This ECCV 2018 paper, provided here by the Computer Vision Foundation, is the author-created version.
The content of this paper is identical to the content of the officially published ECCV 2018
LNCS version of the paper as available on SpringerLink: https://link.springer.com/conference/eccv

Fast and Accurate Camera Covariance Computation for Large 3D Reconstruction

Michal Polic[1], Wolfgang Förstner[2], and Tomas Pajdla[1]
[0000−0003−3993−337X] [0000−0003−1049−8140] [0000−0001−6325−0072]

[1] CIIRC, Czech Technical University in Prague, Czech Republic
{michal.polic,pajdla}@cvut.cz, www.ciirc.cvut.cz
[2] University of Bonn, Germany
wfoerstn@uni-bonn.de, www.ipb.uni-bonn.de

Abstract. Estimating uncertainty of camera parameters computed in Structure from Motion (SfM) is an important tool for evaluating the quality of the reconstruction and guiding the reconstruction process. Yet, the quality of the estimated parameters of large reconstructions has been rarely evaluated due to the computational challenges. We present a new algorithm which employs the sparsity of the uncertainty propagation and speeds the computation up about ten times w.r.t. previous approaches. Our computation is accurate and does not use any approximations. We can compute uncertainties of thousands of cameras in tens of seconds on a standard PC. We also demonstrate that our approach can be effectively used for reconstructions of any size by applying it to smaller sub-reconstructions.

Keywords: uncertainty, covariance propagation, Structure from Motion, 3D reconstruction

1 Introduction

Three-dimensional reconstruction has a wide range of applications (e.g. virtual reality, robot navigation or self-driving cars), and therefore is an output of many algorithms such as Structure from Motion (SfM), Simultaneous location and mapping (SLAM) or Multi-view Stereo (MVS). Recent work in SfM and SLAM has demonstrated that the geometry of three-dimensional scene can be obtained from a large number of images [1],[14],[16]. Efficient non-linear refinement [2] of camera and point parameters has been developed to produce optimal reconstructions.

The uncertainty of detected points in images can be efficiently propagated in case of SLAM [16],[28] into the uncertainty o three-dimensional scene parameters thanks to fixing the first camera pose and scale. In SfM framework, however, we are often allowing for gauge freedom [18], and therefore practical computation of the uncertainty [9] is mostly missing in the state of the art pipelines [23],[30],[32].

In SfM, reconstructions are in general obtained up to an unknown similarity transformation, i.e., a rotation, translation, and scale. The backward uncertainty

propagation [13] (the propagation from detected feature points to parameters the of the reconstruction) requires the "inversion" of a Fischer information matrix, which is rank deficient [9],[13] in this case. Naturally, we want to compute the uncertainty of the inner geometry [9] and ignore the infinite uncertainty of the free choice of the similarity transformation. This can be done by the Moore-Penrose (M-P) inversion of the Fisher information matrix [9],[13],[18]. However, the M-P inversion is a computationally challenging process. It has cubic time and quadratic memory complexity in the number of columns of the information matrix, i.e., the number of parameters.

Fast and numerically stable uncertainty propagation has numerous applications [26]. We could use it for selecting the next best view [10] from a large collection of images [1],[14], for detecting wrongly added cameras to existing partial reconstructions, for improving fitting to the control points [21], and for filtering the mostly unconstrained cameras in the reconstruction to speed up the bundle adjustment [2] by reducing the size of the reconstruction. It would also help to improve the accuracy of the iterative closest point (ICP) algorithm [5], by using the precision of the camera poses, and to provide the uncertainty of the points in 3D [27].

2 Contribution

We present the first algorithm for uncertainty propagation from input feature points to camera parameters that works without any approximation of the natural form of the covariance matrix on thousands of cameras. It is about ten times faster than the state of the art algorithms [19],[26]. Our approach builds on top of Gauss-Markov estimation with constraints by Rao [29]. The novelty is in a new method for nullspace computation in SfM. We introdice a fast sparse method, which is independent on a chosen parametrization of rotations. Further, we combine the fixation of gauge freedom by nullspace, from Förstner and Wrobel [9] and methods applied in SLAM, i.e., the block matrix inversion [6] and Woodbury matrix identity [12].

Our main contribution is a clear formulation of the nullspace construction, which is based on the similarity transformation between parameters of the reconstruction. Using the nullspace and the normal equation from [9], we correctly apply the block matrix inversion, which has been done only approximately before [26]. This brings an improvement in accuracy as well as in speed. We also demonstrate that our approach can be effectively used for reconstructions of any size by applying it to smaller sub-reconstructions. We show empirically that our approach is valid and practical.

Our algorithm is faster, more accurate and more stable than any previous method [19],[26],[27]. The output of our work is publicly available as source code which can be used as an external library in nonlinear optimization pipelines, like Ceres Solver [2] and reconstruction pipelines like [23],[30],[32]. The code, datasets, and detailed experiments will be available online https://michalpolic.github.io/usfm.github.io.

3 Related work

The uncertainty propagation is a well known process [9],[13],[18],[26]. Our goal is to propagate the uncertainties of input measurements, i.e. feature points in images, into the parameters of the reconstruction, e.g. poses of cameras and positions of points in 3D, by using the projection function [13]. For the purpose of uncertainty propagation, a non-linear projection function is in practice often replaced by its first order approximation using its Jacobian matrix [8],[13]. For the propagation using higher order approximations of the projection function, as described in Förstner and Wrobel [9], higher order estimates of uncertainties of feature points are required. Unfortunately, these are difficult to estimate [9, 25] reliably.

In the case of SfM, the uncertainty propagation is called the *backward propagation of non-linear function in over-parameterized case* [13] because of the projection function, which does not fully constrain the reconstruction parameters [22], i.e., the reconstruction can be shifted, rotated and scaled without any change of image projections.

We are primarily interested in estimating *inner geometry* , e.g. angles and ratios of distance, and its *inner precision* [9]. Inner precision is invariant to changes of gauge, i.e. to similarity transformations of the cameras and the scene [18]. A natural choice of the fixation of gauge, which leads to the inner uncertainty of inner geometry, is to fix seven degrees of freedom caused by the invariance of the projection function to the similarity transformation of space [9],[13],[18]. One way to do this is to use the Moore-Penrose (M-P) inversion [24] of the Fisher information matrix [9].

Recently, several works on speeding up the M-P inversion of the information matrix for SfM frameworks have appeared. Lhuillier and Perriollat [19] used the block matrix inversion of the Fisher information matrix. They performed M-P inversion of the Schur complement matrix [34] of the block related to point parameters and then projected the results to the space orthogonal to the similarity transformation constraints. This approach allowed working with much larger scenes because the square Schur complement matrix has the dimension equal to the number of camera parameters, which is at least six times the number of cameras, compared to the mere dimension of the square Fisher information matrix, which is just about three times the number of points.

However, it is not clear if the decomposition of Fisher information matrix holds for M-P inversion without fulfilling the rank additivity condition [33] and it was shown in [26] that approach [19] is not always accurate enough. Polic et al. [26] evaluated the state of the art solutions against more accurate results computed in high precision arithmetics, i.e. using 100 digits instead of 15 significant digits of double precision. They compared the influence of several fixations of the gauge on the output uncertainties and found that fixing three points that are far from each other together with a clever approximation of the inversion leads to a good approximation of the uncertainties.

The most related work is [29], which contains uncertainty formulation for Gauss-Markov model with constraints. We combine this result with our new approach for nullspace computation to fixing gauge freedom.

Finally, let us mention work on fast uncertainty propagation in SLAM. The difference between SfM and SLAM is that in SLAM we know, and fix, the first camera pose and the scale of the scene which makes the information matrix full rank. Thus one can use a fast Cholesky decomposition to invert a Schur complement matrix as well as other techniques for fast covariance computation [16, 17]. Polok, Ila et al. [15],[28] claim addressing uncertainty computation in SfM but actually assume full rank Fisher information matrix and hence do not deal with gauge freedom. In contrary, we solved here the full SfM problem which requires dealings with gauge freedom.

4 Problem formulation

In this section, we describe basic notions in uncertainty propagation in SfM and provide the problem formulation.

The set of parameters of three-dimensional scene $\theta = \{P, X\}$ is composed from n cameras $P = \{P_1, P_2, ..., P_n\}$ and m points $X = \{X_1, X_2, ..., X_m\}$ in 3D. The i-th camera is a vector $P \in \mathbb{R}^8$, which consist of internal parameters (i.e. focal length $c_i \in \mathbb{R}$ and radial distortion $k_i \in \mathbb{R}$) and external parameters (i.e. rotation $r_i \in SO(3)$ and camera center $C_i \in \mathbb{R}^3$). Estimated parameters are labelled with the hat $\hat{}$.

We consider that the parameters $\hat{\theta}$ were estimated by a reconstruction pipeline using a vector of t observations $u \in \mathbb{R}^{2t}$. Each observation is a 2D point $u_{i,j} \in \mathbb{R}^2$ in the image i detected up to some uncertainty that is described by its covariance matrix $\Sigma_{u_{i,j}} = \Sigma_{\epsilon_{i,j}}$. It characterizes the Gaussian distribution assumed for the detection error $\epsilon_{i,j}$ and can be computed from the structure tensor [7] of the local neighbourhood of $u_{i,j}$ in the image i. The vector $\hat{u}_{i,j} = p(\hat{X}_j, \hat{P}_i)$ is a projection of point $\hat{X}_j$ into the image plane described by camera parameters $\hat{P}_i$. All pairs of indices (i, j) are in the index set S that determines which point is seen by which camera

$$\hat{u}_{i,j} = u_{i,j} - \epsilon_{i,j} \tag{1}$$

$$\hat{u}_{i,j} = p(\hat{X}_j, \hat{P}_i) \qquad \forall (i,j) \in S \tag{2}$$

Next, we define function $f(\hat{\theta})$ and vector ϵ as a composition of all projection functions $p(\hat{X}_j, \hat{P}_i)$ and related detection errors $\epsilon_{i,j}$

$$u = \hat{u} + \epsilon = f(\hat{\theta}) + \epsilon \tag{3}$$

This function is used in the non-linear least squares optimization (Bundle Adjustment [2])

$$\hat{\theta} = \arg\min_{\theta} \left\| f(\hat{\theta}) - u \right\|^2 \tag{4}$$

which minimises the sum of squared differences between the measured feature points and the projections of the reconstructed 3D points. We assume the Σ_u as a block diagonal matrix composed of $\Sigma_{u_{i,j}}$ blocks. The optimal estimate $\hat{\theta}$, minimising the Mahalanobis norm, is

$$\hat{\theta} = \arg\min_{\theta} r^\top(\hat{\theta})\Sigma_u^{-1} r(\hat{\theta}) \tag{5}$$

To find the formula for uncertainty propagation, the non-linear projection functions f can be linearized by the first order term of its Taylor expansion

$$f(\theta) \approx f(\hat{\theta}) + J_{\hat{\theta}}(\hat{\theta} - \theta) \tag{6}$$

$$f(\theta) \approx \hat{u} + J_{\hat{\theta}}\Delta\theta \tag{7}$$

which leads to the estimated correction of the parameters

$$\hat{\theta} = \theta + \arg\min_{\Delta\theta}(J_{\hat{\theta}}\Delta\theta + \hat{u} - u)^\top \Sigma_u^{-1}(J_{\hat{\theta}}\Delta\theta + \hat{u} - u) \tag{8}$$

Partial derivatives of the objective function must vanishing in the optimum

$$\frac{1}{2}\frac{\partial(r^\top(\theta)\Sigma_u^{-1}r(\theta))}{\partial\theta^\top} = J_{\hat{\theta}}^\top \Sigma_u^{-1}(J_{\hat{\theta}}\widehat{\Delta\theta} + \hat{u} - u) = J_{\hat{\theta}}^\top \Sigma_u^{-1} r(\hat{\theta}) = 0 \tag{9}$$

which defines the *normal equation system*

$$M\widehat{\Delta\theta} \;=\; \boldsymbol{m} \tag{10}$$

$$M = J_{\hat{\theta}}^\top \Sigma_u^{-1} J_{\hat{\theta}}\ , \quad \boldsymbol{m} = J_{\hat{\theta}}^\top \Sigma_u^{-1}(u - \hat{u}) \tag{11}$$

The normal equation system has seven degrees of freedom and therefore requires to fix seven parameters, called the gauge [18], namely a scale, a translation and a rotation. Any choice of fixing these parameters leads to a valid solution.

The natural choice of covariance, which is unique, has the zero uncertainty in the scale, the translation, and rotation of all cameras and scene points. It can be obtained by the M-P inversion of Fisher information matrix M or by Gauss-Markov Model with constraints [9]. If we assume a constraints $h(\hat{\theta}) = 0$, which fix the scene scale, translation and rotation, we can write their derivatives, i.e. the nullspace H, as

$$H^T\Delta\theta = 0 \qquad H = \frac{\partial h(\hat{\theta})}{\partial\hat{\theta}} \tag{12}$$

Using Lagrange multipliers λ, we are minimising the function

$$g(\Delta\theta, \lambda) = \frac{1}{2}(J_{\hat{\theta}}\Delta\theta + \hat{u} - u)^\top \Sigma_u^{-1}(J_{\hat{\theta}}\Delta\theta + \hat{u} - u) + \lambda^\top(H^\top\Delta\theta) \tag{13}$$

that has partial derivative with respect λ equal to zero in the optimum (as in Eqn. 9)

$$\frac{\partial g(\Delta\theta, \lambda)}{\partial\lambda} = H^T\Delta\theta = 0 \tag{14}$$

This constraints lead to the *extended normal equations*

$$\begin{bmatrix} M & H \\ H^\top & 0 \end{bmatrix} \begin{bmatrix} \hat{\theta} \\ \lambda \end{bmatrix} = \begin{bmatrix} J_{\hat{\theta}}^\top \Sigma_u^{-1}(\hat{u} - u) \\ 0 \end{bmatrix} \tag{15}$$

and allow us to compute the inversion instead of M-P inversion

$$\begin{bmatrix} \Sigma_{\hat{\theta}} & K \\ K^\top & T \end{bmatrix} = \begin{bmatrix} M & H \\ H^\top & 0 \end{bmatrix}^{-1} \tag{16}$$

5 Solution method

We next describe how to compute the nullspace H and decompose the original Eqn. 16 by a block matrix inversion. The proposed method assumes that the Jacobian of the projection function is provided numerically and provides the nullspace independently of the representation of the camera rotation.

5.1 The nullspace of the Jacobian

The scene can be transformed by a similarity transformation[3]

$${}^s\theta = {}^s\theta(\theta, q) \tag{17}$$

depending on seven parameters $q = [T, s, \mu]$ for translation, rotation, and scale without any change of the projection function $f(\theta) - f({}^s\theta(\theta, q)) = 0$. If we assume a difference similarity transformation, we obtain the total derivative

$$J_\theta \Delta\theta - (J_\theta \Delta\theta + J_\theta J_q \Delta q) = J_\theta J_q \Delta q = 0 \tag{18}$$

Since it needs to hold for any Δq, the matrix

$$H = \frac{\partial {}^s\theta}{\partial q} = J_q \tag{19}$$

is the nullspace of J_θ. Next, consider an order of parameters such that 3D point parameters follow the camera parameters

$$\hat{\theta} = \{P, X\} = \{P_1, \dots P_n, X_1, \dots X_m\} \tag{20}$$

The cameras have parameters ordered as $P_i = \{r_i, C_i, c_i, k_i\}$ and the projection function equals

$$p(\hat{X}_j, \hat{P}_i) = \Phi_i(c_i R(\hat{r}_i)(\hat{X}_j - \hat{C}_i)) \qquad \forall (i, j) \in S \tag{21}$$

where Φ_i projects vectors from $\mathbb{R}^3$ to $\mathbb{R}^2$ by (i) first dividing by the third coordinate, and (ii) then applying image distortion with parameters $\hat{P}_i$. Note that

[3] The variable ${}^s\theta$ is a function of θ and q

function Φ_i can be chosen quite freely, e.g. adding a tangential distortion or encountering a rolling shutter projection model [3]. Using Eqn. 17, we are getting for $\forall(i,j) \in S$

$$p(\hat{X}_j, \hat{P}_i) = p({}^s\hat{X}_j(q), {}^s\hat{P}_i(q)) \tag{22}$$

$$p(\hat{X}_j, \hat{P}_i) = \Phi_i(c_i\, {}^sR(\hat{r}_i, s)({}^s\hat{X}_j(q) - {}^s\hat{C}_i(q))) \tag{23}$$

$$p(\hat{X}_j, \hat{P}_i) = \Phi_i(c_i\, (R(\hat{r}_i)R(s)^{-1})\, ((\mu R(s)\hat{X}_j + T) - (\mu R(s)\hat{C}_i + T))) \tag{24}$$

Note that for any parameters q, the projection remains unchanged. It can be checked by expanding the equation above. Eqn. 24 is linear in T and μ. The differences of $\hat{X}_j$ and $\hat{C}_i$ are as follows

$$\Delta\hat{X}_j(\hat{X}_j, q) = \hat{X}_j - {}^s\hat{X}_j(q) = \hat{X}_j - (\mu R(s)\hat{X}_j + T) \tag{25}$$

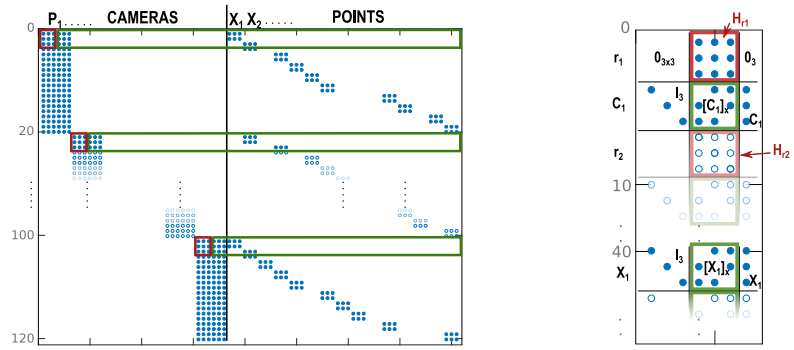
$$\Delta\hat{C}_i(\hat{C}_i, q) = \hat{C}_i - {}^s\hat{C}_i(q) = \hat{C}_i - (\mu R(s)\hat{C}_i + T) \tag{26}$$

The Jacobian $J_{\hat{\theta}}$ and the nullspace H can be written as

$$J_{\hat{\theta}} = \frac{\partial f(\hat{\theta})}{\partial \hat{\theta}} = \begin{bmatrix} \frac{\partial p_1}{\partial \hat{P}_1} \cdots \frac{\partial p_1}{\partial \hat{P}_n} & \frac{\partial p_1}{\partial \hat{X}_1} \cdots \frac{\partial p_1}{\partial \hat{X}_m} \\ \vdots \quad\quad \vdots & \vdots \quad\quad \vdots \\ \frac{\partial p_t}{\partial \hat{P}_1} \cdots \frac{\partial p_t}{\partial \hat{P}_n} & \frac{\partial p_t}{\partial \hat{X}_1} \cdots \frac{\partial p_t}{\partial \hat{X}_m} \end{bmatrix}, \quad H = \begin{bmatrix} H^T_{\hat{P}_1} & H^s_{\hat{P}_1} & H^\mu_{\hat{P}_1} \\ \vdots & \vdots & \vdots \\ H^T_{\hat{P}_n} & H^s_{\hat{P}_n} & H^\mu_{\hat{P}_n} \\ H^T_{\hat{X}_1} & H^s_{\hat{X}_1} & H^\mu_{\hat{X}_1} \\ \vdots & \vdots & \vdots \\ H^T_{\hat{X}_m} & H^s_{\hat{X}_m} & H^\mu_{\hat{X}_m} \end{bmatrix} \tag{27}$$

where p_t is the tth observation, i.e. the pair $(i,j) \in S$. The columns of H are related to transformation parameters q. The rows are related to parameters $\hat{\theta}$. The derivatives of differences of scene parameters $\Delta\hat{P}_i = [\Delta\hat{r}_i, \Delta\hat{C}_i, \Delta\hat{c}_i, \Delta\hat{k}_i]$ and $\Delta\hat{X}_j$ with respect to the transformation parameters $q = [T, s, \mu]$ are exactly the blocks of the nullspace

$$H = \begin{bmatrix} \frac{\partial \Delta r_1}{\partial T} & \frac{\partial \Delta r_1}{\partial s} & \frac{\partial \Delta r_1}{\partial \mu} \\ \frac{\partial \Delta C_1}{\partial T} & \frac{\partial \Delta C_1}{\partial R(s)} & \frac{\partial \Delta C_1}{\partial \mu} \\ \frac{\partial \Delta c_1}{\partial T} & \frac{\partial \Delta c_1}{\partial R(s)} & \frac{\partial \Delta c_1}{\partial \mu} \\ \frac{\partial \Delta k_1}{\partial T} & \frac{\partial \Delta k_1}{\partial R(s)} & \frac{\partial \Delta k_1}{\partial \mu} \\ \vdots & \vdots & \vdots \\ \frac{\partial \Delta X_1}{\partial T} & \frac{\partial \Delta X_1}{\partial R(s)} & \frac{\partial \Delta X_1}{\partial \mu} \\ \vdots & \vdots & \vdots \\ \frac{\partial \Delta X_m}{\partial T} & \frac{\partial \Delta X_m}{\partial R(s)} & \frac{\partial \Delta X_m}{\partial \mu} \end{bmatrix} = \begin{bmatrix} 0_{3\times3} & H_{r_1} & 0_{3\times1} \\ I_{3\times3} & [C_1]_x & C_1 \\ 0_{1\times3} & 0_{1\times3} & 0 \\ 0_{1\times3} & 0_{1\times3} & 0 \\ \vdots & \vdots & \vdots \\ I_{3\times3} & [X_1]_x & X_1 \\ \vdots & \vdots & \vdots \\ I_{3\times3} & [X_m]_x & X_m \end{bmatrix} \tag{28}$$



(a) The Jacobian $J_{\hat{\theta}}$ (b) The nullspace H

Fig. 1: The structure of the matrices $J_{\hat{\theta}}\,H$ for Cube dataset, for clarity, using 6 parameters for one camera $\hat{P}_i$(no focal length and lens distortion shown). The matrices $J_{\hat{r}}$ and $H_{\hat{r}}$ are composed from the red submatrices of J and H. The multiplication of green submatrices equals $-B$, see Eqn. 31.

where $[v]_x$ is the skew symmetric matrix such that $[v]_x\, y = v \times y$ for all $v, y \in \mathbb{R}^3$.

Eqn. 24 is not linear in rotation s. To deal with any rotation representation, we can compute the values of $H_{\hat{r}_i}$ for all i using Eqn. 18. The columns, which contain blocks $H_{\hat{r}_i}$, are orthogonal to the rest of the nullspace and to the Jacobian $J_{\hat{\theta}}$. The system of equations $J_{\hat{\theta}} H = 0$ can be rewritten as

$$J_{\hat{r}} H_{\hat{r}} = B \tag{29}$$

where $J_{\hat{r}} \in \mathbb{R}^{3n \times 3n}$ is composed as a block-diagonal matrix from the red submatrices (see Fig. 1) of $J_{\hat{\theta}}$. The matrix $H_{\hat{r}} \in \mathbb{R}^{3n \times 3}$ is composed from red submatrices $H_{\hat{r}_i} \in \mathbb{R}^{3n \times 3}$ as

$$H_{\hat{r}} = \left[H_{\hat{r}_1}^\top \ldots H_{\hat{r}_n}^\top\right]^\top \tag{30}$$

The matrix $B \in \mathbb{R}^{3n \times 3}$ is composed of the green submatrices (see Fig. 1) of $J_{\hat{\theta}}$ multiplied by the minus green submatrices of H. The solution to this system is

$$H_{\hat{r}} = J_{\hat{r}}^{-1} B \tag{31}$$

where B is computed by a sparse multiplication, see Fig. 1. The inversion of $J_{\hat{r}}$ is the inversion of a sparse matrix with n blocks $\mathbb{R}^{3\times 3}$ on the diagonal.

5.2 Uncertainty propagation to camera parameters

The propagation of uncertainty is based on Eqn. 16. The inversion of extended Fisher information matrix is first conditioned for better numerical accuracy as follows

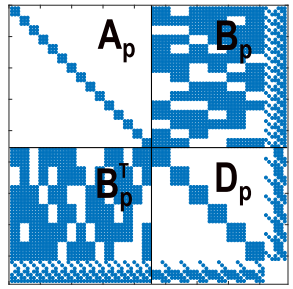


Fig. 2: The structure of the matrix Q_p for Cube dataset and $\hat{P}_i \in \mathbb{R}^6$.

$$\begin{bmatrix} \Sigma_{\hat{\theta}} & K \\ K^\top & T \end{bmatrix} = \begin{bmatrix} S_a & 0 \\ 0 & S_b \end{bmatrix} \left(\begin{bmatrix} S_a & 0 \\ 0 & S_b \end{bmatrix} \begin{bmatrix} M & H \\ H^\top & 0 \end{bmatrix} \begin{bmatrix} S_a & 0 \\ 0 & S_b \end{bmatrix} \right)^{-1} \begin{bmatrix} S_a & 0 \\ 0 & S_b \end{bmatrix} \tag{32}$$

$$\begin{bmatrix} \Sigma_{\hat{\theta}} & K \\ K^\top & T \end{bmatrix} = \begin{bmatrix} S_a & 0 \\ 0 & S_b \end{bmatrix} \begin{bmatrix} M_s & H_s \\ H_s^\top & 0 \end{bmatrix}^{-1} \begin{bmatrix} S_a & 0 \\ 0 & S_b \end{bmatrix} \tag{33}$$

$$\begin{bmatrix} \Sigma_{\hat{\theta}} & K \\ K^\top & T \end{bmatrix} = SQ^{-1}S \tag{34}$$

by diagonal matrices S_a,S_b which condition the columns of matrices J, H. Secondly, we permute the columns of Q to have point parameters followed by the camera parameters

$$\begin{bmatrix} \Sigma_{\hat{\theta}} & K \\ K^\top & T \end{bmatrix} = S\widetilde{P}(\widetilde{P}Q\widetilde{P})^{-1}\widetilde{P}S = S\widetilde{P}Q_p^{-1}\widetilde{P}S \tag{35}$$

where $\widetilde{P}$ is an appropriate permutation matrix. The matrix $Q_p = \widetilde{P}Q\widetilde{P}$ is a full rank matrix which can be decomposed and inverted using a block matrix inversion

$$Q_p^{-1} = \begin{bmatrix} A_p & B_p \\ B_p^\top & D_p \end{bmatrix}^{-1} = \begin{bmatrix} A_p^{-1} + A_p^{-1}BZ_p^{-1}B_p^\top A_p^{-1} & -A_p^{-1}BZ_p^{-1} \\ -Z_p^{-1}B_p^\top A_p^{-1} & Z_p^{-1} \end{bmatrix} \tag{36}$$

where Z_p is the symmetric Schur complement matrix of point parameters block A_p

$$Z_p^{-1} = (D_p - B_p^\top A_p^{-1} B_p)^{-1} \tag{37}$$

Matrix $A_p \in \mathbb{R}^{3m\times 3m}$ is a sparse symmetric block diagonal matrix with $\mathbb{R}^{3\times 3}$ blocks on the diagonal, see Fig. 2. The covariances for camera parameters are computed using the inversion of Z_p with the size $\mathbb{R}^{(8n+7)\times(8n+7)}$ for our model of cameras (i.e., $P_i \in \mathbb{R}^8$)

$$\Sigma_{\hat{P}} = S_P Z_s S_P \tag{38}$$

where $Z_s \in \mathbb{R}^{8n\times 8n}$ is the left top submatrix of Z_p^{-1} and S_P is the corresponding sub-block of scale matrix S_a.

6 Uncertainty for sub-reconstructions

The algorithm based on Gauss-Markov estimate with constraints, which is described in Section 5, works in principle properly for thousands of cameras. However, large-scale reconstructions with thousands cameras would require a large space, e.g. 131GB for Rome dataset [20], to store the matrix Z_p for our camera model $\hat{P}_i \in \mathbb{R}^8$, and its inversion might be inaccurate due to rounding errors.

Fortunately, it is possible to evaluate the uncertainty of a camera $\hat{P}_i$ from only a partial sub-reconstruction comprising cameras and points in the vicinity of $\hat{C}_i$. Using sub-reconstructions, we can approximate the uncertainty computed from a complete reconstruction. The error of our approximation decreases with increasing size of a sub-reconstruction. If we add a camera to a reconstruction, we add at least four observations which influence the Fisher information matrix M_i as

$$M_{i+1} = M_i + M_\Delta \tag{39}$$

where the matrix M_Δ is the Fisher information matrix of the added observations. We can propagate this update using equations in Section 5 to the Schur complement matrix

$$Z_{i+1} = Z_i + Z_\Delta \tag{40}$$

which has full rank. Using Woodbury matrix identity

$$(Z_i + J_\Delta^\top \Sigma_\Delta J_\Delta)^{-1} = Z_i^{-1} - Z_i^{-1} J_\Delta^\top (I + J_\Delta Z_i J_\Delta^\top)^{-1} J_\Delta Z_i^{-1} \tag{41}$$

we can see that the positive definite covariance matrices are subtracted after adding some observations, i.e. the uncertainty decreases.

We show empirically that the error decreases with increasing the size of the reconstruction (see Fig. 3). We have found that for 100–150 neighbouring cameras, the error is usually small enough to be used in practice. Each evaluation of the sub-reconstruction produces an upper bound on the uncertainty for cameras involved in the sub-reconstruction. The accuracy of the upper bound depends on a particular decomposition of the complete reconstruction into sub-reconstructions. To get reliable results, it is useful to decompose the reconstruction several times and choose the covariance matrix with the smallest trace.

The theoretical proof of the quality of this approximation and selection of the optimal decomposition is an open question for future research.

7 Experimental evaluation

We use synthetic as well as real datasets (Table 1) to test and compare the algorithms (Table 2) with respect the accuracy (Fig. 3) and speed (Fig. 4). The evaluations on sub-reconstructions are shown in Figs. 5, 6a, 6b. All experiments were performed on a single computer with one 2.6GHz Intel Core i7-6700HQ with 32GB RAM running a 64-bit Windows 10 operating system.

Table 1: Summary of the datasets: N_P is the number of cameras, N_X is the number of points in 3D and N_u is the number of observations. Datasets 1 and 3 are synthetic, 2, 9 from COLMAP [30], and 4-8 from Bundler [31]

#	Dataset	N_P	N_X	N_u
1	Cube	6	15	60
2	Toy	10	60	200
3	Flat	30	100	1033
4	Daliborka	64	200	5205
5	Marianska	118	80 873	248 511
6	Dolnoslaskie	360	529 829	226 0026
7	Tower of London	530	65 768	508 579
8	Notre Dame	715	127 431	748 003
9	Seychelles	1400	407 193	2 098 201

Table 2: The summary of used algorithms

#	Algorithm
1.	M-P inversion of M using Maple (Kanatani [18]) (**Ground Truth**)
2.	M-P inversion of M using Ceres (Kanatani [18])
3.	M-P inversion of M using Matlab (Kanatani [18])
4.	M-P inversion of Schur complement matrix with correction term (Lhuillier [19])
5.	TE inversion of Schur complement matrix with three points fixed (Polic [27])
6.	**Nullspace bounding uncertainty propagation (NBUP)**

Compared algorithms are listed in Table 2. The standard way of computing the covariance matrix $\Sigma_{\hat{P}}$ is by using the M-P inversion of the information matrix using the Singular Value Decomposition (SVD) with the last seven singular values set to zeros and inverting the rest of them as in [26]. There are many implementations of this procedure that differ in numerical stability and speed. We compared three of them. Alg. 1 uses high precision number representation in Maple (runs 22 hours on Daliborka dataset), Alg. 2 denotes the implementation in Ceres [2], which uses Eigen library [11] internally (runs 25.9 minutes on Daliborka dataset) and Alg. 3 is our Matlab implementation, which internally calls LAPACK library [4] (runs 0.45 seconds on Daliborka dataset). Further, we compared Lhuilier [19] and Polic [26] approaches, which approximate the uncertainty propagation, with our algorithm denoted as *Nullspace bounding uncertainty propagation* (NBUP).

The accuracy of all algorithms is compared against the Ground Truth (GT) in Fig. 3. The evaluation is performed on the first four datasets which have reasonably small number of 3D points. The computation of GT for the fourth

dataset took about 22 hours and larger datasets were uncomputable because of time and memory requirements. We decomposed information matrix using SVD, set exactly the last seven singular values to zero and inverted the rest of them. We also used 100 significant digits instead of 15 digits used by a double number representation. The GT computation follows approach from [26].

The covariance matrices for our camera model (comprising rotation, camera center, focal length and radial distortion) contain a large range of values. Some parameters, e.g. rotations represented by the Euler vector, are in units while other parameters, as the focal length, are in thousands of units. Moreover, the rotation is in all tested examples better constrained than the focal length. This fact leads to approximately $6{\times}10^{-5}$ mean absolute value in rotation part of the covariance matrix and approximately $3{\times}10^{4}$ mean value for the focal length variance. Standard deviations for datasets 1-4 and are about $8{\times}10^{-3}$ for rotations and $2{\times}10^{3}$ for focal lengths. To obtain comparable standard deviations for different parameters, we can divide the mean values of rotations by π and focal length by $2{\times}10^{3}$. We used the same approach for the comparison of the measured errors

$$err_{\hat{P}_i} = \frac{1}{64}\sum_{l=1}^{8}\sum_{m=1}^{8}\left(\sqrt{|\widetilde{\Sigma}_{\hat{P}_i(l,m)} - \widehat{\Sigma}_{\hat{P}_i(l,m)}|} \oslash O_{(l,m)}\right) \tag{42}$$

The error $err_{\hat{P}_i}$ shows the differences between GT covariance matrices $\widetilde{\Sigma}_{\hat{P}_i}$ and the computed ones $\hat{\Sigma}_{\hat{P}_i}$. The matrix

$$O = \sqrt{E(|\hat{P}_i|)\, E(|\hat{P}_i|)^\top} \tag{43}$$

has dimension $O \in \mathbb{R}^{8\times 8}$ and normalises the error to percentages of the absolute magnitude of the original units. Symbol $\oslash$ stands for element-wise division of matrices (i.e. $\bar{C} = \bar{A} \oslash \bar{B}$ equals $\bar{C}_{(i,j)} = \bar{A}_{(i,j)}/\bar{B}_{(i,j)}$ for $\forall(i,j)$).

Fig. 3 shows the comparison of the mean of the errors for all cameras in the datasets. We see that our new method, NBUP, delivers the most accurate results on all datasets.

Speed of the algorithms is shown in Fig. 4. Note that the M-P inversion (i.e. Alg. 1-3) cannot be evaluated on medium and larger datasets 5-9 because of memory requirements for storing dense matrix M. We see that our new method NBUP is faster than all other methods. Considerable speedup is obtained on datasets 7-9 where our NBUP method is about 8 times faster.

Uncertainty approximation on sub-reconstructions was tested on datasets 5-9. We decomposed reconstructions several times using a different number of cameras $\bar{k} = \{5, 10, 20, 40, 80, 160, 320\}$ inside smaller sub-reconstructions, and measured relative and absolute errors of approximated covariances for cameras parameters. Fig. 6 shows the decrease of error for larger sub-reconstructions. There were 25 sub-reconstructions for each $\bar{k}_i$ with the set of neighbouring cameras randomly selected using the view graph. Note that Fig. 6a shows the mean

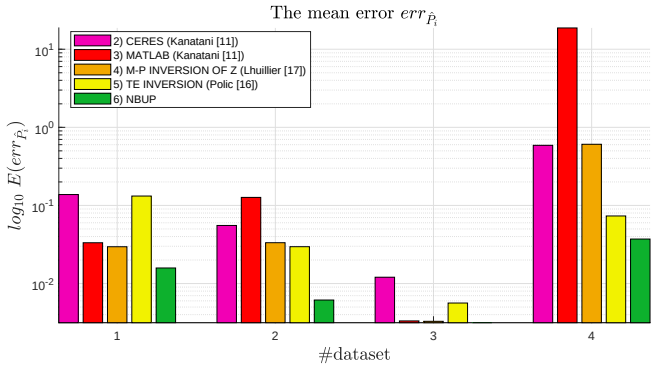


Fig. 3: The mean error $err_{\hat{P}_i}$ of all cameras $\hat{P}_i$ and Alg. 2-6 on datasets 1-4. Note that the Alg. 3, leading to the normal form of the covariance matrix, is numerically much more sensitive. It sometimes produces completely wrong results even for small reconstructions.

of relative errors given by Eqn. 42. Fig. 6b shows that the absolute covariance error decreases significantly with increasing the number of cameras in a sub-reconstruction.

Fig. 5 shows the error of the simplest approximation of covariances used in practice. For every camera, one hundred of its neighbours using view-graph were used to get a sub-reconstruction for evaluating the uncertainties. It produces upper bound estimates for the covariances for each camera from which we selected the smallest one, i.e. the covariance matrix with the smallest trace, and evaluate the mean of the relative error $err_{\hat{P}_i}$.

8 Conclusions

Current methods for evaluating of the uncertainty [19],[26] in SfM rely 1) either on imposing the gauge constraints by using a few parameters as observations, which does not lead to the natural form of the covariance matrix, or 2) on the Moore-Penrose inversion [2], which cannot be used in case of medium and large-scale datasets because of cubic time and quadratic memory complexity.

We proposed a new method for the nullspace computation in SfM and combined it with Gauss Markov estimate with constraints [29] to obtain a full-rank matrix [9] allowing robust inversion. This allowed us to use efficient methods from SLAM such as block matrix inversion or Woodbury matrix identity. Our approach is the first one which allows a computation of natural form of the covariance matrix on scenes with more than thousand of cameras, e.g. 1400 cameras, with affordable computation time, e.g. 60 seconds, on a standard PC. Further, we show that using sub-reconstruction of roughly 100-300 cameras provides reliable estimates of the uncertainties for arbitrarily large scenes.

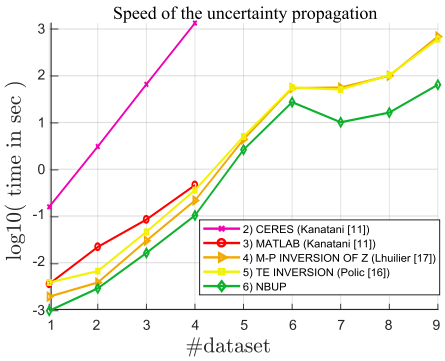


Fig. 4: The speed comparison. Full comparison against Alg. 2, 3 was not possible because of the memory complexity. Alg. 3 failed, see Fig. 3.

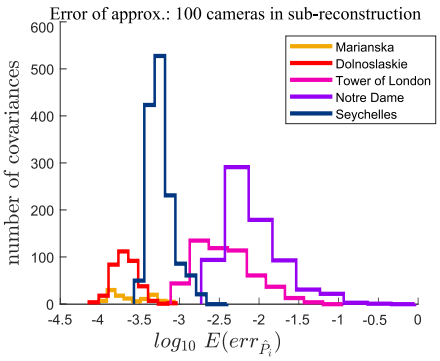
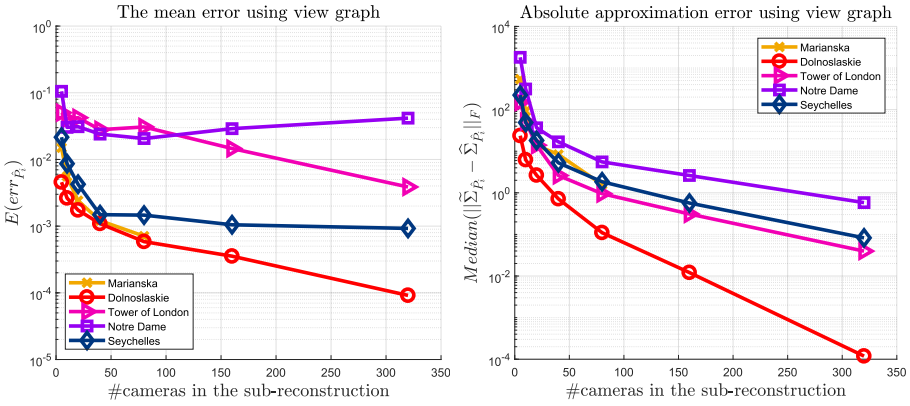


Fig. 5: The relative error for approximating camera covariances by one hundred of their neighbours from the view-graph.

9 Acknowledgement

This work was supported by the European Regional Development Fund under the project IMPACT (reg. no. CZ.02.1.01/0.0/0.0/15_003/0000468), EU-H2020 project LADIO no. 731970, and by Grant Agency of the CTU in Prague projects SGS16/230/OHK3/3T/13, SGS18/104/OHK3/1T/37.



(a) Mean of relative error $err_{\hat{P}_i}$

(b) Median of absolute error

Fig. 6: The error of the uncertainty approximation using sub-reconstructions as a function of the number of cameras in the sub-reconstruction.

References

1. Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S.M., Szeliski, R.: Building rome in a day. Communications of the ACM **54**(10), 105–112 (2011)
2. Agarwal, S., Mierle, K., Others: Ceres solver. http://ceres-solver.org
3. Albl, C., Kukelova, Z., Pajdla, T.: R6p-rolling shutter absolute camera pose. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2292–2300 (2015)
4. Anderson, E., Bai, Z., Dongarra, J., Greenbaum, A., McKenney, A., Du Croz, J., Hammarling, S., Demmel, J., Bischof, C., Sorensen, D.: Lapack: A portable linear algebra library for high-performance computers. In: Proceedings of the 1990 ACM/IEEE conference on Supercomputing. pp. 2–11. IEEE Computer Society Press (1990)
5. Besl, P.J., McKay, N.D.: Method for registration of 3-d shapes. In: Sensor Fusion IV: Control Paradigms and Data Structures. vol. 1611, pp. 586–607. International Society for Optics and Photonics (1992)
6. Eves, H.W.: Elementary matrix theory. Courier Corporation (1966)
7. Förstner, W.: Image Matching, vol. II, chap. 16, pp. 289–379. Addison Wesley (1993)
8. Förstner, W.: Uncertainty and projective geometry. In: Handbook of Geometric Computing, pp. 493–534. Springer (2005)
9. Förstner, W., Wrobel, B.P.: Photogrammetric Computer Vision. Springer (2016)
10. Frahm, J.M., Fite-Georgel, P., Gallup, D., Johnson, T., Raguram, R., Wu, C., Jen, Y.H., Dunn, E., Clipp, B., Lazebnik, S., et al.: Building rome on a cloudless day. In: European Conference on Computer Vision. pp. 368–381. Springer (2010)
11. Guennebaud, G., Jacob, B., et al.: Eigen v3.3. http://eigen.tuxfamily.org (2010)
12. Hager, W.W.: Updating the inverse of a matrix. SIAM review **31**(2), 221–239 (1989)
13. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge university press (2003)
14. Heinly, J., Schönberger, J.L., Dunn, E., Frahm, J.M.: Reconstructing the World* in Six Days *(As Captured by the Yahoo 100 Million Image Dataset). In: Computer Vision and Pattern Recognition (CVPR) (2015)
15. Ila, V., Polok, L., Solony, M., Istenic, K.: Fast incremental bundle adjustment with covariance recovery. In: International Conference on 3D Vision (3DV) (Oct 2017)
16. Ila, V., Polok, L., Solony, M., Svoboda, P.: Slam++-a highly efficient and temporally scalable incremental slam framework. The International Journal of Robotics Research **36**(2), 210–230 (2017)
17. Kaess, M., Dellaert, F.: Covariance recovery from a square root information matrix for data association. Robotics and autonomous systems **57**(12), 1198–1210 (2009)
18. Kanatani, K.i., Morris, D.D.: Gauges and gauge transformations for uncertainty description of geometric structure with indeterminacy. IEEE Transactions on Information Theory **47**(5), 2017–2028 (2001)
19. Lhuillier, M., Perriollat, M.: Uncertainty ellipsoids calculations for complex 3d reconstructions. In: Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006. pp. 3062–3069. IEEE (2006)
20. Li, Y., Snavely, N., Huttenlocher, D.P.: Location recognition using prioritized feature matching. In: European conference on computer vision. pp. 791–804. Springer (2010)

21. Maurer, M., Rumpler, M., Wendel, A., Hoppe, C., Irschara, A., Bischof, H.: Georeferenced 3d reconstruction: Fusing public geographic data and aerial imagery. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on. pp. 3557–3558. IEEE (2012)
22. Morris, D.D.: Gauge freedoms and uncertainty modeling for 3 D computer vision. Ph.D. thesis, Citeseer (2001)
23. Moulon, P., Monasse, P., Marlet, R., Others: Openmvg. an open multiple view geometry library. https://github.com/openMVG/openMVG
24. Nashed, M.Z.: Generalized Inverses and Applications: Proceedings of an Advanced Seminar Sponsored by the Mathematics Research Center, the University of WisconsinMadison, October 8-10, 1973. No. 32, Elsevier (2014)
25. Polic, M.: 3d scene analysis. http://cmp.felk.cvut.cz/~policmic
26. Polic, M., Pajdla, T.: Camera uncertainty computation in large 3d reconstruction. In: International Conference on 3D Vision (2017)
27. Polic, M., Pajdla, T.: Uncertainty computation in large 3d reconstruction. In: Scandinavian Conference on Image Analysis. pp. 110–121. Springer (2017)
28. Polok, L., Ila, V., Smrz, P.: 3d reconstruction quality analysis and its acceleration on gpu clusters. In: 2016 24th European Signal Processing Conference (EUSIPCO). pp. 1108–1112 (Aug 2016). https://doi.org/10.1109/EUSIPCO.2016.7760420
29. Rao, C.R., Rao, C.R., Statistiker, M., Rao, C.R., Rao, C.R.: Linear statistical inference and its applications, vol. 2. Wiley New York (1973)
30. Schönberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
31. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3d. In: ACM transactions on graphics (TOG). vol. 25, pp. 835–846. ACM (2006)
32. Sweeney, C.: Theia multiview geometry library: Tutorial & reference. http://theia-sfm.org
33. Tian, Y.: The moore-penrose inverses of m$\times$ n block matrices and their applications. Linear algebra and its applications **283**(1), 35–60 (1998)
34. Zhang, F.: The Schur Complement and Its Applications. Springer US (2005)