Efficient Multiple Loop Adjustment for Computer Vision Tasks

JOCHEN MEIDOW, Ettlingen

Keywords: loop adjustment, image mosaicking, simultaneous localization and mapping

Summary: Applications performing dead reckoning generally suffer from the inevitable drift of the solutions due to measurement noise and other systematic errors. When dealing with moving sensors such as cameras or laser scanners, loop closing represents one of the most important strategies to compensate for these effects. Imposing loop constraints reduces the uncertainties that arise and permits the construction of consistent maps, mosaics, or scene reconstructions. We propose and demonstrate an approach to correct multiple loops simultaneously by least squares adjustments in 2D and 3D. The results are statistically optimal, since all individual uncertainties and correlations are considered. The use of minimal transformation representations without singularities enables efficient implementation. Furthermore, we address the selection of consistent loop constraints by exploiting the natural order of the sensor datasets given by the acquisition process. The feasibility and the efficiency of the approach are demonstrated using synthetic data reflecting an indoor exploration scenario and by examples for terrestrial and airborne video mosaicking.

Zusammenfassung: Die Lösungen von Anwendungen, die eine Koppelnavigation durchführen, weisen in der Regel eine unvermeidbare Drift aufgrund von Messrauschen und restsystematischen Fehlern auf. Konzeptionell stellen Schleifenschlüsse eine wichtige Strategie dar, um diese Effekte zu kompensieren. Die Schleifenbedingungen reduzieren die auftretenden Ungenauigkeiten und gestatten den Aufbau von konsistenten Karten, Mosaiken oder Szenenrekonstruktionen. Wir stellen einen allgemeinen Ansatz vor, um mehrere auftretende Schleifen gleichzeitig in 2D und 3D durch Ausgleichungen zur korrigieren. Die Ergebnisse sind statistisch optimal, da alle individuellen Unsicherheiten und Korrelationen berücksichtigt werden. Die verwendeten minimalen Repräsentationen für Transformationen in 2D bzw. 3D weisen keine Singularitäten auf und ermöglichen eine effiziente Umsetzung. Des Weiteren wird eine Strategie für die Auswahl konsistenter Bedingungsgleichungen vorgestellt, welche sich aufgrund der zeitlichen Reihenfolge der Daten im Erfassungsprozess ergibt. Die Durchführbarkeit und die Effizienz des Ansatzes wird anhand synthetischer Daten demonstriert, die eine Innenraumexploration simulieren, sowie mit Beispielen für die Mosaikierungen von terrestrisch und luftgetragen erfassten Videobildern.

1 Introduction

1.1 *Motivation*

Today, navigation, image mosaicking, and the exploration of unknown environments by simultaneous localization and mapping (SLAM) are common tasks in robotics, computer vision and photogrammetry. Exteroceptive sensors, such as cameras or laser scanners, measure the proximity of objects to the sensor's frame of reference. The use of moving sensors permits the acquisition of environments for which one builds or updates maps. Furthermore, the sensors enable the determination of ego-motion in 3D by establishing correspondences between consecutively captured datasets. The latter is of special interest when external references such as landmarks with known coordinates or reference signals such as GPS-signals are not available. In this situation, dead reckoning has to be applied, which is subject to inevitable drift due to measurement noise and

remaining systematic errors. The same holds for the mosaicking of video streams: Assuming planar scenes, the chaining of consecutive image pairs suffers from accumulation of random feature-tracking errors and systematic errors due to imperfect calibration. This becomes evident when loops are present: drift appears in discrepancies or gaps at the joints.

Conceptually, loop closing is one of the most important strategies to compensate for drift and to obtain more precise, globally consistent results. Whenever a system recognizes places already visited, the discrepancies that occur should be distributed over the covered path and the scene reconstruction.

1.2 *Related Work*

Contributions to loop adjustment are manifold and can be found in the robotics, photogrammetry, and computer vision literature. The formulation and solution of optimization or interpolation tasks, the use of appropriate parameterizations and the construction of consistent sets of constraints are all topics in which loop adjustment has been considered.

For single loops, the parameter corrections sought can be determined by distributing the updates proportionally over the loop trajectory. In 3D, this can be accomplished by considering the so-called minimal-length trajectory between two poses, which is given by a straight line in the corresponding tangent space. This interpolation can be controlled using scalar weights reasonably chosen proportional to the uncertainties at hand and leads to a minimal bending of the trajectory (DUBBELMAN et al. 2010). In the context of video mosaicking, an optimization procedure is proposed by CABALLERO et al. (2007) for instance. Applying an extendend Kalman filter, an optimization procedure is employed for updating the loop homographies. Within each step of the filter update a normalization of the homography matrices is necessary to fix the scale of the homogeneous representation.

A solution to reduce the errors in a 3D network of observed transformations is provided in GRISETTI et al. (2007). The optimization is performed using a variant of gradient descent. In doing so, the updates are consid-

ered separately for the rotational component (spherical linear interpolation) and the translational component. Results for experiments with thousands of involved transformations are shown. When deriving constraints from a network of observations, the resulting set of equations may not necessarily be consistent. When the network is represented as a graph, this leads to a need for so-called fundamental cycle bases (UNNIKRISHNAN & KELLY 2002b).

The work closest to ours is that of ESTRADA et al. (2005), which covers loop detection and adjustment for the 2D SLAM case. The squared Mahalanobis distance is calculated for hypothesis testing and nonlinear constrained least-squares optimization by sequential quadratic programming is used to impose multiple loop constraints simultaneously. However, this work does not address the need for finding a consistent set of independent constraints.

1.3 *Contribution*

Essentially, we extend the approach of ESTRADA et al. (2005) to the 3D case. We propose an efficient method to build large and globally consistent mosaics and sensor trajectories within the same framework. As proposed by MEIDOW (2011) in the context of video mosaicking, this is achieved by exploiting the power of algebraic projective geometry, by using minimal representations without singularities (STRASDAT et al. 2010), and by compiling consistent sets of loop constraints (UNNIKRISHNAN & KELLY 2002b). For the latter we exploit the natural order of the datasets (video images or laser scans) captured by a moving sensor. The approach allows the simultaneous adjustment of multiple loops in a batch process after solving the place recognition task. By considering image-to-image transformations or pose changes as observations, we chose an adjustment model with constraints for these transformations parameters only, which leads to small equation systems to be solved. The stochastic model involved in these tasks rigorously incorporates the uncertainties of the transformation parameters.

For the image alignment task, this approach assumes planar scenes or a fixed projection

centre. We assume uncalibrated cameras with, however, straight-line preserving optics, i.e., we assume the lens distortion to be negligible.

2 Theoretical Background

2.1 *Notation and Preliminaries*

Homogeneous entities are denoted by upright boldface letters, e.g. $\mathbf{x}$ or $\mathbf{H}$, Euclidean vectors and matrices with italic boldface letters, e.g. $\boldsymbol{l}$ or $\boldsymbol{R}$. For homogeneous coordinates "=" means an assignment or an equivalence up to a common scale factor $\lambda \neq 0$.

For the minimal parameterizations of a 2D homography or 3D motion, we exploit the power series

$$\exp(\boldsymbol{K}) = \sum_{k=0}^{\infty} \frac{1}{k!} \boldsymbol{K}^k = I + \boldsymbol{K} + \frac{1}{2} \boldsymbol{K}^2 + \dots, \quad (1)$$

which is the power series for square matrices analogous to the power series for the scalar exponential function. For the analytical computation of Jacobians, we will frequently use the rule

$$\mathrm{vec}(\boldsymbol{ABC}) = \left(\boldsymbol{C}^\mathsf{T} \otimes \boldsymbol{A} \right) \mathrm{vec}(\boldsymbol{B}) \quad (2)$$

and its specializations. Here, the vec operator stacks all columns of a matrix and $\otimes$ denotes the Kronecker product. The skew-symmetric matrix $\boldsymbol{S}(\boldsymbol{a})$ built from a 3-vector $\boldsymbol{a}$ induces a cross-product.

2.2 *Parameterizations*

We use homogeneous coordinates to represent homographies in 2D and motion in 3D. In doing so, 3D motion can be considered as a special homography in 3D, which paves the way to a common framework to treat 2D and 3D tasks in the same way. Homographies form a group. Thus, one can "undo" a transformation by computing and applying the inverse transformation (matrix inversion). The concatenation or chaining of two or more transformations is carried out by matrix multiplication.

A planar projective transformation is a linear transformation on homogeneous 3-vectors represented by a non-singular 3×3 matrix $\mathbf{H} = (H_{ij})$, cf. (Hartley & Zisserman 2004):

$$\begin{pmatrix} u' \\ v' \\ w' \end{pmatrix} = \begin{pmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{pmatrix} \begin{pmatrix} u \\ v \\ w \end{pmatrix}, \quad (3)$$

or, more briefly, $\mathbf{x}' = \mathbf{H}\,\mathbf{x}$. This transformation is unique up to scale and has therefore eight degrees of freedom. It can be written in inhomogeneous form as

$$x' = \frac{H_{11}x + H_{12}y + H_{13}}{H_{31}x + H_{32}y + H_{33}} \quad \text{and}$$

$$y' = \frac{H_{21}x + H_{22}y + H_{23}}{H_{31}x + H_{32}y + H_{33}} \quad (4)$$

with $\mathbf{x} = [x, y, 1]^\mathsf{T}$ and $\mathbf{x}' = [x', y', 1]^\mathsf{T}$.

A rigid-body transform in $\mathbb{R}^3$ can be expressed by a 4×4 transformation matrix applied to homogeneous 4-vectors:

$$\mathbf{H} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix} \text{ with } \boldsymbol{R} \in SO(3) \text{ and } \boldsymbol{t} \in \mathbb{R}^3 , \quad (5)$$

where $SO(3)$ is the Lie group of rotation matrices. The motion matrices (5) form a smooth manifold and, therefore, the Special Euclidean group $SE(3)$. Its operator is matrix multiplication.

In homogeneous coordinates, all transformations are realized by multiplication. Thus, linearization in a multiplicative manner is straightforward (Förstner 2010). With a multiplicative expansion, a nonlinear update reads

$$\mathbf{H} = \Delta\mathbf{H} \cdot \mathbf{H}_0 \quad (6)$$

with an approximate homography $\mathbf{H}_0$ and the update $\Delta\mathbf{H}$ within the iteration sequence.

The power series for the matrix exponential (1) relates Euclidean updates to representations in the tangent space. For 2D homographies, this is the zero-trace matrix

$$\boldsymbol{K}(\boldsymbol{q}_{2\mathrm{D}}) = \begin{pmatrix} k_1 & k_4 & k_7 \\ k_2 & k_5 & k_8 \\ k_3 & k_6 & -k_1 - k_5 \end{pmatrix}, \quad (7)$$

which depends linearly on the eight correction parameters $\boldsymbol{q}_{2D} = [k_1, \ldots, k_8]^T$ (BEGELFOR & WERMAN 2005). Requiring all matrices to have determinant one, thus constituting a Lie group $SL(3)$, leads to an easy linearization scheme (FÖRSTNER 2012). For 3D motion, the relation is given by the so-called twist representation

$$\boldsymbol{K}(\boldsymbol{q}_{3D}) = \begin{bmatrix} \boldsymbol{S}(\boldsymbol{r}) & \Delta\boldsymbol{t} \\ \boldsymbol{0}^T & 0 \end{bmatrix} \qquad (8)$$

with the six motion parameters $\boldsymbol{q}_{3D} = [\boldsymbol{r}^T, \Delta\boldsymbol{t}^T]^T$ comprising three rotation parameters $\boldsymbol{r}$ and a translational update $\Delta\boldsymbol{t}$ (BREGLER & MALIK 1998).

During the iterative estimation we do not update the homography parameters or the motion parameters in tangent space, but rather the approximate transformations $\mathbf{H}_0$ by mapping the updates onto the manifold $SL(3)$ or $SE(3)$, respectively

$$\mathbf{H}_0^{(\nu+1)} = \exp\big(\boldsymbol{K}(\boldsymbol{q})\big) \cdot \mathbf{H}_0^{(\nu)}, \qquad (9)$$

where ν denotes the iteration step. Thus, we end up with the estimates $\hat{\mathbf{H}} = \mathbf{H}_0$, $\hat{\boldsymbol{q}} = \boldsymbol{0}$, and $\boldsymbol{\Sigma}_{\hat{q}\hat{q}}$ for the covariance matrix of the estimated transformation parameters $\boldsymbol{q}_{2D}$ and $\boldsymbol{q}_{3D}$, respectively.

2.3 *Sequential Links, Cross Links and Cumulative Transformations*

For establishing loops, we consider two different types of image alignment and motion: sequential links and cross links (TURKBEYLER & HARRIS 2010). *Sequential links* are given by the chaining of consecutive, i.e., temporally adjacent, sensor data. For mosaicking and visual SLAM, these transformations are usually determined by tracking or matching corresponding image features and, for laser scanning data, by the registration of point clouds. For the efficient computation of Jacobians within the adjustment process, we will frequently use *cumulative transforms* of sequential links

$$^k\mathbf{H}_1 = \prod_{l=1}^{k-1} {}^{l+1}\mathbf{H}_l = {}^k\mathbf{H}_{k-1} \cdot \ldots \cdot {}^3\mathbf{H}_2 \cdot {}^2\mathbf{H}_1,$$
$$k = 2, \ldots, n \qquad (10)$$

in analogy to cumulative sums and cumulative products. They start with the very first homography $^2\mathbf{H}_1$ of each sequence and result from the kinematic chains. The concatenation of motions is represented by multiplication from the left.

Cross links can be established whenever a system recognizes an already visited place or by visual inspection. Here, the search for correspondences is much more challenging and sophisticated methods such as SIFT (scale invariant feature transform) for images (LOWE 2004) or ICP (iterative closest point) for point clouds (BESL & McKAY 1992) are needed to cope with changed sensor aspects.

3 Multiple Loop Adjustment

After the detection of loop closure events by place recognition and the determination of the corresponding cross link parameters, the mosaic loops or motion involved in these events need to be adjusted. Taking all uncertainties of the transformation parameters into account, we pursue a statistically rigorous and optimal approach (section 3.2). The approach is suboptimal compared to the joint estimation of all transformations together with the common 2D and 3D points, e.g. SZELISKI (2006), TRIGGS et al. (2000), but enables the efficient treatment of large datasets. For multiple loop adjustment, the selection of a proper and consistent set of constraints is crucial (section 3.1).

3.1 *Choice of Cycle Basis*

The set of constraints used within the adjustment must not be redundant, i.e., the equations must be independent. This can be realized by considering a topological graph as a description of the links given by chained motions or image transforms. Fig. 1 shows an example for a sensor path with three established cross links. The vertices of the graph denote locally captured data elements, i.e., images or point clouds, and the edges denote transformations. Additional cross links have been marked by arrows. In this graph, it is possible to establish six different cycles (loops) that do not form an independent set. A complete cover of the
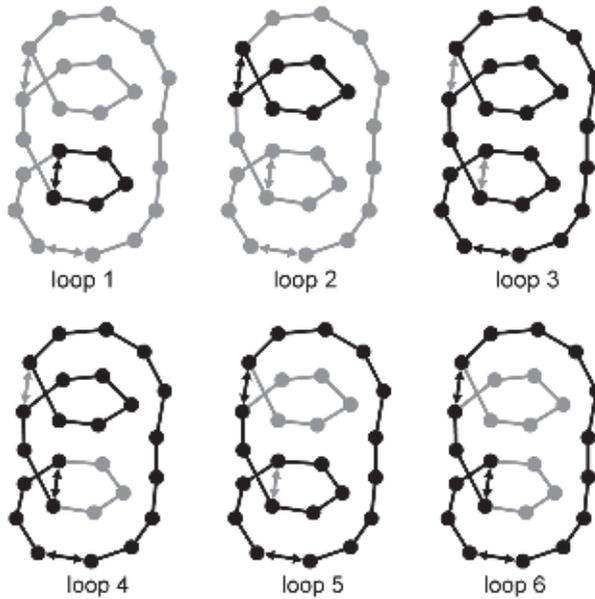
Fig. 1: Six possible loops for a given sensor path with three cross links marked by arrows. Loops 1, 2 and 6 constitute the smallest-size basis, whereas the loops 1, 2 and 3 form the proposed set of loops established along the sensor's path with one cross link each.

graph is given, for instance, by loops 1, 2 and 6. In graph theory, such an independent and complete cycle cover is called a fundamental cycle basis (UNNIKRISHNAN & KELLY 2002b).

The solution of the least squares problem proposed in section 3.2 does not depend on the choice of the cycle basis, since the linearized equations corresponding to any cycle basis form an independent set in the space of constraints provided the approximate values are the same. Choosing the cycle basis with the smallest size reduces the computational costs since the Jacobians involved are sparser. Here "smallest size" refers to the number of edges to be traversed in total (shortest loop lengths). However, the problem of finding a cycle basis of smallest size has been proven to be NP-hard for an arbitrary graph (THOMASSEN 1997). Therefore and for the sake of simplicity, we establish loops along the sensor path immediately whenever they occur, i.e., each loop contains exactly one cross link. This approach is simple and easy and exploits the natural temporal and spatial order of the data provided by the acquisition process. In Fig. 1 the loops 1, 2 and 6 constitute the smallest-size cycle basis, whereas the loops 1, 2 and 3 are loops established immediately whenever the sensor revisits a place.

3.2 Imposing Loop Constraints

The proposed loop adjustments are performed by solving a standard least squares problem. A solution to this problem is obtained by an adaption of sequential quadratic programming (TRIGGS et al. 2000). Considering the transformation parameters to be given but uncertain observations, the solution is equivalent to the adjustment with constraints for observations only (KOCH 1999, MCGLONE et al. 2004). For the sake of completeness, we briefly summarize here the adjustment model used and its corresponding procedure. Then the loop constraints and the corresponding Jacobians are presented.

With the given block-diagonal covariance matrix Σ_{pp} of the unconstrained (observed) parameters $[^2\boldsymbol{q}_1^\mathsf{T}, \, ^3\boldsymbol{q}_2^\mathsf{T}, \, \dots \, ^n\boldsymbol{q}_{n-1}^\mathsf{T}]^\mathsf{T}$ (set of sequential

links) augmented by the observed cross link parameters, the optimization problem is given by

$$\Omega = \left(\boldsymbol{p} - \hat{\boldsymbol{p}} \right)^{\mathsf{T}} \boldsymbol{\Sigma}_{pp}^{-1} \left(\boldsymbol{p} - \hat{\boldsymbol{p}} \right) \ \rightarrow \ \min \qquad (11)$$

subject to $\boldsymbol{c}(\hat{\boldsymbol{p}}) = \boldsymbol{0}$. Here, the sought constrained parameters are $\hat{\boldsymbol{p}}$ and the loop constraints are $\boldsymbol{c}(\hat{\boldsymbol{p}}) = \boldsymbol{0}$. Linearization of the constraints yields

$$\boldsymbol{c}(\hat{\boldsymbol{p}}) = \boldsymbol{c}(\boldsymbol{p}_0) + \boldsymbol{C} \boldsymbol{\Delta} \hat{\boldsymbol{p}} \qquad (12)$$

with the Jacobian $\boldsymbol{C}$. Then the estimator is

$$\hat{\boldsymbol{p}}^{(v+1)} = \boldsymbol{p} + \boldsymbol{\Sigma}_{pp} \boldsymbol{C}^{\mathsf{T}} \left(\boldsymbol{C} \boldsymbol{\Sigma}_{pp} \boldsymbol{C}^{\mathsf{T}} \right)^{-1} \boldsymbol{c}_0 \qquad (13)$$

with $\boldsymbol{C} \boldsymbol{\Sigma}_{pp} \boldsymbol{C}^{\mathsf{T}}$ being the covariance matrix of the contradictions $\boldsymbol{c}(\boldsymbol{p})$ and

$$\boldsymbol{c}_0 = \boldsymbol{C} \left(\hat{\boldsymbol{p}}^{(v)} - \boldsymbol{p} \right) - \boldsymbol{c} \left(\hat{\boldsymbol{p}}^{(v)} \right) \qquad (14)$$

at each iteration v. The covariance matrix for the estimates (13)

$$\boldsymbol{\Sigma}_{\hat{p}\hat{p}} = \boldsymbol{\Sigma}_{pp} - \boldsymbol{\Sigma}_{pp} \boldsymbol{C}^{\mathsf{T}} \left(\boldsymbol{C} \boldsymbol{\Sigma}_{pp} \boldsymbol{C}^{\mathsf{T}} \right)^{-1} \boldsymbol{C} \boldsymbol{\Sigma}_{pp} \qquad (15)$$

is fully occupied in general and therefore computable for problems of moderate size only.

The constraints for the chained homographies or integrated motion are simply

$$^{j}\mathbf{H}'_{i} - \prod_{l=i}^{j-1} {}^{l+1}\mathbf{H}_{l} = \mathbf{O} \qquad (16)$$

for a single loop with $j > i$. The constraints take one cross link (see section 3.1) into account, denoted by a single prime. Please note that all homography matrices are unambiguous due the determinant one constraint or the Euclidean motion parameterization. An alternative formulation of the loop constraints (16) is

$$^{i}\mathbf{H}'_{j} \cdot \prod_{l=i}^{j-1} {}^{l+1}\mathbf{H}_{l} = \boldsymbol{I} \qquad (17)$$

(UNNIKRISHNAN & KELLY 2002a), whereby the Jacobians w.r.t. the cross link depend on the sequential links, too.

The constraints have to be fulfilled for the adjusted parameters. During the optimization

process we update the approximate homographies according to (9). Thus, the approximate values for the parameters $\boldsymbol{p}$ are simply zeros and we can expand the constraints with the corresponding multiplicative updates $\Delta \mathbf{H} = \exp(\boldsymbol{K}(\boldsymbol{q})) = \boldsymbol{I}$. For the vectorization of the constraints (16), we consider this for the cross link (i, j) and a sequential link $(k, k+1)$

$$^{j}\Delta\mathbf{H}'_{i} \cdot {}^{j}\mathbf{H}'_{i} - {}^{j}\mathbf{H}_{k+1} \left({}^{k+1}\Delta\mathbf{H}_{k} \cdot {}^{k+1}\mathbf{H}_{k} \right) {}^{k}\mathbf{H}_{i} = \mathbf{O} \qquad (18)$$

With $\Delta \mathbf{h} = \mathrm{vec}(\Delta \mathbf{H})$ and the rule (2), the vectorization of (18) yields nine equations for 2D homographies and 16 equations for 3D motion respectively:

$$\boldsymbol{c} = \left({}^{j}\mathbf{H}'^{\mathsf{T}}_{i} \otimes \boldsymbol{I} \right) {}^{j}\Delta\mathbf{h}'_{i} - \left({}^{k+1}\mathbf{H}^{\mathsf{T}}_{i} \otimes {}^{j}\mathbf{H}_{k+1} \right) \cdot {}^{k+1}\Delta\mathbf{h}_{k}$$
$$= \mathbf{0} \qquad (19)$$

The parameters $\boldsymbol{q}$ of a single transformation are related to the corresponding homography $\mathbf{H}$ by $\Delta \mathbf{H} = \exp(\boldsymbol{K}(\boldsymbol{q})) \approx \boldsymbol{I} + \boldsymbol{K}$ for small values $\boldsymbol{q}$. Thus, the linearization is

$$\Delta\mathbf{h} \approx \mathrm{vec}(\boldsymbol{I}) + \mathrm{vec}\left(\boldsymbol{K}(\boldsymbol{q}) \right) = \mathrm{vec}(\boldsymbol{I}) + \boldsymbol{G}\boldsymbol{q} \qquad (20)$$

with the constant transformation matrix $\boldsymbol{G}$. For 2D homographies, $\boldsymbol{G}$ is simply the 9×8 matrix

$$\boldsymbol{G}_{2\mathrm{D}} = \begin{bmatrix} \boldsymbol{I}_8 \\ -1,0,0,0,-1,0,0,0 \end{bmatrix} \qquad (21)$$

(MEIDOW 2011). For 3D pose changes, we apply a 16×6 transformation matrix $\boldsymbol{G}_{3\mathrm{D}}$ composed of ones, zeros and minus ones.

The constraints (19) are linearly dependent and a reduction to a set of six and eight constraints, respectively, is necessary. This can be achieved by deleting equations or by computing a linear combination. Applying the pseudo inverses of the matrices $\boldsymbol{G}$, yields reduced constraints $\bar{\boldsymbol{c}} = \boldsymbol{G}^{+}\boldsymbol{c}$.

The Jacobians are then

$$\boldsymbol{C}^{(c)} = \frac{\partial \bar{\boldsymbol{c}}(\hat{\boldsymbol{q}}')}{\partial \hat{\boldsymbol{q}}'} = \frac{\partial \bar{\boldsymbol{c}}(\hat{\boldsymbol{q}}')}{\partial \Delta\mathbf{h}'} \frac{\partial \Delta\mathbf{h}'}{\partial \hat{\boldsymbol{q}}'} = \boldsymbol{G}^{+} \left({}^{j}\mathbf{H}'^{\mathsf{T}}_{i} \otimes \boldsymbol{I} \right) \boldsymbol{G} \qquad (22)$$

w.r.t. the parameters of the cross link and

$$C^{(s)} = \frac{\partial \overline{c}(\hat{q})}{\partial \hat{q}} = G^+ \left({}^{k+1}\mathbf{H}_i^\mathsf{T} \otimes {}^j\mathbf{H}_{k+1} \right) G$$

$$= G^+ \left[\left({}^i\mathbf{H}_1^{-\mathsf{T}} \cdot {}^{k+1}\mathbf{H}_1^\mathsf{T} \right) \otimes \left({}^j\mathbf{H}_1 \cdot {}^{k+1}\mathbf{H}_1^{-1} \right) \right] G \tag{23}$$

w.r.t. the parameters of a sequential link. Note that all Jacobians can be computed with a cross link or cumulative transforms only.

For multiple loops to be adjusted, the entire Jacobian C is then a matrix for the S sequential links

$$C_{\text{seq}} = \begin{bmatrix} C_{11}^{(s)} & C_{12}^{(s)} & \cdots & C_{1S}^{(s)} \\ C_{21}^{(s)} & C_{22}^{(s)} & \cdots & C_{2S}^{(s)} \\ \vdots & \vdots & \ddots & \vdots \\ C_{L1}^{(s)} & C_{L2}^{(s)} & \cdots & C_{LS}^{(s)} \end{bmatrix} \tag{24}$$

augmented by a usually much smaller block diagonal matrix

$$C_{\text{cross}} = \text{Diag}\left(C_1^{(c)}, C_2^{(c)}, \ldots, C_L^{(c)} \right), \tag{25}$$

where L denotes the number of cross links, i.e., loops, and $D = 8$ parameters for 2D homographies and $D = 6$ parameters for 3D motion, respectively. Thus, the size of the Jacobian

$$C = \begin{bmatrix} C_{\text{seq}} & | & C_{\text{cross}} \end{bmatrix} \tag{26}$$

is $DL \times D(S+L)$. Its shape is usually very flat, since $S \gg L$ holds. The sparseness of the Jaco-bian C_{seq} depends on the chosen cycle basis. The size of the matrix to be inverted in (13) depends only on the number of loops L. Note that no special treatment of the homographies not being part of any loop is necessary. No case-by-case analysis is necessary, since the corresponding estimates for the corrections in (13) will simply be zeros.

4 Experiments

In the following, we demonstrate the feasibility of the approach for the tasks of video mosaicking (terrestrial and airborne, section 4.1) and 3D exploration, the latter possibly by simultaneous localization and mapping (SLAM) with the help of video cameras or laser scanners (section 4.2).

4.1 *Video Mosaicking*

For each image of a video stream, salient points have been extracted by the Förstner operator (Förstner & Gülch 1987), and tracked in the corresponding subsequent image by the Lucas-Kanade tracker (Lucas & Kanade 1981). The respective positional uncertainties – represented by covariance matrices – have been determined for incorporation into the adjustment. The correspondences of the cross
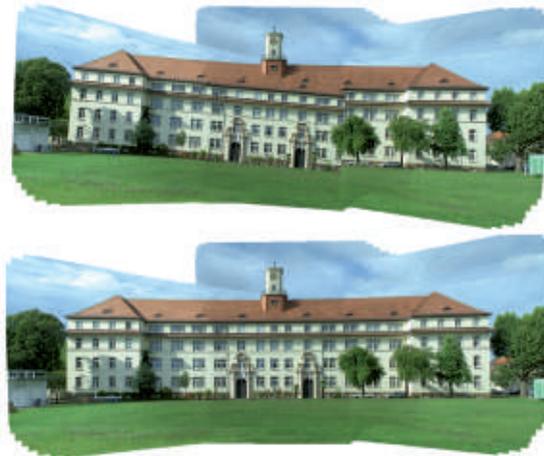
Fig. 2: Mosaicking of 355 video images of size 720 × 576 pixel. Top: Image alignment by considering consecutive links only. Bottom: Result after the adjustment of two loops.

links have then been established by applying the scale invariant feature transform SIFT (LOWE 2004) in combination with random sample consensus (FISCHLER & BOLLES 1981).

Fig. 2 shows the result for the acquisition of a façade by an uncalibrated ordinary video camera before and after adjustment of two loops. The camera was moved in a pattern ac-

cording to the ∞ sign. The mosaic obtained by applying the consecutive homographies only is shown in Fig. 2, top, and reveals numerous discrepancies due to drift. Subsequent adjustment yields a consistent result. The gauge freedoms for the mosaics have been fixed by selecting the mean cumulative homography (10) as reference transformation.
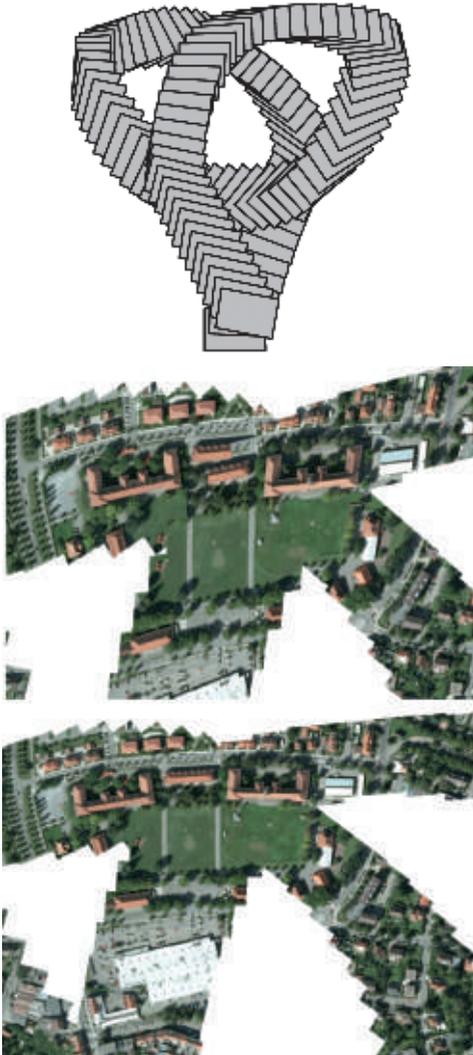
For the provision of airborne imagery we abused a virtual globe system as an image source and camera simulator. This guarantees that the model assumptions are valid – all uncertainties stem from tracking and matching, respectively. The images have a rather poor resolution of 480×320 pixel. The given camera path features varying heights above the ground as well as changing roll, pitch, and yaw angles. During the flight, the camera carried out a 180-degree-turn around its optical axis, see Fig. 3. Four loop closing events have been identified by visual inspection. The computing time for the adjustment was 1.4 seconds on a standard CPU at 1.59 GHz with non-optimized Matlab code in four iterations.

4.2 *Simulated Indoor Exploration*

To the author's knowledge, there are no publicly available 3D data that contain sensor motion for sequential links and additional cross links with uncertainty information as well as ground truth for evaluation, neither derived from visual data nor from laser scans. Therefore, we restricted our investigation to synthetic data reflecting already derived navigation solutions obtained by dead reckoning. In doing so, we are omitting the effects of specific feature tracking and/or feature matching techniques, which are outside the scope of this investigation.

Fig. 4 summarizes the experimental setup, the simulated data and the results. The chosen sensor trajectory imitates indoor exploration by a forward looking moving video camera. The camera visits two rooms, moves along corridors, and finally revisits its starting point. The results could be provided by a visual SLAM algorithm, e.g. a sliding window bundle adjustment. Gaussian noise has been added to the successive relative motion. This results in the camera path depicted in Fig. 4,

Fig. 3: Mosaicking of approx. 1,200 images with 480×320 pixel resolution. Top: Footprints of the adjusted images (every 10th drawn). Middle: Aerial mosaic built by using the estimated sequential links only (detail). Bottom: Mosaic after adjusting four loops (detail). Image data: ©GeoContent provided by Google.

centre. The gap between the starting and end points of the path is huge. The result after closing and adjusting the three loops given by the fundamental cycle basis depicted in Fig. 1 (Loops 1, 2 and 3) is clearly more consistent. The computing time for the adjustment was 0.1 seconds at 1.59 GHz with a non-optimized Matlab implementation.

Most of the absolute sensor poses change considerably during the adjustment. However,

the relative consecutive motions change only slightly due to the least squares approach. Locally, the trajectory is hardly affected while, globally, consistency is achieved. For the determination of the empirical accuracy of the adjustment result, we compute the test statistic

$$T = \frac{1}{R}\left(\hat{\boldsymbol{p}} - \boldsymbol{p}^{(\mathrm{ref})}\right)^{\mathsf{T}}\left(\boldsymbol{\Sigma}_{\hat{p}\hat{p}} + \boldsymbol{\Sigma}_{pp}^{(\mathrm{ref})}\right)^{-1}\left(\hat{\boldsymbol{p}} - \boldsymbol{p}^{(\mathrm{ref})}\right)$$
$$\sim F_{R,\infty} \tag{27}$$

based on the Mahalanobis distance w.r.t. reference values of the parameters, i.e., the ground truth (McGlone et al. 2004, Dickscheid et al. 2008). The test statistic is $F_{R,\infty}$-distributed with $R = DS$ degrees of freedom. We obtain $T = 1.0174$ which is lower than the 0.95-quantile 1.0176 of the F-distribution. Thus the accuracy potential has been fully exploited.

5   Conclusions

We have proposed a statistically optimal approach to adjust multiple loops simultaneously in particular for image alignment in 2D and simultaneous localization and mapping in 3D. The adjustment model chosen and the parameterizations used enable an efficient implementation: Hundreds of images or poses can be adjusted within a second. To come up with a set of consistent loop constraints, we exploit the natural order of datasets given by the sequential acquisition process of moving mapping sensors.

Visual inspection of the test results clearly reveals the capability of the approach to compensate for inevitable drift, which indicates that the approach supports the generation of consistent mosaics and scene reconstructions. For path planning during exploration, such loop closing implies an obvious strategy: loops should be detected and closed as soon as possible and wherever possible.

Fig. 4: Synthetic data with 3,000 camera poses (every 10th is plotted) simulating indoor exploration by a moving camera with subsequent place recognition and loop adjustment. Top: Ground truth with three loops. Middle: Noise and drift added to the consecutive motion. Bottom: Trajectory after adjustment of the loops.

Acknowledgements

References

Begelfor, E. & Werman, M., 2005: How to put Probabilities on Homographies. – IEEE Trans. on Pattern Analysis and Machine Intelligence **27** (10): 1666–1670.

Besl, P.J. & McKay, N.D., 1992: A Method for Registration of 3-D Shapes. – IEEE Trans. on Pattern Analysis and Machine Intelligence **14** (2): 239–256.

Bregler, C. & Malik, J., 1998: Tracking People with Twists and Exponential Maps. – Computer Vision and Pattern Recognition (CVPR **'98**): 8–15.

Caballero, F., Merino, L., Ferruz, J. & Ollero, A., 2007: Homography Based Kalman Filter for Mosaic Building. Applications to UAV Position Estimation. – International Conf. on Robotics and Automation (ICRA **2007**): 2004–2009.

Dickscheid, T., Läbe, Th. & Förstner, W., 2008: Benchmarking Automatic Bundle Adjustment Results. – **21**st Congress of the International Society for Photogrammetry and Remote Sensing (ISPRS) (B3a): 7–12.

Dubbelman, G., Esteban, I. & Schutte, K., 2010: Efficient trajectory bending with applications to loop closure. – Intelligent Robots and Systems (IROS **'10**): 1–7.

Estrada, C., Neira, J. & Tardós, J.D., 2005: Hierarchical SLAM : Real-Time Accurate Mapping of Large Environments. – IEEE Transactions on Robotics **21** (4): 588–596.

Fischler, M.A. & Bolles, R.C., 1981: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. – Communications of the ACM **24** (6): 381–395.

Förstner, W. & Gülch, E., 1987: A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features. – ISPRS Intercommission Workshop on Fast Processing of Photogrammetric Data: 281–305.

Förstner, W., 2010: Minimal Representations for Uncertainty and Estimation in Projective Spaces. – Asian Conf. on Computer Vision **II:** 619–633.

Förstner, W., 2012: Minimal Representations for Testing and Estimation in Projective Spaces. – PFG **2012** (3): 209–220.

Grisetti, G., Grzonka, S., Stachniss, C., Pfaff, P. & Burgard, W., 2007: Efficient Estimation of Accurate Maximum Likelihood Maps in 3D. – Intelligent Robots and Systems: 3472–3478, San Diego, CA, USA.

Hartley, R. & Zisserman, A., 2004: Multiple View Geometry in Computer Vision. – Second edition, Cambridge University Press, Cambridge.

Koch, K.-R., 1999: Parameter Estimation and Hypothesis Testing in Linear Models. – Second Edition, Springer, Berlin.

Lowe, D.G., 2004: Distinctive Image Features from Scale-Invariant Keypoints. – International Journal of Computer Vision **60** (2): 91–110.

Lucas, B.D. & Kanade, T., 1981: An Iterative Image Registration Technique with an Application to Stereo Vision. – Image Understanding Workshop: 212–130.

McGlone, J.C., Mikhail, E.M. & Bethel, J., 2004: Manual of Photogrammetry. – 5th ed., American Society of Photogrammetry and Remote Sensing.

Meidow, J., 2011: Efficient Video Mosaicking by Multiple Loop Closing. – U. Stilla et al. (eds.): Photogrammetric Image Analysis 2011, Lecture Notes in Computer Science **6952:** 1–12, Springer.

Strasdat, H., Montiel, J.M.M. & Davison, A.J., 2010: Scale Drift-Aware Large Scale Monocular SLAM. Robotics: Science and Systems.

Szeliski, R., 2006: Image alignment and stitching: A tutorial. – Foundations and Trends in Computer Graphics and Computer Vision **2** (1): 1–104.

Thomassen, C., 1997: On the complexity of finding a minimum cycle cover of a graph. – SIAM Journal on Computing **26** (3): 675–677.

Triggs, B., McLauchlan, P., Hartley, R. & Fitzgibbon, A., 2000: Bundle adjustment – a modern synthesis. – Triggs et al. (eds.): Vision Algorithms: Theory and Practice, Lecture Notes in Computer Science **1883:** 298–372, Springer.

Turkbeyler, E. & Harris, C., 2010: Mapping of Movement to Aerial Mosaic with Geo-Location Information. – Optro **2010**.

Unnikrishnan, R. & Kelly, A., 2002a: A Constrained Optimization Approach to Globally Consistent Mapping. – Intelligent Robots and Systems (IROS **'02**) 1: 564–569.

Unnikrishnan, R. & Kelly, A., 2002b: Mosaicing Large Cyclic Environments for Visual Navigation in Autonomous Vehicles. – IEEE International Conf. on Robotics and Automation 2002 (ICRA **'02**) 4: 4299–4306.

Address of the Author:

Dr.-Ing. Jochen Meidow, Fraunhofer-Institut für Optronik, Systemtechnik und Bildauswertung IOSB, D-76275 Ettlingen, Tel.: +49 7243 992-117, Fax: -299, e-mail: jochen.meidow@iosb.fraunhofer.de