Paper number 648

Bayesian structure from motion

D.A. Forsyth S. Ioffe J. Haddon

Computer Science Division
U.C. Berkeley
Berkeley, CA 94720
{daf,ioffe,haddon}@cs.berkeley.edu

Abstract: *We formulate structure from motion as a Bayesian inference problem, and use a Markov chain Monte Carlo sampler to sample the posterior on this problem. This results in a method that can identify both small and large tracker errors, and yields reconstructions that are stable in the presence of these errors. Furthermore, the method gives detailed information on the range of ambiguities in structure given a particular dataset, and requires no special geometric formulation to cope with degenerate situations. Motion segmentation is obtained by a layer of discrete variables associating a point with an object. We demonstrate a sampler that successfully samples an approximation to the marginal on this domain, producing a relatively unambiguous segmentation.*

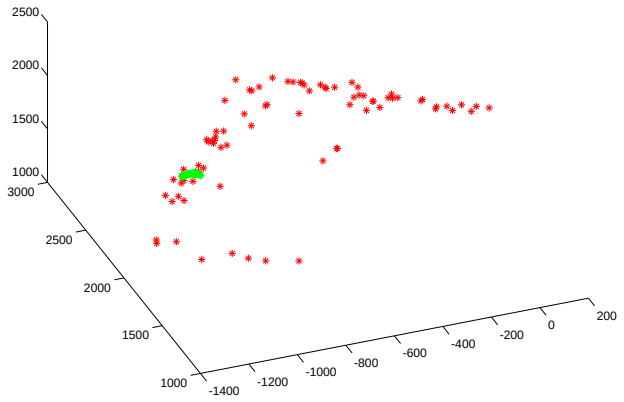

Figure 1: Red points show an overhead view of a single sample of the 3D reconstruction obtained using 40 frames of 80 points in the hotel sequence, rotated by hand to show the right-angled structure in the model indicating that the structure is qualitatively correct; the cloud of green points are samples of the position of a single point, scaled by 1000 to show the (very small) uncertainty available in a single point measurement.

1 Introduction

The Bayesian philosophy is that all information about a model is captured by a *posterior* distribution obtained using Bayes' rule:

$$\text{posterior} \propto (\text{likelihood} \times \text{prior}) = \text{joint}$$

where the likelihood is $P(\text{observations}|\text{world})$ and the prior, $\pi(\text{world})$ is the probability density of the state of the world in the absence of observations. *When computational difficulties can be sidestepped*, the Bayesian philosophy leads to excellent and effective use of data (e.g. [3, 8]), but computing with posteriors is difficult.

A standard method is to represent the posterior by drawing a large number of samples from that distribution. For example, if we wished to decide whether to fight or flee, we would draw samples from the posterior and estimate expected utilities for each decision (as sums of the utilities over the samples) and then choose the best. Sampling algorithms are more general than random search for MAP interpretations precisely because the results give an approximate representation of the entire posterior. In this paper we use a sampling method to compute a Bayesian solution to a structure from motion problem. The method is able to identify and discount tracker errors that significantly affect the performance of factorisation methods. The sampler works on an extremely large domain.

Notation: we write $\mathbf{v}$ for a vector, whose i'th component is v_i and $\mathcal{M}$ for a matrix whose i, j'th component is M_{ij}. Sampler jargon that may be unfamiliar is shown in

italics when first introduced.

1.1 Markov chain Monte Carlo

Markov chain Monte Carlo methods [7, 12] are the standard methods for sampling complex distributions. Assume we wish to draw a sample with density function $p(X)$. A typical algorithm is the Metropolis-Hastings algorithm, which would produce in this case a sequence of samples, by taking a sample X_i and using a stochastic proposal process to suggest a revised version, X_i'. The proposal process has density function $g(X_i, X_i')$ — i.e. the proposal density for the new state can depend on the old state. We now compute

$$\alpha = min(1, \frac{p(X_i')g(X_i', X_i)}{p(X_i)g(X_i, X_i')})$$

and the new sample X_{i+1} is chosen to be X_i' with probability α and X_i otherwise. Because the acceptance probability is a ratio, we can use unnormalised posteriors, which is often extremely convenient.

Assuming technical conditions on the proposal process (e.g. [7, 11, 22, 23]), once sufficient iterations have completed, all subsequent X_i are samples drawn from $p(X)$; the number of iterations required to achieve this is often called the *burn in* time, and $p(X)$ is known as the *stationary distribution*. These samples may or may not be correlated; if this correlation is low, the method is said to *mix* well. It is desirable to have an algorithm that burns in quickly, and mixes well. Sampling has occasionally been used in recognition-like applications (e.g. [1, 13, 20, 27] — the idea dates to at least [14]), but is more commonly seen in early vision applications (particularly segmentation using MRF's [17]; the usual application is to obtain an MAP estimate). Gibbs samplers are quite widely used for reconstruction [9, 10, 28]. The most substantial impact of sampling algorithms in vision has been the use of resampling algorithms in tracking. The best known algorithm is known as CONDENSATION in the vision community [2] and *survival of the fittest* in the AI community [15]. A natural variant on this algorithm is to represent the intermediate stages by an approximation derived from the samples (e.g. [16]).

2 Maximum likelihood methods for structure from motion

Accurate solutions to structure from motion are attractive, because the technique can be used to generate models for rendering virtual environments (e.g. [5, 24]). Assume m distinct views of n points are given. In the influential Tomasi-Kanade formulation of structure from motion,

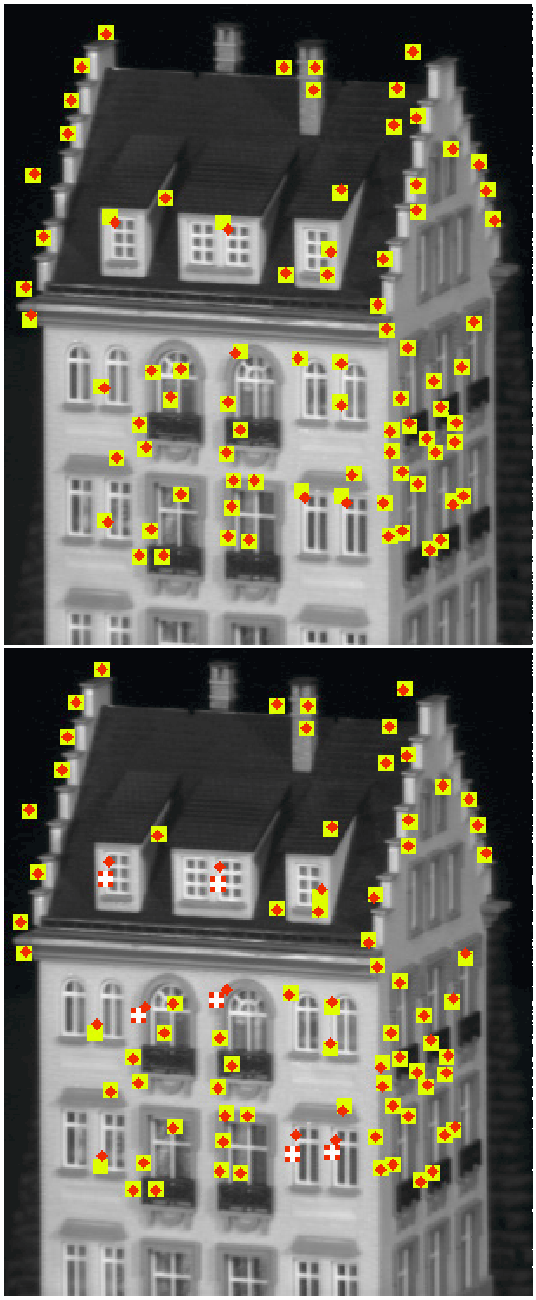

Figure 2: Two (cropped) frames from the hotel sequence showing a single sample reconstruction. Yellow squares correspond to measurements with mask bit one (i.e. the measurement of that point in that frame is believed correct); a white cross on a red background corresponds to measurements with mask bit zero (i.e. the measurement of that point in that frame is believed incorrect); red diamonds correspond to model predictions. In the lower frame, at several locations the tracker has skipped to another feature for unknown reasons. In each case the reconstruction identifies the data point as being erroneous, and reprojects to a point in a significantly different position from the (red) measurement reported by the tracker and lying where a correct measurement would be as seen by the position relative to the surface texture on the object.

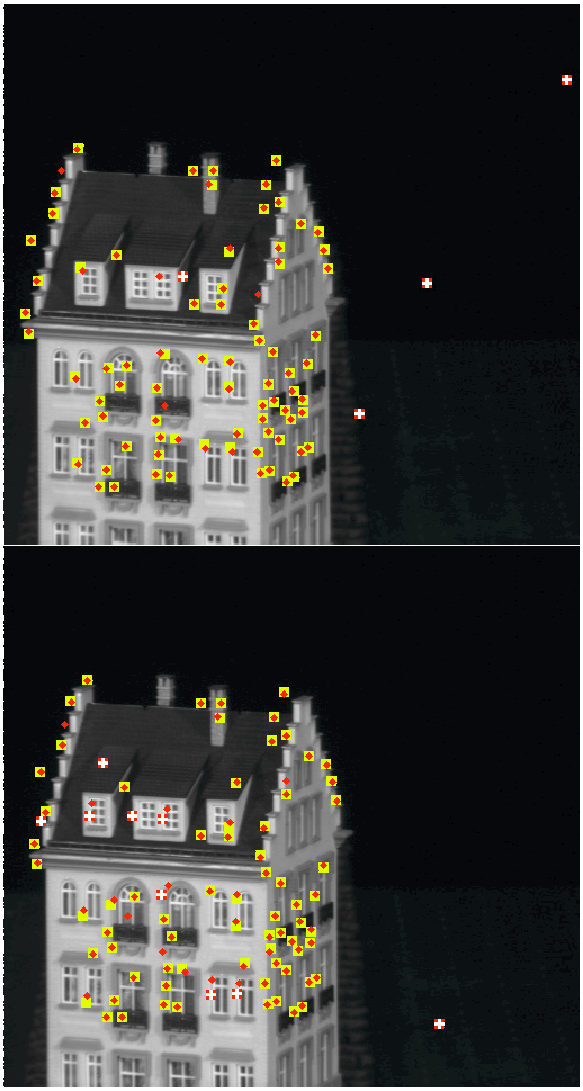

Figure 3: We perturb the hotel sequence by replacing 5% of the data points with draws from a uniform distribution in the image plane. The Bayesian method, started as in section 3.1.1, easily discounts these noise points; the figure shows the same frames in the sequence as in figure 2, uncropped to show the noise but with a sample reconstruction indicated using the same notation as that figure.

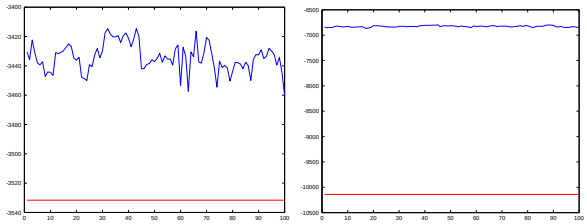

Figure 4: **Left:** the log-posterior for each of 100 samples taken after the sampler has burnt-in (blue) compared with the log-likelihood of the start point (red), for the hotel data set. **Right:** the same plot, but now for the hotel data set with 5% of the points replaced with a draw from a uniform distribution on the image plane. The smaller log-posterior is explained by the larger number of points that must be assigned to noise.

these data are arranged into a $2m \times n$ matrix of measurements $\mathcal{D}$ which must factor as $\mathcal{D} = \mathcal{U}\mathcal{V}$, where $\mathcal{U}$ represents the camera positions and $\mathcal{V}$ represents point positions. An affine transform $\mathcal{A}$ is determined such that $\mathcal{U}\mathcal{A}$ minimises a set of constraints associated with a camera, and $\mathcal{A}^{-1}\mathcal{V}$ then represents structure. In practice, factorisation is achieved using a singular value decomposition. This is a maximum likelihood method if an isotropic Gaussian error model is adopted; for an anisotropic Gaussian error model, see [18]. The formalism has been applied to various camera models [21, 24, 26]; missing data points can be interpolated from known points [24]; methods for motion segmentation exist [4]; and methods for lines and similar primitives are known [18]. There are noise estimates for recovered structure [18], but these assume that errors in the estimates of structure are independent.

Maximum likelihood methods ignore the value of the prior. For structure from motion, this means the method does not allow any payoff between model error — the extent to which the recovered model violates the required set of camera constraints — and measurement error — the extent to which model predictions correspond to data observations. This means that the model cannot be used to identify measurement problems and so is subject to reconstruction errors caused by incorporating erroneous measurements. This is a significant difficulty. Serious tracker errors lead to wildly inaccurate reconstructions, because the singular value decomposition of a matrix can change sharply with small changes in the entries.

3 Bayesian structure from motion

It is useful to think of Bayesian models as generative models (e.g. [14]). In a generative structure from motion model, $\mathcal{U}$ and $\mathcal{V}$ are drawn from appropriate priors. Then $\mathcal{D}$ is obtained by adding noise to $\mathcal{U}\mathcal{V}$. We assume that noise is obtained from a mixture model; with some large proba-

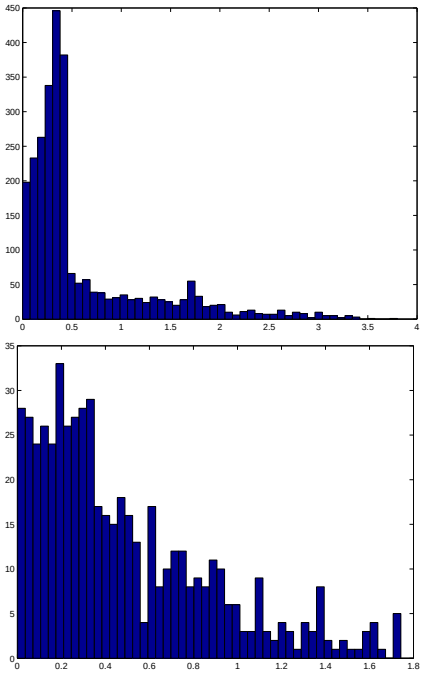

Figure 5: The factorisation method is relatively unstable under noise. We compare reconstructions obtained from the uncorrupted data set with reconstructions obtained when 5% of the entries in $\mathcal{D}$ are replaced with draws from a uniform distribution in the image plane; to represent the factorisation method fairly, we use the start points obtained using the algorithm of section 3.1.1 (which mask off suspect measurements). **Top** shows a histogram of relative variations in distances between corresponding pairs of points and **bottom** shows a histogram of differences in angles subtended by corresponding triples of points. Note the scales — some interpoint distances are misestimated by a factor of 3, and some angles are out by $\pi/2$.

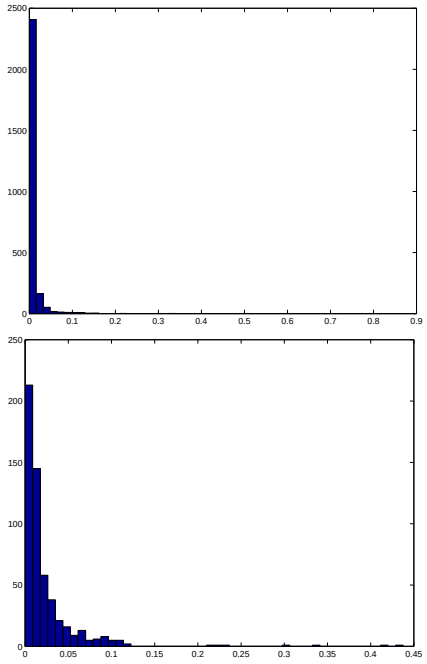

Figure 6: The Bayesian method is stable under noise. We compare reconstructions obtained from the uncorrupted data set with reconstructions obtained when 5% of the entries in $\mathcal{D}$ are replaced with draws from a uniform distribution in the image plane. **Top** shows a histogram of relative variations in distances between corresponding pairs of points and **bottom** shows a histogram of differences in angles subtended by corresponding triples of points. Note the significant increase in stability over the factorisation method; relative errors in distance are now of the order of 10% and angular errors are of the order of $\pi/40$.

bility, Gaussian noise is used, and with a small probability, the measurement value is replaced with a uniform random variable.

The priors on $\mathcal{U}$ and $\mathcal{V}$ are obtained from constraints on camera structure. Tomasi and Kanade set the origin of the point coordinate system at the center of gravity of the points, and so their $\mathcal{U}$ and $\mathcal{V}$ have dimension $2m \times 3$ and $3 \times n$ respectively. We do not fix the origin of the coordinate system, and represent points in homogenous coordinates, so our $\mathcal{U}$ and $\mathcal{V}$ have dimensions $2m \times 4$ and $4 \times n$ respectively. We assume a scaled orthographic viewing model with unknown scale that varies from frame to frame. This yields a set of constraint equations

$$\mathbf{C}(\mathcal{U}, \mathcal{V}) = \mathbf{0}$$

which contains elements of the form

$$\sum_{j=1}^{3} (u_{i,j})^2 - \sum_{j=1}^{3} (u_{i+m,j})^2$$

(expressing the fact that the camera basis consists of elements of the same length),

$$\sum_{j=1}^{3} (u_{i,j} u_{i+m,j})$$

(expressing the fact that the camera basis elements are perpendicular), and

$$v_{j,4} - 1$$

(from the homogenous coordinates). A natural prior to use is proportional to

$$\exp\left(\frac{-\mathbf{C}^T(\mathcal{U}, \mathcal{V})\mathbf{C}(\mathcal{U}, \mathcal{V})}{2\sigma_{\text{constraint}^2}}\right)$$

This prior penalises violations of the constraints quite strongly, but allows constraint violations to be paid off one against the other. An alternative is to insist that the prior is uniform if the constraints are all satisfied and zero otherwise; this approach is numerically more complex to implement.

We can now write a posterior model. We introduce a set of discrete mask bits, one per measurement, in a matrix $\mathcal{M}$; these mask bits determine by which noise model a measurement is affected. A mask bit will be 1 for a "good" measurement (i.e. one affected by isotropic Gaussian noise), and 0 for a "bad" measurement (i.e. one which contains no information about the model). We introduce a prior on $\mathcal{M}$, $\pi(\mathcal{M})$, which is zero for matrices that have fewer than k non-zero elements in some row or column, and uniform otherwise; this prior ensures that we do not attempt inference for situations where we have insufficient measurements.

The likelihood $P(\mathcal{D}|\mathcal{U},\mathcal{V},\mathcal{M})$ is then proportional to

$$\exp\left(-\left\{\sum_{i,j}\frac{(d_{ij}-\sum_l u_{il}v_{lj})^2 m_{ij}}{2\sigma^2_{meas}}+w_{noise}(1-m_{ij})\right\}\right) \quad (1)$$

and the posterior is proportional to:

$$P(\mathcal{D}|\mathcal{U},\mathcal{V},\mathcal{M})\exp\left(\frac{-\mathbf{C}^T(\mathcal{U},\mathcal{V})\mathbf{C}(\mathcal{U},\mathcal{V})}{2\sigma^2_{\text{constraint}}}\right)\pi(\mathcal{M})$$

Notice that the maximum of the posterior could well not occur at the maximum of the likelihood, because although the factorisation might fit the data well, the $\mathcal{U}$ factor may satisfy the camera constraints poorly.

3.1 Sampling the posterior

This formulation contains both a discrete and a continuous component. To sample continuous variables, we use a method due to [6] (and described in detail in [19]), which appends a set of independent Gaussian random variables $\mathbf{p}$ of no external significance to the state variables $\mathbf{q}$ to obtain a posterior of the form $exp(-\Phi(\mathbf{q})-(1/2)\mathbf{p}^T\mathbf{p})$. The negative log of this posterior has the form of the Hamiltonian for a particle in an energy field. We now use two types of proposal move: advance time for this particle; and choose new momenta (which can be done by Gibbs sampling, because each p_i is independent of every other, and of the state variables). This method moves to maxima of the posterior about as fast as gradient descent, and then samples around the maxima. If the state is far from a maximum, then the state moves down the energy field, gathering momentum, which is then thrown away by the second type of move, so the particle will tend to get trapped in maxima and explore them.

For **discrete variables**, we draw a sample from the full conditional for that discrete variable. For a posterior $p(v,\mathbf{s})$ with a discrete variable v which takes values 0 and 1 and a set of variables (both discrete and continuous) $\mathbf{s}$, the full conditional on v is $Prob\{d=0|\mathbf{s}=\mathbf{s}_0\}$, which is given by:

$$Prob\{d=0|\mathbf{s}=\mathbf{s}_0\}=\frac{p(0,\mathbf{s}_0)}{p(0,\mathbf{s}_0)+p(1,\mathbf{s}_0)}$$

If there are many discrete variables, we sample each in turn using this strategy; sampling proceeds in a random order to ensure the chain is reversible [7].

3.1.1 Starting the sampler

The sampler's state is given by $(\mathcal{U},\mathcal{V},\mathcal{M})$. In the examples, $m=40$ and $n=80$. This means the domain of the sampler is then 2^{6400} copies of $\Re^{640}$. This space is far too large to allow the sampler to blunder around in the hope of encountering a peak in the posterior. Furthermore, the relations between the discrete and the continuous variables are complex; for small errors, a sampler started at a random point burns in relatively quickly, but for large errors, the burn in can be very slow.

The values of $\mathcal{U}$ and $\mathcal{V}$ depend strongly on $\mathcal{M}$. If $\mathcal{M}$ has a 1 in a position corresponding to a significant tracker error, then that error can strongly affect the values of $\mathcal{U}$ and $\mathcal{V}$. This effect slows down the convergence of the sampler, because incorrect values of the continuous parameters mean that many data points lie a long way from the values predicted by the model, so that there is little distinction between points that correspond to the model and points that do not.

We start the sampler at a fair initial estimate of the mode. We obtain an initial value for the mask $\mathcal{M}^a$ by sampling an independent distribution on the bits that tends to deemphasize points which are distant from corresponding points in the previous and next frames. In particular, the i, j'th bit of $\mathcal{M}^a$ is 0 with probability

$$\frac{1-\exp\left(\frac{-\Delta_{ij}}{\sigma_w}\right)}{1+\exp\left(\frac{-\Delta_{ij}}{\sigma_w}\right)}$$

where $\Delta_{ij}=(d_{i,j}-d_{i+1,j})^2+(d_{i+m,j}-d_{i+m+1,j})^2+(d_{i,j}-d_{i-1,j})^2+(d_{i+m,j}-d_{i+m-1,j})^2$. Since this is a problem where the quantity of data swamps the number of parameters in the model, the choice of σ_w is fairly unimportant; the main issue is to choose the value to be small enough that large tracker errors are masked almost certainly.

The $\mathcal{U}^a$ and $\mathcal{V}^a$ that maximise

$$\sum_{ij}\left\{(d_{ij}-\sum_k u^s_{ik}v^s_{kj})^2 m^a_{ij}\right\}$$

are then obtained by a sweep algorithm which fixes $\mathcal{U}$ (resp. $\mathcal{V}$) and solves the linear system for $\mathcal{V}$

(resp. $\mathcal{U}$), and then swaps variables; the sweeps continue until convergence (which is guaranteed). We now compute an affine transformation $\mathcal{A}$ such that $\mathbf{C}^T(\mathcal{U}^a\mathcal{A}, \mathcal{A}^{-1}\mathcal{V}^a)\mathbf{C}(\mathcal{U}^a\mathcal{A}, \mathcal{A}^{-1}\mathcal{V}^a)$ is minimised; then $\mathcal{U}^s = \mathcal{U}^a\mathcal{A}$ and $\mathcal{V}^s = \mathcal{A}^{-1}\mathcal{V}^a$. We now draw a sample from the full conditional on each bit in the mask matrix, given $\mathcal{U}^s$ and $\mathcal{V}^s$ to obtain $\mathcal{M}^s$ The start state is then $(\mathcal{U}^s, \mathcal{V}^s, \mathcal{M}^s)$.

3.2 Results

While no convergence diagnostic is available, it is clear from the number of measurements marked good and the quality of fit that the sampler has burnt in (figure 4). A sampler started as described above burns in within 100 samples (which can be drawn in about 1 hour a Macintosh computer using Matlab). Results are obtained using the hotel dataset, courtesy of the Modeling by Videotaping group in the Robotics Institute, Carnegie Mellon University.

Figure 1 illustrates the kind of information a sampler can produce. Notice that the estimate of variation in inferred point position is obtained without geometric analysis; if a dataset is obtained by a camera translating in its plane, the sampler will return a set of samples with substantial variance perpendicular to that plane without further ado. Figure 2 illustrates tracker errors identified by a sampler; these errors are in the hotel data set as distributed.

3.2.1 Stability

Reconstructions cannot be compared on the basis of accuracy, because "ground truth" is not available. However, we can demonstrate the Bayesian method is significantly more stable under tracker errors and noise.

In the 40 frames used, six point measurements in nine frames are affected by small tracker errors as shown in figure 2. These (very small) errors affect the reconstruction obtained using the factorisation method because the factorisation of a matrix is a function of all entries.

Because the reconstruction is in some unknown scaled Euclidean frame, reconstructions are best compared by comparing angles subtended by corresponding triples of points, and by comparing distances between corresponding points scaled to minimize the errors.

To compare the stability of the methods, we now introduce larger tracker errors; a small percentage of data points, randomly selected, are replaced with draws from a uniform distribution on the image plane. If these points are included in the factorisation, the results are essentially meaningless. To provide a fair comparison, we use factorisations obtained using the method of section 3.1.1 (these are the start points of our sampler). These reconstructions are guaranteed to ignore large error points but will

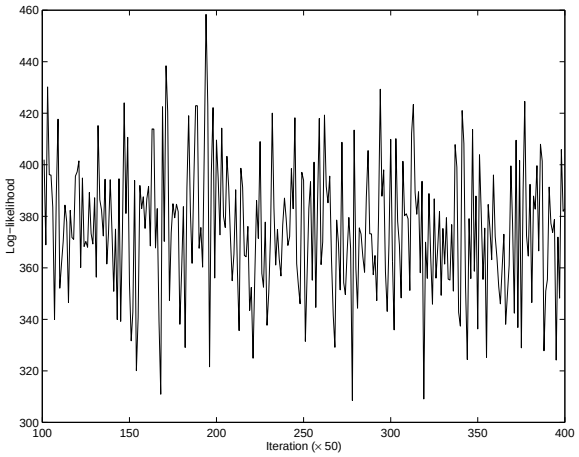

Figure 7: Log-posterior values for every fiftieth sample of the approximation to $P_{\mathbf{t}}(\mathbf{t})$ drawn by our sampler. The variation in values suggests that the sampler explores its domain effectively.

ignore a significant percentage of the data. In comparison, the sampler quickly accretes all points consistent with its model, and so gives significantly more stable measurements (cf [25], which uses maximum likelihood to identify correspondences).

4 Bayesian motion segmentation

We describe a segmentation method for the case of a maximum of two objects; extending this to the case of an unknown number of objects is straightforward. We require a new vector of discrete variables $\mathbf{t}$, one per point (i.e. one per column of $\mathcal{D}$). These variables take three values, allocating the column to noise, object one or object two respectively. Write n_{noise} for the number of columns allocated to noise and $n_{objects}$ for the total number of objects (which can be zero, one or two). The state of the sampler is now given by $(\mathbf{t}, \mathcal{U}_1, \mathcal{V}_1, \mathcal{M}_1, \mathcal{U}_2, \mathcal{V}_2, \mathcal{M}_2)$. The likelihood is obtained by evaluating equation 1 for each object separately.

The prior on $\mathcal{M}_1$ and $\mathcal{M}_2$ is as before. Priors on $\mathcal{U}_1$ and $\mathcal{U}_2$ are as before. We multiply by a further prior term that penalizes large numbers of noise columns and splitting a single object into two components:

$$\pi(n_{noise}, n_{objects}) \propto \exp\left(-w_{noise}n_{noise} - w_{object}n_{objects}\right)$$

4.1 Sampling for segmentation

The discrete component of the domain of this posterior is extremely large (of the order of $3^n 2^{2mn}$). An appropriate strategy partitions the variables. First, we sample an approximation to the marginal on $(\mathbf{t}, \mathcal{M}_1, \mathcal{M}_2)$,

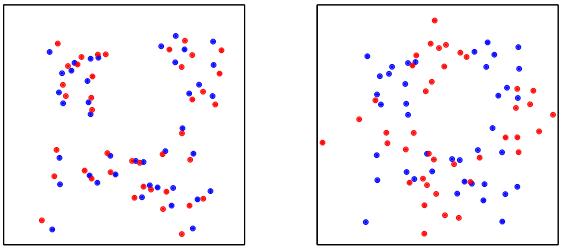

Figure 8: Two frames from our segmentation test sequence, which is obtained from 30 frames showing 40 of the tracked points of the hotel data set. We created a second (red) object by rotating the first one at a constant angular speed about its centroid.

$P_{\mathrm{d}iscrete}(\mathbf{t}, \mathcal{M}_1, \mathcal{M}_2)$. We now use this set of samples as a proposal mechanism for a sampler on the full posterior. Section 3 has established that, given a segmentation is correct, structure and motion can be recovered. We focus here on sampling $(\mathbf{t}, \mathcal{M}_1, \mathcal{M}_2)$.

A fair approximation to $P_{\mathrm{d}iscrete}(\mathbf{t}, \mathcal{M}_1, \mathcal{M}_2)$ can be obtained by solving for a maximum likelihood solution for structure and motion given $\mathbf{t}$, and then evaluating the posterior at that point. The maximum likelihood solution is obtained using the method of section 3.1.1. We now build an MCMC sampler, where each proposal involves mutation of some of the discrete variables (followed by the maximum likelihood solution to evaluate the approximation to the posterior). We require that each there be at least five columns of $\mathcal{D}$ associated with a given object. We use the following kinds of proposal move:

- Add a column of $\mathcal{D}$ to an object;
- Remove a column of $\mathcal{D}$ from an object;
- Swap a pair of points between two objects;
- Swap a random number of points between an object and noise;
- Mutate $\mathcal{M}_1$ or $\mathcal{M}_2$ by setting or clearing bits;
- Remove an object, making all of its points into noise;
- If there is only one object or none, create a new one, randomly choosing 5 points from noise;
- Merge the two objects into one;
- If only one object is present, split it randomly into two objects, each with 5 points or more.

We omit calculations of accept probabilities for lack of space.

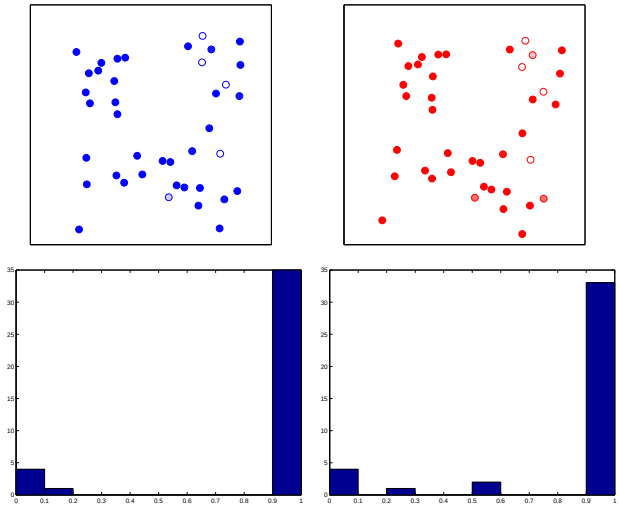

Figure 9: Our approximation to $P_{\mathrm{t}}(\mathbf{t})$ is relatively unambiguous. For each point, we compute the marginal probability that that point is associated with object one, noise, or object two. There are no samples where points from one object are confused with points from the other. **Top left** shows points from the first object, where the saturation is chosen according to the probability that they are noise (whiter means a point is more likely to be noise); **top right** shows points from the second object, shaded similarly. **Bottom left** shows a histogram of the probabilities that points from object one are allocated to that object (as opposed to noise; in no samples are points from different objects mixed) **bottom right** a histogram of the probabilities that points from object two are allocated to that object.

4.2 Results

Figure 8 illustrates our data set for this experiment. The chain mixes relatively well, as figure 7 indicates. The stationary distribution of the chain is relatively unambiguous; for every sample, all points allocated to object one originate with a single object and all allocated to object two originate with the other object. For most points, the marginal probability that that point is noise is low (figure 9).

5 Discussion

The output of a properly built sampler is an excellent guide to the inferences which can be drawn and to the ambiguities in a dataset. No independence assumptions are required to obtain this information. The sampler is robust to substantial noise; furthermore, we are not required to use specialised methods when the camera motion is degenerate — if, for example, the camera translates within a plane, the effect will appear in scatter plots that vary widely along the axis perpendicular to the plane.

The samplers described are superior to current methods

because they are robust and give a detailed representation of ambiguity, but are fairly slow. Current work is studying an incremental algorithm that appears to offer substantial speedups.

Acknowledgments

We thank Stuart Russell for pointing out the significance of MCMC as an inference mechanism, and Jean Ponce for suggesting a sampling method might apply to structure from motion.

References

[1] Y. Amit, U. Grenander, and M. Piccioni. Structural image restoration through deformable templates. *J. Am. Statist. Ass.*, 86:376–387, 1991.

[2] A. Blake and M. Isard. Condensation - conditional density propagation for visual tracking. *Int. J. Computer Vision*, 29(1):5–28, 1998.

[3] B.P. Carlin and T.A. Louis. *Bayes and empirical Bayes methods for data analysis*. Chapman and Hall, 1996.

[4] J. P. Costeira and T. Kanade. A multibody factorisation method for independently moving objects. *Int. J. Computer Vision*, 29(3):159–180, 1998.

[5] P.E. Debevec, C.J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach. In *SIGGRAPH '96*, pages 11–20, 1996.

[6] S. Duane, A.D. Kennedy, B.J. Pendleton, and D. Roweth. Hybrid monte carlo. *Physics Letters B*, 195:216–222, 1987.

[7] D. Gamerman. *Markov chain Monte Carlo*. Chapman-Hall, 1997.

[8] A. Gelman, J.B. Carlin, H.S. Stern, and D.B. Rubin. *Bayesian Data Analysis*. Chapman and Hall, 1995.

[9] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions and the bayesian restoration of images. *IEEE T. Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[10] S. Geman and C. Graffigne. Markov random field image models and their application to computer vision. In *Proc. Int. Congress of Math.*, 1986.

[11] W.R. Gilks, S. Richardson, and D.J. Spiegelhalter. Introducing markov chain monte carlo. In W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors, *Markov chain Monte Carlo in practice*. Chapman and Hall, 1996.

[12] W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors. *Markov chain Monte Carlo in practice*. Chapman and Hall, 1996.

[13] P.J. Green. Mcmc in image analysis. In W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors, *Markov chain Monte Carlo in practice*, pages 381–400. Chapman and Hall, 1996.

[14] U. Grenander. Tutorial in pattern theory. Technical report, Brown University, Providence, Rhode Island, 1983.

[15] K. Kanazawa, D. Koller, and S. Russell. Stochastic simulation algorithms for dynamic probabilistic networks. In *Proc Uncertainty in AI*, 1995.

[16] D. Koller and R. Fratkina. Using learning for approximation in stochastic processes. In *Proc. Machine Learning*, 1998.

[17] S.Z. Li. *Markov random field modeling in computer vision*. Springer-Verlag, 1995.

[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*, pages 696–702, 1998.

[19] R.M. Neal. Probabilistic inference using markov chain monte carlo methods. Computer science tech report crg-tr-93-1, University of Toronto, 1993.

[20] D.B. Phillips and A.F.M. Smith. Bayesian model comparison via jump diffusion. In W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors, *Markov chain Monte Carlo in practice*. Chapman and Hall, 1996.

[21] C. Poelman. The paraperspective and projective factorisation method for recovering shape and motion. Cmu cs-93-219, Carnegie-Mellon University, 1993.

[22] G.O. Roberts. Markov chain concepts related to sampling algorithms. In W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors, *Markov chain Monte Carlo in practice*. Chapman and Hall, 1996.

[23] L. Tierney. Introduction to general state-space markov chain theory. In W.R. Gilks, S. Richardson, and D.J. Spiegelhalter, editors, *Markov chain Monte Carlo in practice*. Chapman and Hall, 1996.

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

[25] P. Torr and A. Zisserman. Robust computation and parametrization of multiple view relations. In *Int. Conf. on Computer Vision*, pages 485–491, 1998.

[26] B. Triggs. Factorization methods for projective structure and motion. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 845–851, 1995.

[27] S.C. Zhu. Stochastic computation of medial axis in markov random fields. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 72–80, 1998.

[28] S.C. Zhu, Y. Wu, and D. Mumford. Filters, random fields and maximum entropy (frame): towards a unified theory for texture modelling. *Int. J. Computer Vision*, 27:107–126, 1998.