

Uncertainty Computation in Large 3D Reconstruction

Michal Polic^(✉) and Tomas Pajdla

Department of Cybernetics, Faculty of Electrical Engineering,
Center for Machine Perception, Czech Technical University in Prague,
Prague, Czech Republic
policmic@fel.cvut.cz
<http://cmp.felk.cvut.cz/~policmic>

Abstract. Many automatic methods for reconstructing camera motion and scene geometry from a large number of images evaluate the quality of the reconstruction by error propagation from measurements to the estimated parameters. Unfortunately, uncertainty propagation is computationally challenging for large scenes and hence cannot be used for large scenes in practice. We present a new algorithm for efficient uncertainty propagation which works with millions of feature points, thousands of cameras and millions of 3D points on a single computer and achieves about twenty times speedup.

Keywords: Uncertainty propagation · 3D reconstruction · Schur complement · Information matrix · Taylor expansion

1 Introduction

Recent work in Structure from Motion (SfM) has demonstrated the possibility of reconstructing geometry from large photo collections [1, 16]. Efficient non-linear refinement [3, 34] of camera and point parameters was developed to produce optimal reconstruction. However, the important information about the reconstruction quality [12] is missing in state of the art pipelines [28, 33] and optimization solvers [2] because of the computational and memory requirements. If the quality evaluation is fast enough with reasonable memory requirements, it can be used for filtering cameras and 3D points before Bundle Adjustment which usually run in each step of SfM and therefore remove the bottleneck of a reconstruction pipelines [28, 29]. The second usage may be sophisticated smoothing in the most unconstrained directions in dense reconstructions [19, 27].

The mathematics of uncertainty propagation is well understood [15], and specific equations have been derived for many camera setups, e.g. for stereo rigs [22, 25], laser scans [17, 26], lines [4] and edges [6]. Except for few a works, e.g. [20], general error propagation has not received significant attention in the computer vision community. We believe that it is due to the high computational complexity of uncertainty propagation, which is, as a function of the number

of camera parameters, cubic in time and quadratic in memory. To evaluate the quality, classical photogrammetric methods propagate the uncertainty of image measurements to the uncertainty of estimated parameters.

General propagation requires transport of covariances of measurements to parameters. The standard non-linear projection function is usually simplified by replacing it by its first order approximation, and the covariance of the estimated parameters is then computed by inverting its information matrix. Common objective function [15] of reconstruction from images is the sum of squared reprojection errors.

Camera rotation angle and radial distortion parameters are usually much smaller than coordinates of 3D points and, at the same time, have a large impact on the objective function. Hence, the Jacobian of the objective function often contains a large range of values which are further squared to information matrix. Thus, the information matrix becomes numerically rank deficient for larger collections of images. The information matrix can be decomposed into blocks which simplify its inversion. The block related to 3D points parameters is, up to special artificial configurations, full rank block diagonal matrix which can be easily inverted. The block related to camera parameters appears as over-parametrised. We have to find the closest one-one mapping from measurements to parameters with respect the Mahalanobis norm to be able to inverse submatrix of camera parameters [15]. The inversion should be computed as Moore-Penrose (MP) inversion [2]. The state of the art algorithm use Singular Value Decomposition (SVD) [20]. This algorithm works well for problems with a few hundreds of photos. As the number of images in reconstructions grows up to several hundred thousand, the efficiency and scalability of the quality evaluation become a critical issue.

We present an iterative algorithm based on the Taylor expansion of Schur complement matrix Z in a regularization point λ , which computes an approximation of the MP inverse. This algorithm is approximately three times faster and reduces approximately two times the memory requirements in comparison with SVD. The second improvement is that we use the structure of matrices for uncertainty propagation from camera parameters to the point parameters, which increases the speed about two orders of magnitude for this particular step for real datasets. The output of our work is publicly available source code which can be used as an external library in nonlinear optimization pipelines like Ceres Solver [2]. The code, test problems, and detailed experiments are available online at cmp.felk.cvut.cz/~policmic.

The rest of the paper is organized as follow. Section 2 overviews previous works. Then, Sect. 3 provides our formulation of the problem of uncertainty propagation and Sect. 4 introduced our Taylor expansion algorithm. Section 5 describe how to use the structure of information matrix to compute covariances of point parameters. Finally, Sect. 6 reports the results of the experiments.

2 Previous Work

Many methods for 3D reconstruction of camera motion and scene geometry from a large number of images have been developed [1, 13, 16]. To evaluate the quality of the reconstruction, classical photogrammetric methods propagate the uncertainty in image measurements to the uncertainty of estimated parameters. The propagation is often based on the first order approximations [12, 20, 21]. This classical approach becomes computationally challenging with a large number of cameras and points [30, 32] and hence is often not fully implemented in computer vision reconstruction pipelines [2].

Let us next review previous work related to computing uncertainties in 3D reconstruction from images. Blostein and Huang [8] provided closed form expressions for evaluating parameters of error distributions of the disparity for a stereo rig. The confidence parameters for stereo setup were presented in [18]. Work [25] minimizes the Mean Square Error of measurement errors to compute optimal camera poses and to compute covariance matrices representing their uncertainties. The influence of errors in calibration on the errors in 3D reconstruction for stereo setup was investigated in [22]. In the recent paper [5], uncertainties of estimated parameters are studied for each step of a stereo-vision based 3D reconstruction (e.g., for calibration, contour segmentation, matching, and reconstruction), and used to estimate the error distributions of the output geometry. Inverse perspective equations for describing 3D line error were investigated in [4] and later generalized for reconstruction from edges in images [6]. In [23] the uncertainties are expressed as likelihood map of 3D points position created by aggregating local linear extrapolations computed from weighted least squares fits of 2D curves between 3D points.

The most relevant previous work. The most relevant work is from Lhuillier and Perriollat [20] where forming information matrix of estimated parameters and its inverse has been investigated. Lhuillier has computed MP inverse of Schur complement [35] of the submatrix of point parameters using the SVD algorithm. We extend Lhuillier work using faster MP inverse and improve 3D points covariance computation speed using the structure of Jacobian. A general algorithm for covariance propagation forward/backward for system linear/nonlinear which is or isn't overparameterized was also described by Hartley in the book [15].

3 Problem Formulation

We next review the standard approach for computing covariances of reconstructed parameters. Our contributions are described in Sects. 4 and 5.

We consider a setup with n cameras $C = \{C_1, C_2, \dots, C_n\}$ where $C_i \in \mathbb{R}^p$ is p -dimensional vector of camera parameters, m points $X = \{X_1, X_2, \dots, X_m\}$ in 3D and k image observations represented by vector $\mathbf{u} \in \mathbb{R}^{2k}$. Each observation $\mathbf{u}_{i,j} \in \mathbf{u}$, i.e. an image point, is a projection of 3D point X_j by camera C_i , using projection function $\mathbf{p}(C_i, X_j)$. Noise $\epsilon_{i,j}$ has zero mean, $E(\epsilon_{ij}) = 0$, and

standard deviation $V(\epsilon_{i,j})$. All pairs of indices (i, j) are in an index set S that determines which point X_j is visible in which camera view C_i .

$$\mathbf{u}_{i,j} = \mathbf{p}(C_i, X_j) + \epsilon_{i,j} \quad \forall (i, j) \in S \quad (1)$$

Further, we define the vector θ which represents all 3D reconstruction parameters $[C_1, \dots, C_n, X_1, \dots, X_m]$, the vector ϵ which gathers all $\epsilon_{i,j}$ where $(i, j) \in S$ and the function f which is composed from projection functions \mathbf{p} and project vector θ into the image observations \mathbf{u} .

$$\mathbf{u} = f(\theta) + \epsilon \quad (2)$$

The function (2) leads to a non-linear least squares optimization

$$\theta^* = \arg \min_{\theta} \|\mathbf{u} - f(\theta)\|_2^2 \quad (3)$$

minimizing the sum of squares of the differences $r(\theta) = \mathbf{u} - f(\theta)$ between observations \mathbf{u} and reprojections $f(\theta)$. The standard way to solve the nonlinear differentiable system is to use its first order approximation

$$r(\theta + \delta\theta) \approx r(\theta) + J(\theta) \delta\theta \quad (4)$$

One step $\delta\theta$ towards the solution θ^* is obtained by solving

$$J^\top J \delta\theta = -J^\top r \quad (5)$$

with $J(\theta)$ and $r(\theta)$ replaced by simpler J and r . Using Gauss-Markoff theorem, the covariance $\hat{V}(\theta)$ can be computed as

$$\hat{V}(\theta) = \sigma^2 (J^\top J)^{-1} \quad (6)$$

Parameter σ can be estimated from residuals as $\sigma^2 \approx \|r\|^2 / (pn)$, which is described in Hartley [15]. If the content of Jacobian is permuted to have cameras followed by points, i.e. $J = [J_C \ J_X]$, the information matrix

$$(J^\top J)^{-1} = \begin{bmatrix} U & W \\ W^\top & V \end{bmatrix}^{-1} \quad (7)$$

will be sparse and block diagonal [11]. See the information matrix for *cube* dataset in Fig. 1(a).

To compute the inverse of $J^\top J$, we introduce $Y = -V^{-1}W^\top$. We note, first, that V is composed of 3×3 blocks on the diagonal and its inverse can be done separately for each block, and then also that forming Y is fast thanks to the sparsity of V and W . The Upper triangular–Diagonal–Lower triangular (UDL) decomposition of the block matrix $\hat{V}(\theta) = \sigma^2 (J^\top J)^{-1}$ leads to

$$\hat{V}(\theta) = \sigma^2 \left(\begin{bmatrix} I & -Y^\top \\ 0 & I \end{bmatrix} \begin{bmatrix} Z & 0 \\ 0 & V \end{bmatrix} \begin{bmatrix} I & 0 \\ -Y & I \end{bmatrix} \right)^{-1} \quad (8)$$

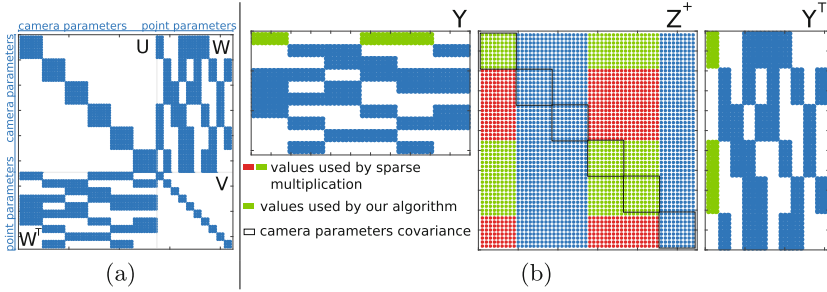


Fig. 1. (a) Structure of information matrix $J^T J$ for *cube* dataset. (b) The structure of matrix Y , Z^+ and Y^T for *cube* dataset, where sparse multiplication use red and green values, our approach use green values only and the black boxes highlight final covariances of camera parameters (Color figure online)

where the matrix Z is the Schur complement of the block V of the matrix $J^T J$

$$Z = U + WY \quad (9)$$

Notice that we are not interested in off-diagonal blocks. All covariances of parameters are in the blocks on the diagonal of the dense matrix $(J^T J)^{-1}$. Hence, Lhuillier and Perriollat [20] computed the interesting sub-matrices as

$$(J^T J)^{-1} \simeq \begin{bmatrix} Z^{-1} & - \\ - & YZ^{-1}Y^T + V^{-1} \end{bmatrix} \quad (10)$$

The blocks of size $\mathbb{R}^{p \times p}$ on the diagonal Z^{-1} are covariances of camera parameters. Matrix Z is rank deficient and its inversion has to be solved by MP inversion. The blocks of size $\mathbb{R}^{3 \times 3}$ on the diagonal of sub-matrix $YZ^{-1}Y^T + V^{-1}$ are covariances of point parameters.

4 The Taylor Expansion Approach

We next denote the MP inverse of matrix Z as Z^+ . Assume general vectors x and b such that

$$Zx = b \quad (11)$$

holds true. Our approach is to redefine x as a function where

$$Zx(0) = b \quad (12)$$

holds true. Unfortunately, the pseudoinverse can't be computed as $(Z^T Z)^{-1} Z$ since $Z^T Z$ is also rank deficient. Hence, we compute the inversion for $x(\lambda)$ as

$$x(\lambda) = (Z^T Z + \lambda I)^{-1} Z^T b \quad (13)$$

where λ is close to zero. Finally, we estimate $x(0)$ and Z^+ from Taylor series of function $x(\lambda)$ at the point 0.

We need a general formula for the k -th derivative with respect λ

$$x^k(\lambda) = -k(ZZ + \lambda I)^{-1} x^{k-1}(\lambda) \quad (14)$$

for building Taylor series. The matrix Z is symmetric and $Z^\top Z = ZZ$. Using the derivatives, we can approximate $x(\lambda)$ around λ using the t -th order Taylor series

$$x(0) = x(\lambda) + \sum_{k=1}^t \frac{(-1)^{k-1} \lambda^k}{k!} x^k(\lambda). \quad (15)$$

Next we substitute $B = (ZZ + \lambda I)^{-1}$, which is computed using Cholesky decomposition [10]. $x(\lambda) = BZ$ can be factored out and we get the formula for Z^+ as

$$x(0) = Z^+ b \quad Z^+ = \left(I + \sum_{k=1}^t (-1)^{k-1} \lambda^k B^k \right) BZ \quad (16)$$

In our experiments, the algorithm converges for $t \in [2, 5]$. Each iteration performs one symmetric matrix multiplication and one addition. If we scale the matrix ZZ by scalar $c = 1/\text{mean}(ZZ)$, we get $\hat{\lambda} = c\lambda$, $\hat{B} = (c\hat{Z}\hat{Z} + \hat{\lambda}I)^{-1}$ and

$$Z^+ = \left(I + \sum_{k=1}^t (-1)^{k-1} c^k \hat{\lambda}^k \hat{B}^k \right) \hat{c} \hat{B} \hat{Z} \quad (17)$$

This scaling decrease the range of values and allows computation in double precision. Numerical stability of the algorithm depends on the value of $\hat{\lambda}$. We use $\hat{\lambda} = \text{trace}(ZZ) c/10^{16}$ which is usually $[10^{-8}, 10^{-10}]$. The parameter $\hat{\lambda}$ depends on the size of Schur complement matrix which depends on the number of parameters of cameras. Reconstructions, which contains more than 10^6 cameras, may require higher parameter $\hat{\lambda}$, i.e. above 10^{-4} .

5 Point Covariances

We compute the covariance matrices of point parameters from Eq. 10 using the structure of the matrix Y . The matrix Y contains values in blocks where 3D points (triples of rows) are seen by cameras (p columns). We compute only blocks $\bar{H} = Z^+ Y^\top$ which are required for multiplication $Y \bar{H}$, Fig. 1(b). The speed up in comparison with sparse multiplication is $n m/k$ times.

6 Experimental Evaluation

6.1 Datasets

We experimented with realistic synthetic scenes, as well as a number of medium to large-scale Internet datasets: Flat, Tower of London, Notre Dame, Trench, Seychelles. The Cube dataset was used for visualization of the matrices. Flat is

Table 1. The table summarize number of cameras N_{Cams} , number of points N_{Pts} , number of observations N_{Obs} and one side length N_{Schur} of rectangular Schur complement matrix Z for compared datasets

Dataset	SfM	N_{Cams}	N_{Pts}	N_{Obs}	N_{Schur}
Flat	Synthetic	361	251	10 517	3249
Tower of London	COLORMAP	530	65 768	508 579	4770
Trench	YaSfM	698	434 049	1 017 909	6282
Notre Dame	Bundler	715	127 431	748 003	6435
Seychelles	COLORMAP	1400	407 193	2 098 201	12600

the only dataset that is synthetic. Tower of London and Seychelles are reconstructed with COLORMAP [28], Notre Dame is reconstructed by Bundler [29] and the remaining one by our internal reconstruction pipeline YaSfM. The images for Tower of London and Notre Dame were collected in projects [29, 32] from Flickr by using geotag. The remaining datasets are available on web page cmp.felk.cvut.cz/~policmic. The parameters of datasets are summarized in Table 1.

6.2 Algorithms

We compared the performance, memory requirements and the sensitivity on input parameters for three algorithms: qr-iteration, divide-and-conquer and taylor-expansion. The state of the art method is done using SVD [20]. The algorithms qr-iteration, divide-and-conquer are described in [7]. We do not compare our algorithm with recent [24] paper because the Cholesky decomposition on Schur complement matrix is performed there. However, the Schur complement matrix is numerically rank deficient for middle and large reconstructions (i.e. Cholesky decomposition can't be used). This subsection has following structure: we describe how the Schur complement matrix was formed, how its MP inversion was computed and lastly focus on the precision of Taylor expansion approach. Let us pinpoint the main figures and tables: Fig. 2 shows the structure of C++ code and labels used for comparison of the speed. The comparisons of the speeds are in Tables 2, 3. The Fig. 3 visualize the uncertainty ellipsoids for different values of parameters γ, λ .

Forming Schur Complement. Given Jacobian $J(\theta) = [J_C(\theta) J_X(\theta)]$ from Ceres [2], we form an information matrix in Eq. 7 and Schur complement in Eq. 9 using sparse matrix multiplication for all algorithms. For each experiment, $\theta = [\theta_C \theta_X]$ were $\theta_C = [\theta_{C_1}, \dots, \theta_{C_n}]$ and $\theta_X = [\theta_{X_1}, \dots, \theta_{X_m}]$. One camera $\theta_{C_i} \in \mathbb{R}^9$ is represented by 3 angle-axis, 3 camera center, 1 focal length, 2 radial distortion parameters and point $\theta_{X_i} \in \mathbb{R}^3$ by its 3D coordinates.

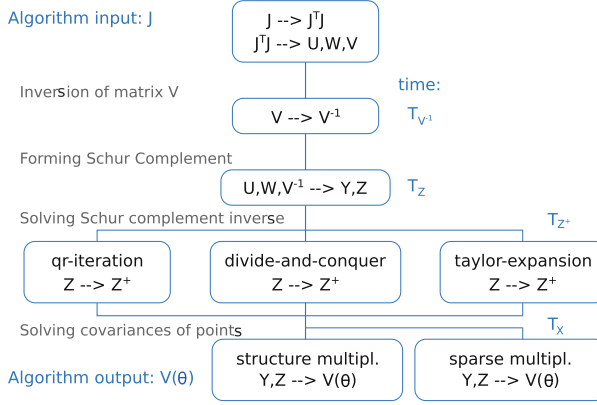


Fig. 2. The diagram of algorithm implementation

Solving Schur Complement Inverse. The first two methods qr-iteration, divide-and-conquer are SVD algorithms described in [7]. Both algorithms decompose Schur complement into $Z = \bar{U}\bar{S}\bar{V}^T$ and invert the diagonal values $\bar{S}'_{i,i} = 1/\bar{S}_{i,i}$ for which holds $\bar{S}'_{i,i} > \gamma$ where $i \in 1, 2, \dots, N_{Schur}$. The remaining values on the diagonal $\bar{S}_{i,i} \leq \gamma$ are set to zero. Note that no exact estimate of γ exists. The parameter γ , i.e. the number of inverted values, is crucial for the output values. Figure 3 visualize the influence of γ (SVD algorithms) and $\hat{\lambda}$ (Taylor expansion algorithm) on uncertainty ellipsoids where the ellipsoid show the position of a 3D point or a 3D camera center. Taylor expansion algorithm compute inverse $(cZZ + \hat{\lambda}I)^{-1}$ using Cholesky factorization. The parameter $\hat{\lambda} = trace(ZZ)c/10^{16}$ appears as reasonable trade-off between regularization cZZ and distance from zero which we try to approximate. The algorithm performs several iterations. The i -th iteration computes the i -th term T_i of Taylor series. We compute next T_i until $l_\infty(T_i) > 10^{-5}$. The usual number of iterations is [2,5].

Table 2. Additional processing times, especially the time $T_{V^{-1}}$ required for inversion of the matrix V , the time T_Z required for forming the Schur complement matrix, values of parameter $\hat{\lambda}$, the time T_{X_sparse} required for *sparse* multiplication and the time $T_{X_structure}$ for dense multiplication with using the *structure* of matrix Y

Dataset	$\hat{\lambda}$	$T_{V^{-1}}$	T_Z	T_{X_sparse}	$T_{X_structure}$	$T_{X_speedup}$
Flat	$5.20 \cdot 10^{-9}$	0.001 s	4.12 s	0.90 s	0.10 s	8.62
Tower of London	$2.28 \cdot 10^{-9}$	0.09 s	5.02 s	244.68 s	3.57 s	68.54
Trench	$3.95 \cdot 10^{-9}$	0.62 s	3.31 s	720.28 s	2.43 s	297.64
Notre Dame	$2.42 \cdot 10^{-8}$	0.17 s	7.51 s	1131.6 s	9.29 s	121.81
Seychelles	$5.08 \cdot 10^{-4}$	0.52 s	15.32 s	2982.94 s	10.98 s	271.69

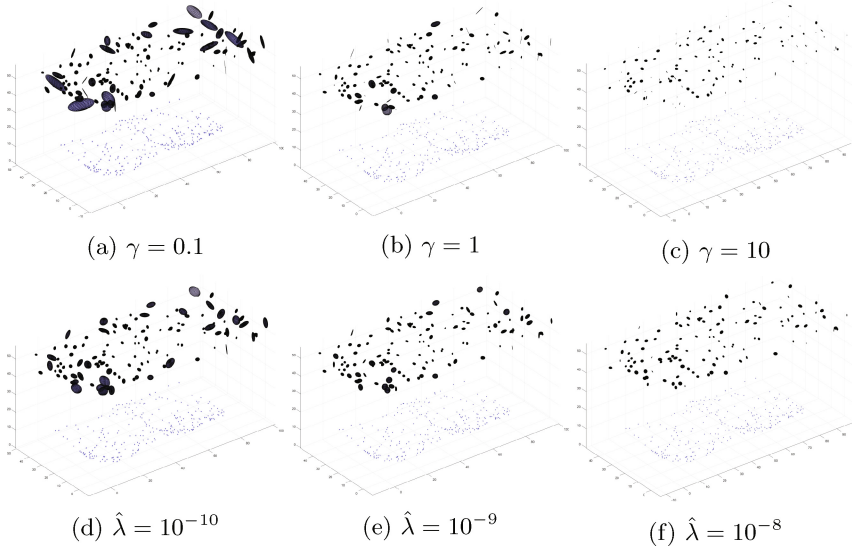


Fig. 3. Parameter sensitivity of the SVD based algorithms (parameter γ) and Taylor expansion algorithm (parameter $\hat{\lambda}$) for *flat* dataset

Assuming that all algorithms store Z and Z^+ in the same format, the difference between their memory usages depend on how they work with matrix Z . Taylor expansion algorithm requires one additional symmetric matrix of size $\mathbb{R}^{p^n \times p^n}$. Both remaining algorithms require matrices $\bar{U}, \bar{V} \in \mathbb{R}^{p^n \times p^n}$ and vector $\text{diag}(\bar{S}) \in \mathbb{R}^{p^n}$. Further qr-iteration require additional $(pn)^2 + 67(pn)$ and divide-and-conquer additional $4(pn)^2 + 7(pn)$ values for SVD.

Precision of the MP Inverse Approximation. We computed the ground truth using SVD with 100 significant digits and compare it with Taylor expansion and SVD algorithm computed in double (≈ 15 significant digits). The mean difference of camera parameters (angle axis and camera position) between the ground truth covariance matrix and the covariance matrix computed by Taylor expansion approach for $\lambda = 10^{-9}$ was 3.910^{-3} while the difference for SVD for $\gamma = 10^{-7}$ was 1.410^{-4} for Cube dataset. We are not able to test the precision of our approximations for larger dataset because of computational complexity. The covariance matrices computed by Taylor expansion approach may be useful because we get large ellipsoids for the most unconstrained cameras and points and vice versa, Fig. 3. So, we are able to determine which cameras are more and which less uncertain.

All algorithms were implemented as part of a single C++ code base. They share inversion of matrix V and forming Schur complement matrix. We use Intel MKL 11.3.1 [9] implementation of Blas and Lapack for dense linear algebra, Eigen 3.3 [14] for sparse linear algebra and matrix representation. Blas and

Table 3. Memory and time requirements. The columns M_{Z+} and T_{Z+} represent memory and time requirements for computing PM inverse of Schur complement. $\sum T$ expresses the complete run-time of the algorithm. The parameter γ was equal to 10^{-10} in all evaluations.

Flat	M_{Z+}	T_{Z+}	$\sum T$
qr-iteration	424 MB	12.57 s	13.14 s
divide-and-conquer	676 MB	10.61 s	11.18 s
taylor-expansion	253 MB	3.82 s	4.40 s
Tower of London	M_{Z+}	T_{Z+}	$\sum T$
qr-iteration	913 MB	46.79 s	52.13 s
divide-and-conquer	1 456 MB	31.02 s	36.36 s
taylor-expansion	546 MB	8.05 s	13.39 s
Trench	M_{Z+}	T_{Z+}	$\sum T$
qr-iteration	1 582 MB	90.91 s	102.04 s
divide-and-conquer	2 526 MB	69.15 s	80.28 s
taylor-expansion	947 MB	19.7 s	30.83 s
Notre Dame	M_{Z+}	T_{Z+}	$\sum T$
qr-iteration	1 660 MB	76.20 s	96.72 s
divide-and-conquer	2 651 MB	65.77 s	86.29 s
taylor-expansion	994 MB	21.24 s	41.76 s
Seychelles	M_{Z+}	T_{Z+}	$\sum T$
qr-iteration	6 357 MB	464.89 s	502.07 s
divide-and-conquer	10 161 MB	379.13 s	416.81 s
taylor-expansion	3 810 MB	139.59 s	176.47 s

Lapack functions are called from MAGMA 2.2.0 [31]. All experiments were performed on a single computer with one 2.6 GHz Intel Core i7-6700HQ with 32 GB RAM running a 64-bit Windows 10 operating system.

7 Conclusion

The state of the art method for finding covariances is based on SVD [20] which is computationally challenging for a large number of cameras and points. We propose a new method which works for scenes with 10^3 cameras and 10^6 points in reasonable time 60s on a single computer. The algorithm output changes smoothly with input parameter which is beneficial for large reconstructions. We scale Schur complement to sufficiently reduce numerical errors when using double precision. We hope that more sophisticated pre-conditioners used in Bundle Adjustment solvers [3, 34] can scale values to the range where single precision will be sufficient. That representation will allow GPU usage and order of magnitude speedup.

Acknowledgment. This work has been supported by the EU-H2020 project LADIO (number 731970) and Grant Agency of the CTU Prague project SGS16/230/OHK3/3T/13.

References

1. Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S.M., Szeliski, R.: Building rome in a day. *Commun. ACM* **54**(10), 105–112 (2011)
2. Agarwal, S., Mierle, K., et al.: Ceres solver. <http://ceres-solver.org>
3. Agarwal, S., Snavely, N., Seitz, S.M., Szeliski, R.: Bundle adjustment in the large. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) *ECCV 2010*. LNCS, vol. 6312, pp. 29–42. Springer, Heidelberg (2010). doi:[10.1007/978-3-642-15552-9_3](https://doi.org/10.1007/978-3-642-15552-9_3)
4. Balasubramanian, R., Das, S., Swaminathan, K.: Error analysis in reconstruction of a line in 3-D from two arbitrary perspective views. *Int. J. Comput. Math.* **78**(2), 191–212 (2001)
5. Belhaoua, A., Kohler, S., Hirsch, E.: Estimation of 3D reconstruction errors in a stereo-vision system. In: *SPIE Europe Optical Metrology*, p. 73900X. International Society for Optics and Photonics (2009)
6. Belhaoua, A., Kohler, S., Hirsch, E.: Error evaluation in a stereovision-based 3D reconstruction system. *EURASIP J. Image Video Process.* **2010**(1), 1 (2010)
7. Ake, B.: *Numerical Methods for Least Squares Problems*. SIAM, Philadelphia (1996)
8. Blostein, S.D., Huang, T.S.: Error analysis in stereo determination of 3-D point positions. *IEEE Trans. Pattern Anal. Mach. Intell. PAMI* **9**(6), 752–765 (1987)
9. Intel Corporation. Intel math kernel library. <https://software.intel.com/en-us/intel-mkl>
10. Dong, T., Haidar, A., Tomov, S., Dongarra, J.: A fast batched Cholesky factorization on a GPU. In: *2014 43rd International Conference on Parallel Processing*, pp. 432–440. IEEE (2014)
11. Faugeras, O., Luong, Q.-T., Papadopoulou, T.: *The Geometry of Multiple Images: The Laws That Govern the Formation of Images of a Scene and Some of Their Applications*. MIT Press, Cambridge (2001)
12. Förstner, W.: Uncertainty and projective geometry. *Handbook of Geometric Computing*, pp. 493–534. Springer, Heidelberg (2005)
13. Frahm, J.-M., et al.: Building Rome on a cloudless day. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) *ECCV 2010*. LNCS, vol. 6314, pp. 368–381. Springer, Heidelberg (2010). doi:[10.1007/978-3-642-15561-1_27](https://doi.org/10.1007/978-3-642-15561-1_27)
14. Guennebaud, G., Jacob, B., et al.: Eigen v3.3 (2010). <http://eigen.tuxfamily.org>
15. Hartley, R., Zisserman, A.: *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York (2003)
16. 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)
17. Höhle, J., Höhle, M.: Accuracy assessment of digital elevation models by means of robust statistical methods. *ISPRS J. Photogrammetry Remote Sens.* **64**(4), 398–406 (2009)
18. Kamberova, G., Bajcsy, R.: Precision in 3-D points reconstructed from stereo. Technical reports (CIS), p. 201 (1997)

19. Langguth, F., Sunkavalli, K., Hadap, S., Goesele, M.: Shading-aware multi-view stereo. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 469–485. Springer, Cham (2016). doi:[10.1007/978-3-319-46487-9_29](https://doi.org/10.1007/978-3-319-46487-9_29)
20. Lhuillier, M., Perriollat, M.: Uncertainty ellipsoids calculations for complex 3D reconstructions. In: Proceedings 2006 IEEE International Conference on Robotics and Automation, ICRA 2006, pp. 3062–3069. IEEE (2006)
21. Min, S., Rixin, H., Daojun, W.: Precision analysis to 3D reconstruction from image sequences. In: The 5th ISPRS Workshop on DMGISs. CiteSeer (2007)
22. Park, S.-Y., Subbarao, M.: A multiview 3D modeling system based on stereo vision techniques. *Mach. Vis. Appl.* **16**(3), 148–156 (2005)
23. Pauly, M., Mitra, N.J., Guibas, L.J.: Uncertainty and variability in point cloud surface data. In: Symposium on point-based graphics, vol. 9 (2004)
24. Polok, L., Ila, V., Smrz, P.: 3D reconstruction quality analysis and its acceleration on GPU clusters (2016)
25. Rivera-Rios, A.H., Shih, F.-L., Marefat, M.: Stereo camera pose determination with error reduction and tolerance satisfaction for dimensional measurements. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation, pp. 423–428. IEEE (2005)
26. Schaer, P., Skaloud, J., Landtwing, S., Legat, K.: Accuracy estimation for laser point cloud including scanning geometry. In: Mobile Mapping Symposium 2007, Padova, number TOPO-CONF-2008-015 (2007)
27. Schönberger, J.L., Zheng, E., Frahm, J.-M., Pollefeys, M.: Pixelwise view selection for unstructured multi-view stereo. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 501–518. Springer, Cham (2016). doi:[10.1007/978-3-319-46487-9_31](https://doi.org/10.1007/978-3-319-46487-9_31)
28. Schönberger, J.L., Frahm, J.-M.: Structure-from-motion revisited. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
29. 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)
30. Snavely, N., Seitz, S.M., Szeliski, R.: Skeletal graphs for efficient structure from motion. In: CVPR, vol. 1, p. 2 (2008)
31. Tomov, S., Dongarra, J., Baboulin, M.: Towards dense linear algebra for hybrid GPU accelerated manycore systems. *Parallel Comput.* **36**(5–6), 232–240 (2010)
32. Wilson, K., Snavely, N.: Robust global translations with 1DSfM. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8691, pp. 61–75. Springer, Cham (2014). doi:[10.1007/978-3-319-10578-9_5](https://doi.org/10.1007/978-3-319-10578-9_5)
33. Wu, C.: Towards linear-time incremental structure from motion. In: 2013 International Conference on 3D Vision-3DV 2013, pp. 127–134. IEEE (2013)
34. C. Wu, Agarwal, S., Curless, B., Seitz, S.M.: Multicore bundle adjustment. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3057–3064. IEEE (2011)
35. Zhang, F.: *The Schur Complement and Its Applications*. Springer, New York (2005)