

Bayesian Modelling of Camera Calibration and Reconstruction

Rashmi Sundareswara and Paul R. Schrater
Computer Science
University of Minnesota
Twin Cities, MN 55455
sundares@cs.umn.edu, schrater@umn.edu

Abstract

Camera calibration methods, whether implicit or explicit, are a critical part of most 3D vision systems. These methods involve estimation of a model for the camera that produced the visual input, and subsequently to infer the 3D structure that gave rise to the input. However, in these systems the error in calibration is typically unknown, or if known, the effect of calibration error on subsequent processing (e.g. 3d reconstruction) is not accounted for. In this paper, we propose a Bayesian camera calibration method that explicitly computes calibration error, and we show how knowledge of this error can be used to improve the accuracy of subsequent processing. What distinguishes the work is the explicit computation of a posterior distribution on unknown camera parameters, rather than just a best estimate. Marginalizing (averaging) subsequent estimates by this posterior is shown to reduce reconstruction error over calibration approaches that rely on a single best estimate. The method is made practical using sampling techniques, that require only the evaluation of the calibration error function and the specification of priors. Samples with their corresponding probability weights can be used to produce better estimates of the camera parameters. Moreover, these samples can be directly used to improve estimates that rely on calibration information, like 3D reconstruction. We evaluate our method using simulated data for a structure from motion problem, in which the same point matches are used to calibrate the camera, estimate the motion, and reconstruct the 3D geometry. Our results show improved reconstruction over non-linear Camera calibration methods like the Maximum Likelihood estimate. Additionally, this approach scales much better in the face of increasingly noisy point matches.

1 Introduction

Camera Calibration and 3D reconstruction are used in a variety of fields ranging from robotics to entertainment to visualization. Most of the work in the calibration involves

estimating the camera model from visual input. However, camera calibration is highly susceptible to noise, because of which there are numerous linear [6, 9] and non-linear optimization methods [7, 9, 11, 21, 22, 23] that minimize least squares error likelihood defined on the point matches. In all of these linear and non-linear methods, the goal is a best estimate of the camera parameters. However, statistically speaking, the “best” answer is still a noisy one [8, 16]. Because the camera parameter estimates are used in subsequent steps to obtain the 3D reconstruction, shape analysis, etc, the error inherent in the camera parameter estimates contaminates all subsequent estimates.

Our aim in this work is to present an approach to 3D reconstruction via camera calibration such that the viewing parameters (i.e. the camera parameters) are discounted. We do so because one would like to find the most probable 3D scene that gave rise to the images capturing the scene. We accomplish this goal by exploring the posterior of the camera parameter space, i.e. $P(\theta|Data)$ using Bayesian Analysis with both uniform priors (in the case, where we do not know anything about the parameters) and gaussian and directional priors (where we have some knowledge of the parameters, say from a previous calibration). Rather than providing a best estimate of the camera parameters, the method produces a set of samples that can be used in several ways—to produce expected values of subsequent estimates (e.g. reconstruction), and to quantify the uncertainty in the camera parameter estimates, etc. Marginalizing over the camera parameters in this way to obtain the 3D structure allows one to “discount” the effect of the viewing parameters on the 3D scene to be estimated, especially when the posterior of the 3D reconstruction and camera parameters, $P(\theta, X|Data)$ is a non-linear manifold [18].

The paper is organized as follows. After a brief discussion of related work, we discuss Bayesian vs. non-Bayesian approaches to estimation and show that the Bayesian estimation reduces expected prediction error. Next we show how to specify priors and derive the posterior distribution for extrinsic camera parameters given a set of point

matches, and show how to make the procedure tractable using a slice sampler. Finally we test the method on a simulated structure from motion reconstruction problem and on a real data set.

2 Previous Work

Previous work in camera calibration has treated camera calibration as an explicit or implicit parameter estimation problem, and the approaches vary according to the assumptions made about the number of intrinsic vs. extrinsic parameters, the number of cameras, and the type of data (e.g. 2D point matches, the presence of 3D points from calibration objects, known angles/lengths in the scene, and whether autocalibration is used – the same data is used for calibration as reconstruction.) The heart of point match based calibration revolves around estimating the fundamental matrix, either linearly [6, 9, 17] or non-linearly [11, 22, 23]. Camera parameters can then be extracted from the fundamental matrix. Bundle adjustment methods give simultaneous estimates of camera and reconstruction parameters [19]. These methods minimize a cost function on feature prediction error, that is typically quadratic but robust variants also exist. Interpreted statistically, bundle adjustment performs maximum likelihood (or maximum a posteriori) inference in a joint camera parameter/3D point state space. The approaches most similar to ours formulate the calibration problem in explicit probabilistic terms, but these authors still only use the probability expressions to derive camera parameter estimates. In particular [20, 2, 3, 10] use priors on extrinsic parameters to generate MAP estimates of camera parameters. [18] also formulates the problem via a Bayesian model, however, he applies the framework to the problem of estimating the camera parameter model.

3 Theory

We break the theory of our work into several sections. First, we discuss Bayesian reconstruction in general, and then show that Bayesian estimates reduce reconstruction error. Then we show how to compute the required posterior distribution on camera parameters for the case of self-calibration.

3.1 Bayesian Estimation

In the Bayesian approach to estimation, parameters that are not directly estimated (*nuisance* parameters) are eliminated by marginalization. For example, in a maximum a posteriori estimate of x given data D with nuisance parameters θ :

$$\hat{x}(D) = \arg \max_x p(x|D) = \arg \max_x \int_{\theta} p(x|\theta, D)p(\theta|D)d\theta \quad (1)$$

The importance of this idea is that for most computer vision problems, the camera parameters are nuisance variables.

This contrasts with nested parameter estimation, where nuisance variances are estimated as well:

$$x^*(\hat{\theta}, D) = \arg \max_{x, \theta} p(x|\theta, D)p(\theta|D) \quad (2)$$

Because these maximizations can be performed separately, this is equivalent to finding the estimate $\hat{\theta}$ and plugging it in before maximizing x .

The averaging over the posterior performed by Bayesian estimation reduces the uncertainty of the parameter estimate [1, 8], and hence reduces the expected prediction error of the estimate. We derive a specific expression for the amount of this reduction below for the case of perspective reconstruction.

The benefits of the Bayesian approach depend on the relative uncertainties of x and θ . In particular, the advantages of Bayesian estimation over the “plug-in” approach disappear as the uncertainty in conditional distribution $p(x|\theta, D)$ approaches zero (or is tiny compared to $p(\theta|D)$).

$$\hat{x}(D) = \arg \max_x \int_{\theta} p(x|\theta, D)p(\theta|D)d\theta \quad (3)$$

$$= \arg \max_x \int_{\theta} \delta(x - x^*(\theta, D))p(\theta|D)d\theta \quad (4)$$

$$= \arg \max_{\theta} p(x^*(\theta, D), \theta|D) \quad (5)$$

$$= x^*(\hat{\theta}, D) \quad (6)$$

where $p(x^*(\theta, D), \theta|D)$ represents the slice through the joint distribution induced by the integral. An equivalent argument can be used to show that the benefits disappear as the uncertainty on θ goes to zero. To summarize, Bayesian estimation helps whenever the uncertainty on both parameter types is not negligible.

These results would appear to only apply to probabilistic approaches to camera calibration, which would exclude the most popular approaches [7, 9, 22, 23]. Although not optimal, in most cases it is still possible to improve estimates by averaging across the posterior of the camera parameters.

$$\bar{x}^*(D) = E_{p(\theta|D)} [x^*(\hat{\theta}, D)] \quad (7)$$

This procedure can be shown to reduce the expected prediction error of the estimate. However, it can bias the estimate if $p(\theta|D)$ is multimodal, and the modes produce incompatible values of $x^*(\hat{\theta}, D)$. In this case, averaging can still be used, but it should be performed separately around each mode (if possible) using local approximations or mixture distributions.

3.2 Averaging improves Reconstruction

The problem of reconstructing an unknown surface from a set of point matches under perspective projection can be for-

mulated as error minimization. For example, using a least squares error criterion, the reconstructed points minimize the error function:

$$\epsilon_{recon} = \sum_{j=1}^{N_{cam}} \sum_{i=1}^{N_{pts}} |z_{ij}u_{ij} - M_j p_i|^2 \quad (8)$$

where i indexes over data points and j indexes over cameras, u_{ij} are the projected points, z_{ij} are the unknown depths, M_j is the homogeneous projection matrix for camera j , and p_i is the surface point in world coordinates. More general loss functions can be used without altering the essence of the Bayesian approach. Let \mathbf{M} denote the full set of camera parameters required to specify the matrices M_j , and let D represent the set of point matches.

Let $\hat{p}_i(\mathbf{M}, D)$ represent the reconstruction generated by one of the many *plug-in* methods available to minimize equation 8. We are interested in relating the expected prediction error for *plug-in* and *averaged* estimates. Let $\bar{p}_i(D) = E_{\mathbf{M}} [\hat{p}_i(\mathbf{M}, D)]$ denote the plug-in estimate averaged across the posterior distribution on the camera parameters. The expected prediction error for *averaged* estimates is given by:

$$\epsilon_{ave} = E_D [|\bar{p}_i(D) - p_i|^2] \quad (9)$$

where p_i are the true surface point coordinates.

Now the expected prediction error for *plug-in* estimates is given by:

$$\epsilon_{plug} = E_D [E_{\mathbf{M}} [|\hat{p}_i(\mathbf{M}, D) - p_i|^2]] \quad (10)$$

A simple argument relates the two:

$$\begin{aligned} \epsilon_{plug} &= E_D [E_{\mathbf{M}} [|\hat{p}_i(\mathbf{M}, D) - p_i|^2]] \\ &= E_D [E_{\mathbf{M}} [|(\hat{p}_i(\mathbf{M}, D) - \bar{p}_i(D)) + (\bar{p}_i(D) - p_i)|^2]] \\ &= E_D [E_{\mathbf{M}} [|(\hat{p}_i(\mathbf{M}, D) - \bar{p}_i(D))|^2]] \\ &\quad - 2E_D [E_{\mathbf{M}} [(\bar{p}_i(D) - p_i)^T (\hat{p}_i(\mathbf{M}, D) - \bar{p}_i(D))]] \\ &\quad + E_D [E_{\mathbf{M}} [|(\bar{p}_i(D) - p_i)|^2]] \\ &= E_D [E_{\mathbf{M}} [|(\hat{p}_i(\mathbf{M}, D) - \bar{p}_i(D))|^2]] + \epsilon_{ave} \end{aligned} \quad (11)$$

This shows that the plug-in estimate's error is the sum of the average error and an additional term that can be interpreted as the variation in the estimate produced by the uncertainty in the camera parameters. This also means that the plug-in estimate's variance should always be higher than the averaged variance.

3.3 Camera Model and Calibration method

In this paper we restrict our attention to the case in which the same image data is used to estimate the camera parameters as perform the reconstruction. We assume that the scene

is viewed by multiple pinhole cameras with known internal parameters but unknown relative positions and orientations.

A pinhole camera model relates homogenous 3D world points $\mathbf{p}_i = [x_i, y_i, z_i, 1]$ to their image projections $\mathbf{u}_i = [u_i, v_i, 1]$:

$$\mathbf{u}_i = \frac{\mathbf{M}\mathbf{p}_i}{\mathbf{m}_3^T \mathbf{p}_i}$$

and

$$\mathbf{M} = \mathbf{K}[\mathbf{R} \ \mathbf{t}]$$

where \mathbf{m}_3^T is the third row of \mathbf{M} . The extrinsic parameters $[\mathbf{R} \ \mathbf{t}]$ are the rotation and translation respectively that relate the world coordinate frame to a camera's coordinate frame and \mathbf{K} denotes the camera's internal parameter matrix

$$\mathbf{K} = \begin{pmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

that encodes the origin u_0, v_0 , the image axes scale factors α, β and the skewness of the u, v axes in γ . The world coordinate frame is taken to be one of the camera's coordinate frames, so that the extrinsic parameters can be considered as the transformation between camera frames.

Camera parameter estimation for this case can be considered a self-calibration problem. In self-calibration, the data that is used to determine the extrinsic and intrinsic parameters are point matches between images of the same scene taken by different cameras. Calibration up to an overall scale factor can be accomplished using as few as 8 point matches in two images [6, 11]. However, these point matches are almost always noisy, due to a number of reasons - resolution, ambiguity, lighting, etc, resulting in errors in the camera parameter estimates. Below we explicitly model this error by formulating a posterior distribution on the camera parameters given the set of point matches. This requires that the joint probability distribution of point matches, object points to be reconstructed, and camera parameters $P(\{\mathbf{u}_{ij}\}, \theta, \{\mathbf{p}_{ij}\})$ be made explicit, which we do in the next section.

3.4 Bayesian Camera Parameter Analysis

Bayesian Analysis requires the specification of several probability distributions: likelihood, priors, posterior and the normalization factor. To put it simply, we have $Posterior = \frac{Likelihood * Prior}{NormalizationFactor}$. In our case, this translates to

$$P(X, \theta | D) = \frac{P(D | \theta, X) P(\theta) P(X)}{P(D)} \quad (12)$$

where $D = \{\mathbf{u}_{ij}\}$, θ contains the rotation and translation parameters and $X = \{\mathbf{p}_{ij}\}$. We ignore the normalization factor, $P(D)$, as it is a fixed constant for this problem. In

addition, we ignore the prior on X for simplicity. Below, we provide expressions for each term in our Bayesian Analysis of Camera Calibration.

1. Priors

This is where we incorporate any previous knowledge we have about the camera's rotation \mathbf{R} , and translation, \mathbf{t} . To be specific, we characterize the set of extrinsic parameters with five variables. We denote this vector as θ . The first three components of θ form a vector θ_R that parametrizes the rotation. The direction of the vector gives the rotation axis and the angle is given by the vector's magnitude. The rotation matrix \mathbf{R} can be obtained from this representation using Rodriguez' Formula [4]. The next two components of θ , θ_t parameterize the direction of \mathbf{t} . Given the scale ambiguity, we do not need to specify the magnitude of \mathbf{t} . No information about \mathbf{R} and \mathbf{t} can be modelled as a uniform prior on θ , in which case the posterior is equal to the likelihood. Otherwise, we model the prior on θ as a 3D Gaussian distribution for rotation parameters (which is only evaluated within a shell of radius π), and as a vonMises-Fisher distribution (the analog of the Gaussian distribution for a sphere) for the direction of the translation [12]. The probability of θ_R is given by the formula for a multivariate 3D Gaussian:

$$P(\theta_R) = \frac{1}{(2\pi)^{\frac{3}{2}}|C_R|} \exp\left[-\frac{1}{2}(\theta_R - \mu_R)^T C_R^{-1}(\theta_R - \mu_R)\right] \quad (13)$$

where μ_R and C_R are the mean and covariances respectively.

The translation is defined by a translation direction given by two spherical coordinates (since the overall solution is defined up to a scale factor, the use of spherical coordinates on a unit sphere is justified). Let $\theta_T = [\gamma, \phi]$. The probability of θ_T is given by the vonMises-Fisher expression:

$$P(\theta_T) = \frac{\kappa}{2 \sinh \kappa} e^{\kappa(\cos \gamma \cos \alpha + \sin \gamma \sin \alpha \cos(\phi - \beta))} \sin \gamma \quad (14)$$

where α and β are the mean spherical angles and κ is the concentration parameter.

2. *Likelihood:* We assume that the error in the point matches is Gaussian in nature, and that point match errors are independent of each other. This assumption is quite common in the camera calibration literature [23, 18]. Because of these assumptions, we can formulate the Likelihood as the following:

$$P(D|\theta, X) =$$

$$\left(\frac{1}{2\pi\sqrt{|C|}}\right)^n \exp\left[-\frac{1}{2}\sum_{i=1}^n \sum_{j=1}^J e_{ij}^T C^{-1} e_{ij}\right] \quad (15)$$

where n is the number of points, j indexes the cameras, C is the overall covariance matrix of the match error, and $e_{ij} = \mathbf{u}_{ij} - \frac{\mathbf{M}_j \mathbf{p}_i}{\mathbf{m}_{j3}^T \mathbf{p}_i}$ is the error in the point matches given that \mathbf{M}_j is the projection matrix for camera j and \mathbf{m}_{j3}^T is the third row of \mathbf{M}_j .

3. Posterior

The posterior distribution is now defined as

$$P(X, \theta|D) = P(D|\theta, X)P(\theta) \quad (16)$$

As explained before, we don't include the normalization factor or $P(X)$. However, $P(X)$ will be used in our extension to this work, where we have priors on the 3D structure to be estimated. Our current work uses this Posterior distribution to average across θ to reduce average reconstruction error on X .

3.5 Computing solutions

An initial parameter solution is obtained by the Fundamental 8-point algorithm [6]. This initial solution is used by the Slice Sampler to explore the posterior of the extrinsic parameter space by sampling. To evaluate each sample, we have to compute the likelihood and the prior. The likelihood is computed as given by Eq. 15, where the error is defined by the squared difference between the measured points and the re-projected points. The re-projected points are computed by the projection matrices defined by the sample and reconstructed 3D points which can be computed using eigenanalysis, given point matches and the projection matrices. The priors are computed as in Eq.13 and Eq. 14. At the end of the process, we have samples of the extrinsic parameters and world points X , along with their corresponding probabilities. We use the probabilities to compute an expected value for the model and a covariance matrix for the extrinsic parameters in the following way:

$$E[\theta] = \sum_{i=1}^{N_{samples}} P(\theta_i|D)\theta_i \quad (17)$$

$$\mathbf{C}_R = \sum_{i=1}^{N_{samples}} P(\theta_R^i|D)(\theta_R^i - E[\theta_R])(\theta_R^i - E[\theta_R])^t \quad (18)$$

$$\mathbf{C}_T = \sum_{i=1}^{N_{samples}} P(\theta_T^i|D)(\theta_T^i - E[\theta_T])(\theta_T^i - E[\theta_T])^t \quad (19)$$

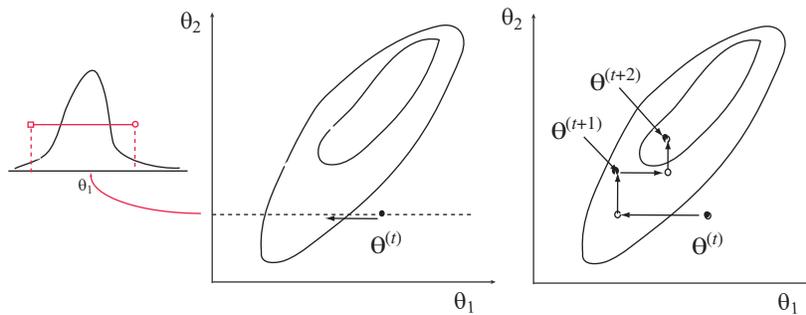


Figure 1: Slice Sampler: The sampler cycles through one-dimensional slices in the parameter space, choosing uniformly in a particular interval of the slice

where θ_R^i and θ_T^i represent a sample Rotation and translation respectively.

$$E[\mathbf{p}] = \sum_i f(\theta_i, D)P(\theta_i|D) \quad (20)$$

4 Implementation

4.1 Sampling

Since the posterior is difficult to compute analytically, we can sample from it directly instead. Specifically, we sample various θ and assign the value of $P(\theta|D)$ as the goodness of that θ . We use a Slice Sampler [15, 13]. Slice Sampling is a Markov Chain Monte Carlo method. It can be evaluated as long as (a potentially unnormalized) $P(\mathbf{x})$ can be obtained for any point \mathbf{x} . The sampler cycles through one-dimensional slices in the parameter space (See Fig. 1), choosing uniformly in a particular interval of the slice. At the end of sampling process, we obtain samples of the parameters (θ) with their corresponding probabilities. The description of one-dimensional sampling is given as follows (adapted from [15]):

1. Pick a value, y , uniformly between 0 and $p(x_0)$, where $p(x_0)$ denotes the probability of the current point x_0 . This is used to define a horizontal slice in the posterior for points x having the same probability $p(x_0)$.
2. Place an interval of random width, w , around x_0 . This width is expanded by a stepsize, s , until both ends are outside the slice. Points picked outside the horizontal slice are used to shrink the interval.
3. A new point, x_1 is picked when a point is found inside the slice.

We use these samples to reconstruct the 3D points, and then take the Expected value of all the 3D points to obtain the average reconstructed set of points, as shown in Fig. 2.

where $\mathbf{p}_i = f(\theta_i, D)$ is the function that reconstructs the 3D world point from the parameters θ and point matches D . For more information on f , see [5].

While sampling methods are not as efficient as linear methods, the complexity is $O(NML)$, where N is the number of function evaluations per sample, M is the number of dimensions, and L is the required number of samples. Thus if posterior evaluation is cheap and N is small, then the method can be quite efficient. The largest problem for sampling methods is the potential for N to become large due to random walk behavior. However, there are many methods to overcome this behavior, including overrelaxation [15] and exact sampling [13]. In our implementation, an evaluation of a sample entails the computation of projection matrices, reprojection of computed 3D points, and computing prior values.

5 Results

We compared the reconstructed points returned by the Slice Sampler, i.e. the expected value of a set of 3D points, $E[\mathbf{p}]$ (see Eqn. 20) with the 3D points reconstructed by the solution θ returned by the Maximum Likelihood (ML) estimate using a Levenberg-Markquardt maximization procedure [14, 23] implemented in MATLAB. Our data was a pair of simulated 512x512 images of a 3D cube, and also a pair of real images (1704 x 2272), where the ground truth point matches were selected by hand. The camera, a Minolta S414, was calibrated using a calibration object. We discuss both below.

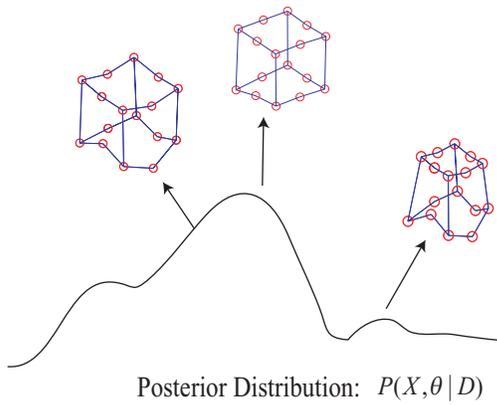


Figure 2: Cubes reconstructed with different θ , each with probability $P(X, \theta|D)$. The distribution shown is an example of a 1D slice of the Posterior $P(X, \theta|D)$. The Bayesian approach reconstructs different 3D structures by sampling the posterior, and finally computed an Expected Value over all reconstructions.

5.1 Experiments with Simulated Data

For the simulated pair of images of a cube, we generated 100 data sets at three different levels of noise: low noise (pixel standard deviation/resolution = 0.0062), medium noise (pixel standard deviation/resolution = 0.138) and high noise (pixel standard deviation/resolution = 0.0195). We reconstructed points on the cube for these data sets using both ML and our method with three levels of priors: high prior (angular variance = 2.8 degrees), low prior (angular variance = 25.7 degrees), and no-prior or the uniform prior where $P(\theta)$ is always 1.

The *bias* in the reconstructed points was calculated as the mean of the squared distance from the truth.

$$\frac{1}{M_{reps} * N_{vertices} * 3} \sum_{M_{reps}} \sum_{N_{vertices}} \sum_{xyz} (X_{bayes} - X_{truth})^2 \quad (21)$$

The *standard deviation* of the reconstructed estimates was computed as the mean of the individual 3D covariances of the computed points over all the repetitions. If X_i represents a 3D point reconstructed in a repetition and \hat{X}_i is the average over all repetitions for that vertex, then the average Standard deviation per vertex was computed as:

$$\frac{1}{N_{verts}} \frac{1}{M_{reps} - 1} \sum_{i=1}^{M_{reps}} (X_i - \hat{X})(X_i - \hat{X})^T \quad (22)$$

Fig.3 and Fig. 4 show the average 3D bias and average per vertex standard deviation, respectively for our method at the three prior levels and the ML estimate.

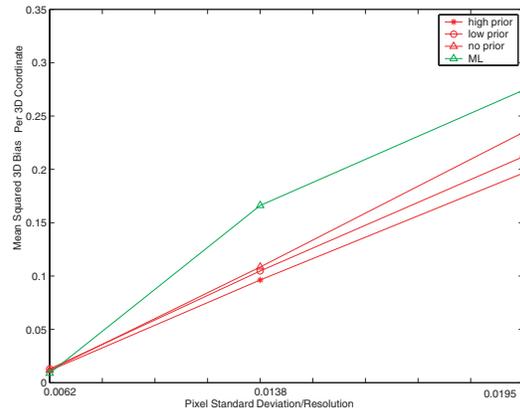


Figure 3: Real data reconstruction bias as computed in eqn. 21 vs. pixel variance noise. “ML” estimate refers to Maximum Likelihood estimate. “High prior” refers to angular variance of 2.8 degrees, “low prior” refers to angular variance of 25.7 degrees), and “no prior” is a uniform distribution. The figure shows all three versions of the Bayesian approach to be accounting for noise better than the Maximum Likelihood approach.

Not surprisingly, the figures show that prior information is helpful. But they also show that averaging provides improved prediction regardless of prior, with prediction improvement increasing with the amount of feature matching error, as predicted by equation 11. This is because the Maximum Likelihood estimate trusts the noisy data too much, unlike the Bayesian estimates which average the 3D point samples. Fig. 4 also shows the linear increase in Bias as the pixel noise increases. Figure 4 shows the average standard deviation of a reconstructed point using the ML estimate varies more those reconstructed using the Bayesian estimates. This is due to the fact again that the ML estimate follows the noisy data closely whereas the Bayesian sample averaging tend to lean centrally in the posterior distribution as illustrated in figure 2.

5.2 Experiments with Real Data

Using a calibration object and 18 point matches on a 1704x2272 resolution image, we computed the intrinsic parameters and the 3D ground truth of the calibration object shown in Fig. 5b.

We generated 40 data sets at three different levels of point match error: low (pixel standard deviation/resolution = 0.0016 pixels), Medium (pixel standard deviation/resolution = 0.0036 pixels) and High (pixel standard deviation/resolution = 0.0112). Although the deviations may seem large, it is somewhat justified considering the resolution of the image. Moreover, in applications such as tracking, large deviations are quite common. Since we know from the previous experiments on the simulated data

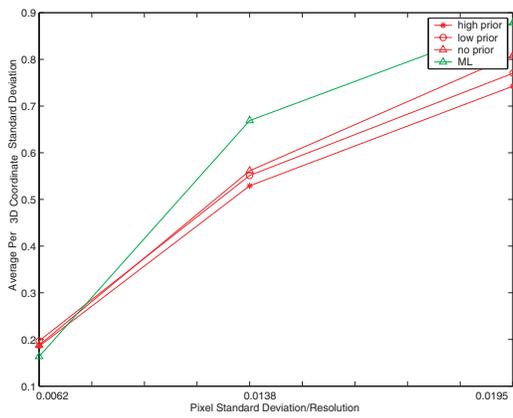


Figure 4: Average per vertex Standard Deviation as computed in Eqn. 22 vs. Pixel Noise variance. “ML” estimate refers to the Maximum Likelihood estimate. “High prior” refers to angular variance of 2.8 degrees, “Low prior” refers to angular variance = 25.7 degrees), and “No prior” is a uniform distribution. Plug-in estimates should theoretically have a higher variance than the Bayesian approach.

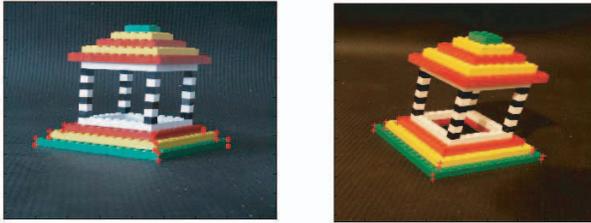


Figure 5: Real data images 1704x2272 used in the experiments

that the prior information is helpful, we decided to only compare the uniform Bayes model with the Maximum Likelihood approach as a fair test on real data. The results are shown in fig.6.

6 Discussion

Bayesian methods offer distinct advantages over traditional estimation solutions. For calibration, the most important advantages are less sensitivity to data errors and the ability to learn prior distributions on parameters from past reconstructions. The results show the Bayesian approach is better able to cope with increasing point-match error than the Maximum Likelihood approach. In the case where an uncalibrated camera is used for multiple reconstructions, the posterior distributions from earlier reconstructions can be used as the prior for the current reconstruction. This is es-

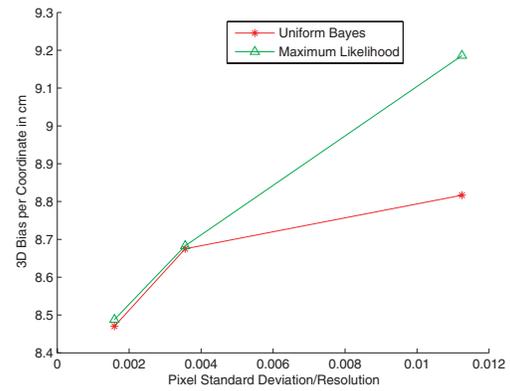


Figure 6: Reconstruction Bias as computed in Eqn. 21 vs. Pixel Standard Deviation/Resolution noise. “ML” estimate refers to Maximum Likelihood estimate. “Uniform Bayes” refers to the Bayesian reconstruction performed with no prior information on the extrinsic camera parameters. The Bayesian method scales better with noise than the Maximum Likelihood approach. This figure shows the advantage of the Bayesian model in high noise situations it has performed by marginalizing over all the reconstructions it has performed by sampling the camera parameters. ML picks the best answer it has encountered so far but doesn’t account for the fact that there is error in the camera estimate

pecially important for intrinsic parameters that are unlikely to change between reconstructions. In addition, the current implementation will be extended to account for $P(X_i)$, so that we can incorporate a prior for the shape of an object.

7 Conclusion

Inaccurate estimates of intrinsic and extrinsic parameters during camera calibration introduces error in subsequent processing. We showed methods that plug-in camera parameter estimates will underperform Bayesian averaging in terms of estimate variance. We showed this phenomenon directly for the case of a Maximum Likelihood estimation in a structure-from-motion simultaneous camera motion/3D reconstruction problem. Maximum Likelihood methods rely on the data too much to minimize error, thereby not being very effective in the case when there is lots of noise. We approach the problem using Bayesian principles. Having priors on the extrinsic parameters θ , and a likelihood for $P(D|\theta)$, we are able to sample the Posterior space, $P(\theta|D)$ and evaluate a sample’s (θ) probability. Using these probabilities, we compute an Expected Value of reconstructed 3D world points, $E[X_i]$. Our results show that this method does outperform the Maximum Likelihood method, even in the case, where we do not have a prior, verifying that the Expected Value is a better predictor.

References

- [1] Berger, J., Liseo, B. and Wolpert, R., "Integrated Likelihood Methods for Eliminating Nuisance Parameters" *Statistical Science*, 14, 1-28, 1999.
- [2] J. Lasenby, W. J. Fitzgerald, C. J. L. Doran and A. N. Lasenby. "New Geometric Methods for Computer Vision," *Int. J. Comp. Vision* 36(3), p. 191-213, 1998.
- [3] Chris Doran, "Bayesian Inference and Geometric Algebra: an application to Camera Localization," Editors E.B. Corrochano and G. Sobczyk *Geometric Algebra with Applications in Science and Engineering*, Birkhauser, Boston, 2001
- [4] Faugeras, O.D., *Three Dimensional Vision, a Geometric Viewpoint*. MIT Press, 1993
- [5] Forsyth and Ponce, *Computer Vision, a Modern Approach*. Prentice Hall Press, 2003.
- [6] R. Hartley, "In defense of the 8-point algorithm," *In Proceedings of the 5th International Conference on Computer Vision*, pages 1064-1070, Boston, MA, June 1995. IEEE Computer Society Press.
- [7] R. I. Hartley, "An algorithm for self calibration from several views", *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 908-912, Seattle, WA, June 1994. IEEE.
- [8] J.A. Hoeting, D. Madigan, A. E. Raftery and C. T. Volinsky, "Bayesian Model Averaging: A Tutorial," *Statistical Science*, Vol. 14, No. 4, 382-417
- [9] T.S. Huang and A.N. Netravali, "Motion and structure from feature correspondences: A review," *Proceedings of the IEEE*, 82(2):252-268, 1994.
- [10] J. Lasenby and A. Stevenson, "Using geometric algebra in optical motion capture". In E. Bayro and G. Sobczyk, editors, *Geometric algebra: A geometric approach to computer vision, neural and quantum computing, robotics and engineering*. Birkhauser, 2000.
- [11] Q.-T. Luong and O. Faugeras, "Self-calibration of a moving camera from point correspondences and fundamental matrices," *The International Journal of Computer Vision*, 22(3):261-289, 1997.
- [12] Mardia, K. V., "Directional statistics," New York : Wiley, 2000.
- [13] MacKay, David, "Information Theory, Inference, and Learning Algorithms," Cambridge University Press, September 2003.
- [14] J. More, "The levenberg-marquardt algorithm, implementation and theory", *Numerical Analysis, Lecture Notes in Mathematics* 630. Springer-Verlag, 1977.
- [15] Neal, R. M. (2000) "Slice sampling", Technical Report No. 2005, Dept. of Statistics, University of Toronto.
- [16] D.S. Sivia, "Data Analysis, A Bayesian tutorial," Oxford University Press, 1996.
- [17] Torr, P.H.S and Zisserman, A, "MLESAC: A new robust estimator with application to estimating imaging geometry," *CVIU*, 78(1):138-156
- [18] Torr, P.H.S, "Bayesian Model Estimation and Selection for Epipolar Geometry and Generic Manifold Fitting," *International Journal of Computer Vision*, 50(1),36-61, 2002.
- [19] Triggs B., McLauchlan P., Hartley R. and Fitzgibbon A., "Bundle Adjustment - A Modern Synthesis," *Vision Algorithms: Theory and Practice, Springer-Verlag LNCS 1883, 2000*
- [20] RJ. Valkenburg, "A Bayesian approach to camera system calibration/spatial intersection," *Proceedings of IVCNZ 98*, Auckland, New Zealand, December 1998, pp 11-16.
- [21] J. Weng, T.S. Huang, and N. Ahuja, "Motion and structure from two perspective views: Algorithms, error analysis and error estimation". *IEEE Trans. Pattern Anal. Mach. Intelligence*, 11(5):451-476, 1989.
- [22] Z. Zhang, "Determining the epipolar geometry and its uncertainty: A review," *Int. J. of Comp. Vision*, 27(2):161-195, 1998.
- [23] Zhengyou Zhang, "A flexible new technique for Camera Calibration," *Technical Report, MSR-TR-98-71*, Microsoft Research, Microsoft Corporation.