hoped that the lessons learned there can equally apply in this context.

References

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

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

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

[4] M.A.R. Cooper and S. Robson. Theory of close range photogrammetry. In K.B. Atkinson, editor, *Close range photogrammetry and machine vision*, chapter 1, pages 9–51. Whittles Publishing, 1996.

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

[6] O.D. Faugeras. *Three-dimensional computer vision: A geometric viewpoint*. The MIT press, Cambridge, MA, 1993.

[7] O.D. Faugeras and R. Keriven. Complete dense stereovision using level set methods. In *Proc. 5th European Conf. on Computer Vision*, pages 379–393, 1998.

[8] O.D. Faugeras, Q.T. Luong, and S.J. Maybank. Camera self-calibration: Theory and experiments. In *Eur. Conf. on Computer Vision (ECCV)*, pages 321–334, 1992.

[9] D.A. Forsyth, S. Ioffe, and J. Haddon. Bayesian structure from motion. In *Int. Conf. on Computer Vision (ICCV)*, pages 660–665, 1999.

[10] H.O. Hartley. Maximum likelihood estimation from incomplete data. *Biometrics*, 14:174–194, 1958.

[11] R.I. Hartley. Euclidean reconstruction from uncalibrated views. In *Application of Invariance in Computer Vision*, pages 237–256, 1994.

[12] D.P. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *Int. J. of Computer Vision*, 5(2):195–212, November 1990.

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

[14] K.N. Kutulakos and S.M. Seitz. A theory of shape by space carving. In *Proc. Sevent Int. Conf. on Computer Vision*, pages 307–314, 1999.

[15] D.G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(5):441–450, May 1991.

[16] D.J.C. MacKay. Introduction to Monte Carlo methods. In M.I.Jordan, editor, *Learning in graphical models*, pages 175–204. MIT Press, 1999.

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

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

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

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

[21] M. Pollefeys, R. Koch, and L. VanGool. Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. *Int. J. of Computer Vision*, 32(1):7–25, August 1999.

[22] S.M. Seitz and C.R. Dyer. Complete structure from four point correspondences. In *Proc. Fifth Int. Conf. on Computer Vision*, pages 330–337, 1995.

[23] R. Szeliski and S.B. Kang. Recovering 3d shape and motion from image streams using non-linear least squares. Technical Report CRL 93/3, DEC Cambridge Research Lab, 1993.

[24] M.A. Tanner. *Tools for Statistical Inference*. Springer, 1996.

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

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

[27] Z.Y. Zhang. Determining the epipolar geometry and its uncertainty - a review. *Int. J. of Computer Vision*, 27(2):161–195, March 1998.