Planar motion bundle adjustment

Citation for the original published paper (version of record):
Örnhag, M., Wadenbäck, M. (2019). Planar motion bundle adjustment. ICPRAM 2019 - Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods: 24-31.
http://dx.doi.org/10.5220/0007247700240031

Planar Motion Bundle Adjustment

Marcus Valtonen Örnhag[1] and Mårten Wadenbäck[2]
[1]*Centre for Mathematical Sciences, Lund University, Lund, Sweden*
[2]*Department of Mathematical Sciences, Chalmers University of Technology and the University of Gothenburg, Gothenburg, Sweden*

Keywords: Planar Motion, Bundle Adjustment.

Abstract: In this paper we consider trajectory recovery for two cameras directed towards the floor, and which are mounted rigidly on a mobile platform. Previous work for this specific problem geometry has focused on locally minimising an algebraic error between inter-image homographies to estimate the relative pose. In order to accurately track the platform globally it is necessary to refine the estimation of the camera poses and 3D locations of the feature points, which is commonly done by utilising bundle adjustment; however, existing software packages providing such methods do not take the specific problem geometry into account, and the result is a physically inconsistent solution. We develop a bundle adjustment algorithm which incorporates the planar motion constraint, and devise a scheme that utilises the sparse structure of the problem. Experiments are carried out on real data and the proposed algorithm shows an improvement compared to established generic methods.

1 INTRODUCTION

Structure from Motion (SfM) is a classic problem in computer vision, and concerns the simultaneous determination of the scene geometry (the structure) and the pose of the cameras (the motion) from a number of images of a scene (Hartley and Zisserman, 2004; Szeliski, 2011). Modern systems for SfM, such as the ones famously used to "build Rome" (Agarwal et al., 2011; Frahm et al., 2010) or the popular *Bundler* system (Snavely et al., 2008), are able to generate impressive reconstructions of increasingly large scenes from large unordered and unlabelled collections of images.

Among the key enabling technologies for these successes in large scale SfM are algorithms for performing Bundle Adjustment (BA), i.e. solving the SfM problem as a large optimisation problem (Triggs et al., 1999). In this optimisation problem, a cost function—often chosen as the sum of squared geometric reprojection errors—is to be minimised with respect to a set of parameters describing the scene geometry and the camera poses. Formulating SfM as a BA problem has a number of benefits when it comes to the problem modelling, e.g. (a) it can, in a unified way, incorporate assumptions concerning the camera calibration, (b) it allows the use of a suitable parametrisation and/or explicit constraints for the purpose of enforcing a particular motion model, and (c) the cost function can be chosen to be a physically meaningful quantity, as opposed to a purely algebraic error.

Due to its relatively high computational cost, BA has traditionally been applied mainly in offline batch processing systems such as the ones mentioned above. During the last two decades, however, BA has started to surface in online SfM systems used for camera based Simultaneous Localisation and Mapping (SLAM) and Visual Odometry (VO), where it can be performed at regular intervals e.g. to reduce scale drift or to improve consistency in general. This development has been driven by improvements in the performance of hardware as well as by advances in the algorithms and their implementation, and we anticipate that more and more application specific implementations of BA will move towards real-time systems.

In visual SLAM there is sometimes additional information available compared to a general SfM problem, and this can be exploited to improve the performance of the system. For instance, the images are acquired in an ordered sequence, and this avoids expensive "all-vs-all" matching of the images when searching for correspondences. Another possible source of valuable information is a suitable motion model, which can be used e.g. for feature prediction (Davison et al., 2007; Davison, 2003), to exploit

Örnhag, M. and Wadenbäck, M.
Planar Motion Bundle Adjustment.
DOI: 10.5220/0007247700240031
In *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2019)*, pages 24-31
ISBN: 978-989-758-351-3

non-holonomic constraints (Zienkiewicz and Davison, 2015; Scaramuzza, 2011a; Scaramuzza, 2011b), or to constrain the camera motion to a plane (Hajjdiab and Laganière, 2004; Ortín and Montiel, 2001; Wadenbäck and Heyden, 2014).

In this paper, we will consider BA in the constrained planar motion case for a pair of cameras—not necessarily with overlapping fields of view—attached to a mobile platform in such a way that each of the two cameras are subject to a planar motion model, in addition to the rigid body motion connecting them.

2 RELATED WORK

An early approach by Ortín and Montiel used a planar motion model to parametrise the essential matrix between successive views in terms of two translation parameters and one rotation angle, which allowed the relative motion to be recovered from two point correspondences using a non-linear solver (Ortín and Montiel, 2001). One limitation of this approach is that it does not contain any way to determine the length of the translation between the camera positions. Another limitation is that the camera must be mounted in such a way that the optical axis is horizontal, to a reasonably high precision, in order to allow the simple parametrisation employed. A similar approach was considered for the stereo case in (Chen and Liu, 2006).

In the monocular case, the problems of scale ambiguity and scale drift are connected to the use of the fundamental matrix to solve the relative pose problem. An additional drawback of these methods is that the fundamental matrix cannot be determined from co-planar correspondences—see e.g. (Hartley and Zisserman, 2004) for further discussion of this degeneracy—which is a considerable issue in indoor environments where planar structures are common. These considerations, among others, have led researchers to consider homography based methods instead.

The homography based method by Liang and Pears used correspondences in the ground plane, together with a planar motion model (Liang and Pears, 2002). They showed that the rotation angle about the vertical axis can be found via the eigenvalues of the homography matrix, regardless of how the camera is mounted. A similar geometric situation, but allowing only one tilt angle in the possible camera orientations, was studied in (Hajjdiab and Laganière, 2004). They also devised an effective scheme for estimating the full set of motion parameters.

More recent work by Wadenbäck and Heyden extended the homography based methods for planar motion and co-planar keypoints to the general 5-DoF situation (Wadenbäck and Heyden, 2013). Their method used a decoupling of the underlying 2D rigid body motion from the camera tilt, which was first estimated iteratively. The same general geometric situation was also considered by Zienkiewicz and Davison, who proposed a dense matching of the images based on non-linear optimisation for determining the correct motion parameters (Zienkiewicz and Davison, 2015).

The general 5-DoF situation was extended to a binocular setup, with possibly non-overlapping fields of view, in (Valtonen Örnhag and Heyden, 2018a; Valtonen Örnhag and Heyden, 2018b). The cameras were assumed to be connected by a rigid body motion, and it was shown that it is possible to recover the relative pose between the cameras.

Bundle adjustment is a well-studied problem and an excellent overview is that of (Triggs et al., 1999), mentioned in the introduction. Since bundle adjustment often involves solving a large system of equations it is necessary to account for the structure of the problem, e.g. by exploiting sparsity patterns. One sparse bundle adjustment package available is SBA (Lourakis and Argyros, 2009), which utilises the sparsity in the Jacobian matrix by using Schur complementation in order to speed up the algorithm. The SBA package has been successfully used in e.g. (Snavely et al., 2008; Agarwal et al., 2011; Frahm et al., 2010).

Among more recent implementations of sparse bundle adjustment is *Sparse Sparse Bundle Adjustment* (sSBA) (Konolige, 2010) and *Simple Sparse Bundle Adjustment* (SSBA) (Zach, 2014), which uses a similar approach as SBA in the sense that the augmented normal equations are solved, but utilises packages that exploit the sparsity more efficiently. To speed up convergence, and move into the domain of real-time applications, *Parallel Bundle Adjustment* was introduced in (Wu et al., 2011) which supports GPU acceleration where a preconditioned Conjugate Gradients (CG) system is solved. Another GPU implementation by Hänsch et al. shows that is is possible to efficiently parallelise the Levenberg-Marquardt algorithm (LM) (Hänsch et al., 2016). Recent papers dealing with very large scale SfM problems have succesfully employed distributed methods, by employing splitting methods (Eriksson et al., 2016; Zhang et al., 2017).

Furthermore, choosing the cost function to be the sum of squared geometric reprojection errors is not the only viable option—for monocular visual odometry photometric bundle adjustment, where the photo-

metric consistency is maximised, has proven to be a good competitor (Alismail et al., 2016).

3 THEORY

3.1 Problem Geometry

In this paper we consider a mobile platform with two cameras directed towards the floor. The world coordinate system is chosen such that the ground floor is positioned at $z = 0$, whereas the cameras move in the planes $z = a$ and $z = b$, respectively. Furthermore, the fields of view of the cameras are not assumed to be overlapping, and both cameras are assumed to be mounted rigidly onto the platform. Due to this setup, the cameras are connected by a rigid body motion, and, without loss of generality, we may assume that the centre of rotation of the mobile platform is located in the first camera centre, as is illustrated in Figure 1.

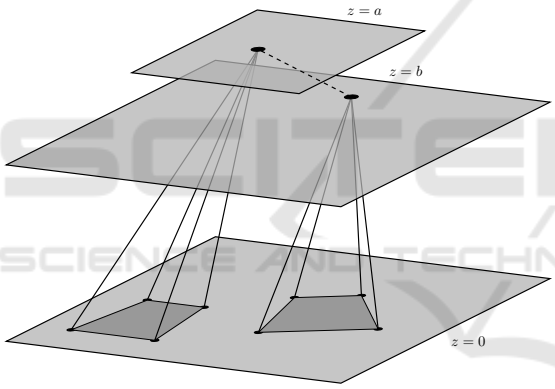


Figure 1: The problem geometry considered in this paper. The cameras are assumed to move in the planes $z = a$ and $z = b$, the relative orientation between them as well as the tilt towards the floor normal is assumed to be constant as the mobile platform moves freely.

3.2 Camera Parametrisation

A camera parametrisation well suited to the internally calibrated monocular case of this specific problem geometry was derived in Wadenbäck and Heyden (Wadenbäck and Heyden, 2013), and we adopt this parametrisation here. The camera matrix associated with the image taken at position j is thus written as

$$\boldsymbol{P}^{(j)} = \boldsymbol{R}_{\psi\theta}\boldsymbol{R}_{\varphi}^{(j)}[\boldsymbol{I} \mid -\boldsymbol{t}^{(j)}], \tag{1}$$

where $\boldsymbol{R}_{\psi\theta}$ is a rotation θ about the y-axis followed by a rotation of ψ about the x-axis. The movement of the mobile platform is modelled by a rotation $\varphi^{(j)}$ about the z-axis, corresponding to $\boldsymbol{R}_{\varphi}^{(j)}$, and a translation vector $\boldsymbol{t}^{(j)}$. In (Valtonen Örnhag and Heyden, 2018a) the camera matrices for the second camera are parametrised as

$$\boldsymbol{P}'^{(j)} = \boldsymbol{R}_{\psi'\theta'}\boldsymbol{R}_{\eta}\boldsymbol{T}_{\boldsymbol{\tau}}(b)\boldsymbol{R}_{\varphi}^{(j)}[\boldsymbol{I} \mid -\boldsymbol{t}^{(j)}], \tag{2}$$

where ψ' and θ' are the tilt angles, defined analogously as for the first camera, $\boldsymbol{\tau}$ is the relative translation between the camera centres and η is the constant rotation about the z-axis relative to the first camera. None of the constant parameters are assumed to be known. The translation matrix $\boldsymbol{T}_{\boldsymbol{\tau}}(b)$ is defined as $\boldsymbol{T}_{\boldsymbol{\tau}}(b) = \boldsymbol{I} - \boldsymbol{\tau}\boldsymbol{n}^{\mathsf{T}}/b$, where $\boldsymbol{\tau} = (\tau_x, \tau_y, 0)^{\mathsf{T}}$, $\boldsymbol{n}$ is a floor normal and b is the height above the ground floor. Due to the global scale ambiguity we may assume $a = 1$.

4 PREREQUISITES

4.1 Geometric Reprojection Error

Consider the pose of the first camera at position j, given by the camera matrix in (1), and let $\hat{\boldsymbol{x}}_i^{(j)}$ denote the estimated measurement of the scene point $\boldsymbol{X}_i$ in homogeneous coordinates, i.e. $\hat{\boldsymbol{x}}_i^{(j)} \sim \boldsymbol{P}^{(j)}\boldsymbol{X}_i$. Let $\boldsymbol{x}_i^{(j)}$ denote the measured image point, and define the residual $\boldsymbol{r}_{ij}$ as $\boldsymbol{r}_{ij} = \boldsymbol{x}_i^{(j)} - \hat{\bar{\boldsymbol{x}}}_i^{(j)}$, where $\hat{\bar{\boldsymbol{x}}}_i^{(j)}$ is the inhomogeneous representation of $\hat{\boldsymbol{x}}_i^{(j)}$. Analogously to the first camera, define the residual $\boldsymbol{r}'_{ij}$ for the image of $\boldsymbol{X}_i$ in the second camera. Given N stereo camera locations and M scene points, we seek to minimise the geometric reprojection error E given by

$$E(\boldsymbol{\beta}) = \sum_{i=1}^{N}\sum_{j=1}^{M} \|\boldsymbol{r}_{ij}\|_2^2 + \|\boldsymbol{r}'_{ij}\|_2^2, \tag{3}$$

where $\boldsymbol{\beta}$ is the parameter vector consisting of the camera parameters and the scene points.

4.2 The Levenberg-Marquardt Algorithm

For bundle adjustment it is common to use the Levenberg-Marquardt algorithm (LM) which solves the augmented normal equations

$$\left(\boldsymbol{J}^{\mathsf{T}}\boldsymbol{J} + \mu\boldsymbol{I}\right)\boldsymbol{\delta} = \boldsymbol{J}^{\mathsf{T}}\boldsymbol{\varepsilon}, \tag{4}$$

where $\boldsymbol{J}$ is the Jacobian of the cost function and $\boldsymbol{\varepsilon}$ the residual vector. The reader is referred to (Triggs et al., 1999; Lourakis and Argyros, 2009) for more details regarding the LM algorithm and its application

to bundle adjustment. There are other options to the LM algorithm, e.g. the dog-leg solver (Lourakis and Argyros, 2005) and preconditioned CG (Byröd and Åström, 2010); however, LM is one of the most commonly used algorithms today, and is used in modern systems such as SBA (Lourakis and Argyros, 2009) and sSBA (Konolige, 2010). Note, however, that SBA assumes the camera parameters for each camera to be decoupled, which is not the case for this specific problem geometry.

4.3 Initial Solution for Camera Poses

A good initial solution for the camera poses can be generated using the method described in Wadenbäck and Heyden (Wadenbäck and Heyden, 2014) for the monocular case. The method takes as input the inter-image homographies for the path. These can be estimated using Direct Linear Transform (DLT) (Hartley and Zisserman, 2004) from point correspondences established by automatic matching of features, e.g. SIFT (Lowe, 2004) or SURF (Bay et al., 2006). Regardless of how the homographies are obtained, the method continues to estimate the overhead tilt $\boldsymbol{R}_{\psi\theta}$ from one or several homographies, and then computes the translation and orientation about the floor normal by QR decomposition of $\boldsymbol{R}_{\psi\theta}^{\mathsf{T}}\boldsymbol{H}\boldsymbol{R}_{\psi\theta}$.

In order to initialise the stereo parameters the method proposed in (Valtonen Örnhag and Heyden, 2018a) can be used. This method relies on the the estimates from the monocular method described above, by first treating the two trajectories individually. When the trajectories are known individually, the relative pose between the two cameras may be extracted. Both methods benefit from using more than one homography to estimate the motion of the mobile platform.

4.4 Triangulation of 3D Points

Linear triangulation of 3D points can be posed as finding the null-space of a matrix relating the scene points and the camera matrices, see e.g. (Hartley and Zisserman, 2004); however, this may not result in a physically meaningful solution in the sense that all points may not be on the plane $z = 0$. There is a homography between the measured points and the ground plane; namely, given an image point $\boldsymbol{x}$ and the corresponding camera $\boldsymbol{P}$ and scene point $\boldsymbol{X} \sim (X,\, Y,\, 0,\, 1)^{\mathsf{T}}$ it holds that $\boldsymbol{x} \sim \boldsymbol{P}\boldsymbol{X} = \boldsymbol{H}\tilde{\boldsymbol{X}}$. Let $\boldsymbol{P}_i$ denote the i:th column of $\boldsymbol{P}$, then $\boldsymbol{H} = [\boldsymbol{P}_1\ \boldsymbol{P}_2\ \boldsymbol{P}_4]$ is the homography we seek and $\tilde{\boldsymbol{X}} \sim (X,\, Y,\, 1)^{\mathsf{T}}$ contains the unknown scene point coordinates. It follows that the corresponding scene point can be extracted from $\tilde{\boldsymbol{X}} \sim \boldsymbol{H}^{-1}\boldsymbol{x}$.

Having more than one camera will generally result in different 3D points; however, all of them will be on the plane $z = 0$. We use a simple heuristic to triangulate the points by computing the centre of mass. This is fast, but suffers from the presence of outliers, which must be removed prior to triangulation in order to achieve reasonable performance.

5 PLANAR MOTION BUNDLE ADJUSTMENT

5.1 Block Structure

Consider the general case of N stereo camera positions, with M scene points. For convenience, let $\boldsymbol{\gamma} = (\psi,\, \theta)$ and $\boldsymbol{\gamma}' = (\psi',\, \theta',\, \tau_x,\, \tau_y,\, b,\, \eta)$ denote the unknown and constant parameters for each camera path and $\boldsymbol{\xi}_j = (\varphi^{(j)},\, t_x^{(j)},\, t_y^{(j)})$ be the nonconstant parameters for position j. Then, the Jacobian $\boldsymbol{J}$ has the following block structure:

$$\boldsymbol{J} = \begin{bmatrix}
\boldsymbol{\Gamma}_{11} & & & \boldsymbol{A}_{11} & & & \boldsymbol{B}_{11} & & \\
\vdots & & & \ddots & & & \vdots & & \\
\boldsymbol{\Gamma}_{1N} & & & & \boldsymbol{A}_{1N} & & \boldsymbol{B}_{1N} & & \\
\vdots & & & \vdots & & \ddots & & & \\
\boldsymbol{\Gamma}_{M1} & & & \boldsymbol{A}_{M1} & & & & & \boldsymbol{B}_{M1} \\
\vdots & & & & \ddots & & & & \vdots \\
\boldsymbol{\Gamma}_{MN} & & & & & \boldsymbol{A}_{MN} & & & \boldsymbol{B}_{MN} \\
\boldsymbol{\Gamma}'_{11} & & & \boldsymbol{A}'_{11} & & & \boldsymbol{B}'_{11} & & \\
\vdots & & & \ddots & & & \vdots & & \\
\boldsymbol{\Gamma}'_{1N} & & & & \boldsymbol{A}'_{1N} & & \boldsymbol{B}'_{1N} & & \\
\vdots & & & \vdots & & \ddots & & & \\
\boldsymbol{\Gamma}'_{M1} & & & \boldsymbol{A}'_{M1} & & & & & \boldsymbol{B}'_{M1} \\
\vdots & & & & \ddots & & & & \vdots \\
\boldsymbol{\Gamma}'_{MN} & & & & & \boldsymbol{A}'_{MN} & & & \boldsymbol{B}'_{MN}
\end{bmatrix}, \tag{5}$$

where the derivative blocks are defined as

$$\begin{aligned}
\boldsymbol{A}_{ij} &= \frac{\partial \boldsymbol{r}_{ij}}{\partial \boldsymbol{\xi}_j}, \quad \boldsymbol{B}_{ij} = \frac{\partial \boldsymbol{r}_{ij}}{\partial \tilde{\boldsymbol{X}}_i}, \qquad \boldsymbol{\Gamma}_{ij} = \frac{\partial \boldsymbol{r}_{ij}}{\partial \boldsymbol{\gamma}}, \\
\boldsymbol{A}'_{ij} &= \frac{\partial \boldsymbol{r}'_{ij}}{\partial \boldsymbol{\xi}_j}, \quad \boldsymbol{B}'_{ij} = \frac{\partial \boldsymbol{r}'_{ij}}{\partial \tilde{\boldsymbol{X}}_i}, \qquad \boldsymbol{\Gamma}'_{ij} = \frac{\partial \boldsymbol{r}'_{ij}}{\partial \boldsymbol{\gamma}'},
\end{aligned} \tag{6}$$

and where $\tilde{\boldsymbol{X}}_i = (X_i,\, Y_i)$ are the unknown scene coordinates. We write this compactly as

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{\Gamma} & \boldsymbol{0} & \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{\Gamma}' & \boldsymbol{A}' & \boldsymbol{B}' \end{bmatrix}. \tag{7}$$

5.2 Utilising the Sparse Structure

As in SBA (Lourakis and Argyros, 2009) and other similar frameworks, we would like to use Schur complementation; however, it is not directly applicable

due to the contributions from the constant parameters. Consider, the approximate Hessian $\boldsymbol{J}^\mathsf{T}\boldsymbol{J}$ in the compact form

$$\boldsymbol{J}^\mathsf{T}\boldsymbol{J} = \begin{bmatrix} \boldsymbol{C} & \boldsymbol{E} \\ \boldsymbol{E}^\mathsf{T} & \boldsymbol{D} \end{bmatrix}, \tag{8}$$

where $\boldsymbol{C}$ contains the contribution from the constant parameters, $\boldsymbol{D}$ contains the contribution from the non-constant parameters and the scene points and $\boldsymbol{E}$ the mixed contributions. The matrix $\boldsymbol{D}$ may further be decomposed into

$$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{U} & \boldsymbol{W} \\ \boldsymbol{W}^\mathsf{T} & \boldsymbol{V} \end{bmatrix}, \tag{9}$$

with block diagonal matrices $\boldsymbol{U} = \operatorname{diag}(\boldsymbol{U}_1, \ldots, \boldsymbol{U}_N)$ and $\boldsymbol{V} = \operatorname{diag}(\boldsymbol{V}_1, \ldots, \boldsymbol{V}_M)$, where

$$\begin{aligned} \boldsymbol{U}_j &= \sum_{i=1}^{M} \boldsymbol{A}_{ij}^\mathsf{T}\boldsymbol{A}_{ij} + \boldsymbol{A}_{ij}'^\mathsf{T}\boldsymbol{A}_{ij}', \\ \boldsymbol{V}_i &= \sum_{j=1}^{N} \boldsymbol{B}_{ij}^\mathsf{T}\boldsymbol{B}_{ij} + \boldsymbol{B}_{ij}'^\mathsf{T}\boldsymbol{B}_{ij}', \\ \boldsymbol{W}_{ij} &= \boldsymbol{A}_{ij}^\mathsf{T}\boldsymbol{B}_{ij} + \boldsymbol{A}_{ij}'^\mathsf{T}\boldsymbol{B}_{ij}' \,. \end{aligned} \tag{10}$$

The solution to a system on the form $(\boldsymbol{D} + \mu\boldsymbol{I})\boldsymbol{\delta} = \boldsymbol{\varepsilon}$, where $\boldsymbol{D}$ is defined as in (9), is well-known, and is solved efficiently, with minor modifications, using existing software packages by utilising Schur complementation.

The main idea of our method is to incorporate the constant parameters and consider the decomposition of (8) as nested Schur complements, which reduces the problem to the form used in SBA and other well-established software packages, which in turn can be efficiently solved. To achieve this, consider the augmented normal equations (4) in block form

$$\begin{bmatrix} \boldsymbol{C}^* & \boldsymbol{E} \\ \boldsymbol{E}^\mathsf{T} & \boldsymbol{D}^* \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta}_c \\ \boldsymbol{\delta}_d \end{bmatrix} = \begin{bmatrix} \boldsymbol{\varepsilon}_c \\ \boldsymbol{\varepsilon}_d \end{bmatrix}, \tag{11}$$

where $\boldsymbol{C}^*$ and $\boldsymbol{D}^*$ denote the augmented matrices, with the μ term added on the main diagonals, as in (4). Applying Schur complementation yields

$$\begin{bmatrix} \boldsymbol{C}^* - \boldsymbol{E}\boldsymbol{D}^{*-1}\boldsymbol{E}^\mathsf{T} & \boldsymbol{0} \\ \boldsymbol{E}^\mathsf{T} & \boldsymbol{D}^* \end{bmatrix} \begin{bmatrix} \boldsymbol{\delta}_c \\ \boldsymbol{\delta}_d \end{bmatrix} = \begin{bmatrix} \boldsymbol{\varepsilon}_c - \boldsymbol{E}\boldsymbol{D}^{*-1}\boldsymbol{\varepsilon}_d \\ \boldsymbol{\varepsilon}_d \end{bmatrix} \tag{12}$$

Some remarks are in order. First, note that $\boldsymbol{D}^{*-1}$ appears in (12) twice, and is infeasible to compute explicitly, which is avoided using the following observations: introduce the auxiliary variable $\boldsymbol{\delta}_{\text{aux}}$, such that

$$\boldsymbol{D}^*\boldsymbol{\delta}_{\text{aux}} = \boldsymbol{\varepsilon}_d, \tag{13}$$

which can be solved with e.g. SBA. In a similar manner $\boldsymbol{D}^*\boldsymbol{\Delta}_{\text{aux}} = \boldsymbol{E}^\mathsf{T}$ can be solved by iterating over the columns of $\boldsymbol{E}^\mathsf{T}$. This may at first seem like a time consuming task, however, given that the number of constant parameters is low—as in the problem geometry considered in this paper—having already solved for (13), the computation of the Schur complement, as well as intermediate matrices not depending on the right hand side, can be stored and reused.

When the auxiliary variables are solved for, it is possible to compute $\boldsymbol{\delta}_c$ from

$$\left(\boldsymbol{C}^* - \boldsymbol{E}\boldsymbol{\Delta}_{\text{aux}}\right)\boldsymbol{\delta}_c = \boldsymbol{\varepsilon}_c - \boldsymbol{E}\boldsymbol{\delta}_{\text{aux}}, \tag{14}$$

and, finally, for $\boldsymbol{\delta}_d$ by back-substitution

$$\boldsymbol{D}^*\boldsymbol{\delta}_d = \boldsymbol{\varepsilon}_d - \boldsymbol{E}^\mathsf{T}\boldsymbol{\delta}_c \,. \tag{15}$$

Again, note the resemblance of (13) and (15), hence the computation of the Schur complement and intermediate matrices can be stored and reused.

6 EXPERIMENTS

6.1 Initial Solution

The inter-image homographies were estimated using the MSAC algorithm (Torr and Zisserman, 2000) from four point correspondences by extracting SURF keypoints and applying a KNN algorithm to establish the matches.

Using all available homographies, the monocular parameters were recovered by the method proposed in (Wadenbäck and Heyden, 2014) and the binocular parameters using (Valtonen Örnhag and Heyden, 2018a). The output is given in terms of the relative pose between the frames, and by aligning the first camera position to the origin the absolute poses for the remaining cameras can be computed. Knowing the poses, the scene points were triangulated as discussed in Section 4.4.

6.2 Choice of Dataset

Due to the lack of a good and established planar motion evaluation dataset the KITTI Visual Odometry / SLAM benchmark (Geiger et al., 2012), was chosen to demonstrate the proposed method. The dataset contains several sequences and subsequences of planar or near planar motion, in which a significant portion of the images depict the road. There are, however, sequences containing irregularities in the road causing the camera to move up and down, which is not a valid motion according to the planar motion model. Such sequences were shortened to contain images where the assumption is a reasonable approximation. Furthermore, it is known *a priori* that parts

Figure 2: Images from the KITTI Visual Odometry / SLAM benchmark, Sequence 01 (left) and 03 (right). The input images are cropped (thick border) in order for them to contain a large portion of a near planar surface. This assumption is only valid in a subset of the sequences, e.g. the highway of Sequence 01 (left). In many cases occlusions, such as the car in Sequence 03 (right), or the non-planar background surface, is not approximated well by the planar motion model. These situations often occur at road intersections and turns. Image credit: KITTI dataset (Geiger et al., 2012).

of the image is not approximated well by the planar motion model, e.g. the sky and non-planar structures often visible on the side of the road. Therefore these parts are cropped out before estimating the homography, see Figure 2.

6.3 Bundle Adjustment Comparison

From the initial trajectory a general 6-DoF model was used and solved with SBA for both camera trajectories and compared to the proposed method. For a fair comparison, no feature points were matched between the stereo views, as to demonstrate that the overlapping of fields of view are not necessary to achieve better performance. The different BA algorithms used the same settings for termination and control of the damping parameter μ. The results are shown in Figure 3.

In all cases the performance of the proposed method is better or as good as the ones obtained with the general 6-DoF model and SBA. In the cases where the initial trajectory is irregular SBA often converges to a solution where these irregularities are still present, and thus produces a physically improbable solution. This phenomenon is rarely seen in our method, which converges to a smooth trajectory under fairly general circumstances, regardless of the initial solution. This holds true even in cases where the planar motion model is not valid, see e.g. Figure 3(b) depicting Sequence 03, where the subsequence of the turn in the road, cf. Figure 2, is non-planar—apart from the degree of the turn being too sharp, the remaining characteristics of the ground truth trajectory are present, which is not the case for the general 6-DoF model.

7 CONCLUSION

In this paper we have devised a bundle adjustment method taking the specific planar motion problem geometry into account. An implementation scheme that utilises the sparse structure of the problem has been proposed and the method has been tested on subsequences of the KITTI Visual Odometry / SLAM benchmark and compared to state-of-the-art methods for sparse bundle adjustment. The results show that the method performs well and gives a physically reasonable solution, despite some of the model assumptions not being fulfilled.

ACKNOWLEDGEMENTS

This work has been funded by the Swedish Research Council through grant no. 2015-05639 'Visual SLAM based on Planar Homographies'.

REFERENCES

Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S. M., and Szeliski, R. (2011). Building Rome in a Day. *Communications of the ACM*, 54(10):105–112.

Alismail, H., Browning, B., and Lucey, S. (2016). Photometric Bundle Adjustment for Vision-Based SLAM. In *Asian Conference on Computer Vision (ACCV)*, pages 324–341, Taipei, Taiwan.

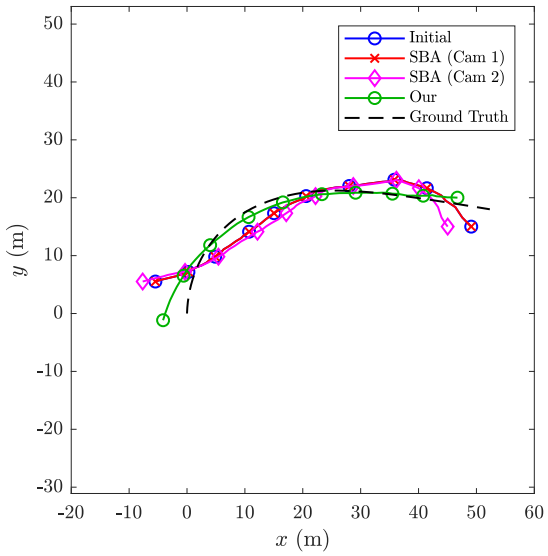
Bay, H., Tuytelaars, T., and Van Gool, L. (2006). SURF: Speeded Up Robust Features. In *European Conference on Computer Vision (ECCV)*, pages 404–417, Graz, Austria.

Byröd, M. and Åström, K. (2010). Conjugate gradient bundle adjustment. In *European Conference on Computer Vision (ECCV)*, pages 114–127, Heraklion, Crete, Greece.

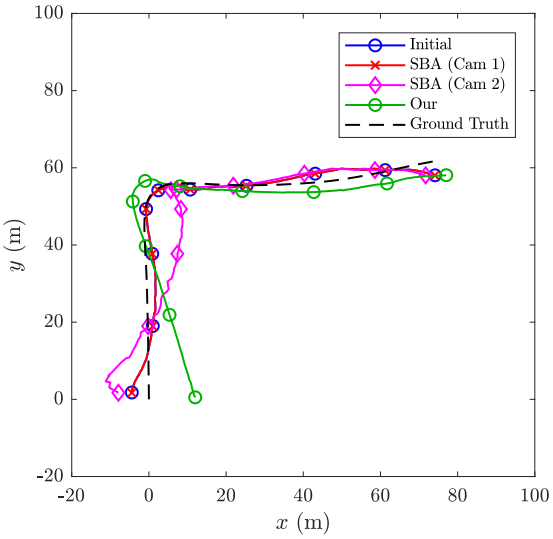
Chen, T. and Liu, Y.-H. (2006). A robust approach for structure from planar motion by stereo image sequences. *Machine Vision and Applications (MVA)*, 17(3):197–209.

Davison, A. J. (2003). Real-Time Simultaneous Localisation and Mapping with a Single Camera. In *International Conference on Computer Vision (ICCV)*, pages 1403–1410, Nice, France.

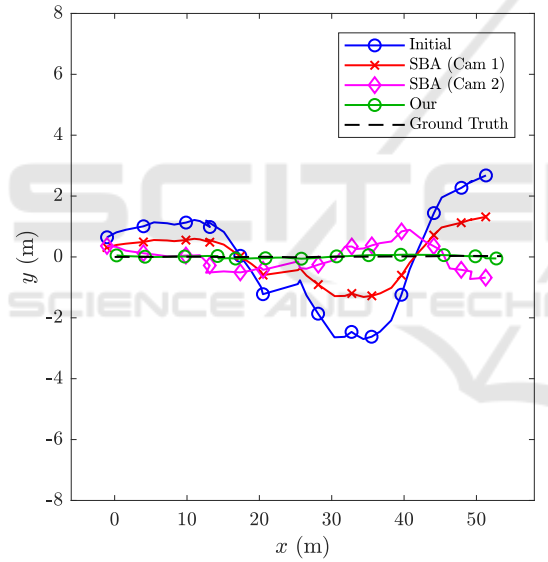
Davison, A. J., Reid, I. D., Molton, N. D., and Stasse, O. (2007). MonoSLAM: Real-Time Single Camera



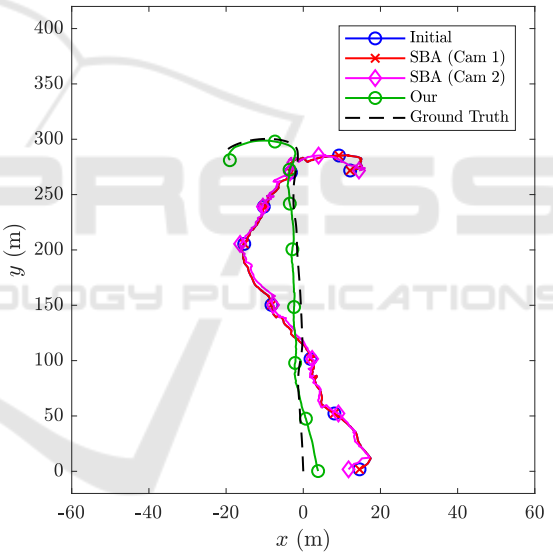
(a) Sequence 01 (60 images).



(b) Sequence 03 (200 images).



(c) Sequence 04 (40 images).



(d) Sequence 06 (330 images).

Figure 3: Estimated trajectories of subsequences of Sequence 01, 03, 04 and 06. Procrustes analysis has been carried out to align the estimated paths with the ground truth. N.B. the axes do not have the same aspect ratio in (c) in order to clearly visualise the difference.

SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29(6):1052–1067.

Eriksson, A., Bastian, J., Chin, T., and Isaksson, M. (2016). A consensus-based framework for distributed bundle adjustment. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1754–1762, Las Vegas, NV, USA.

Frahm, J.-M., Fite-Georgel, P., Gallup, D., Johnson, T., Raguram, R., Wu, C., Jen, Y.-H., Dunn, E., Clipp, B., Lazebnik, S., and Pollefeys, M. (2010). Building Rome on a Cloudless Day. In *European Conference on Computer Vision (ECCV)*, pages 368–381, Heraklion, Crete, Greece.

Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, Providence, RI, USA.

Hajjdiab, H. and Laganière, R. (2004). Vision-based Multi-Robot Simultaneous Localization and Mapping. In *Canadian Conference on Computer and Robot Vision (CRV)*, pages 155–162, London, ON, Canada.

Hänsch, R., Drude, I., and Hellwich, O. (2016). Mod-

ern Methods of Bundle Adjustment on the GPU. In *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Congress)*, pages 43–50, Prague, Czech Republic.

Hartley, R. I. and Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, England, UK, second edition.

Konolige, K. (2010). Sparse Sparse Bundle Adjustment. In *British Machine Vision Conference (BMVC)*, pages 102.1–11, Aberystwyth, Wales, UK.

Liang, B. and Pears, N. (2002). Visual Navigation using Planar Homographies. In *International Conference on Robotics and Automation (ICRA)*, pages 205–210, Washington, DC, USA.

Lourakis, M. I. A. and Argyros, A. A. (2005). Is levenberg-marquardt the most efficient optimization algorithm for implementing bundle adjustment? In *International Conference on Computer Vision (ICCV)*, pages 1526–1531, Beijing, China (PRC).

Lourakis, M. I. A. and Argyros, A. A. (2009). SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Transactions on Mathematical Software (TOMS)*, 36(1):2:1–2:30.

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

Ortín, D. and Montiel, J. M. M. (2001). Indoor robot motion based on monocular images. *Robotica*, 19(3):331–342.

Scaramuzza, D. (2011a). 1-Point-RANSAC Structure from Motion for Vehicle-Mounted Cameras by Exploiting Non-holonomic Constraints. *International Journal of Computer Vision (IJCV)*, 95(1):74–85.

Scaramuzza, D. (2011b). Performance Evaluation of 1-Point-RANSAC Visual Odometry. *Journal of Field Robotics (JFR)*, 28(5):792–811.

Snavely, N., Seitz, S. M., and Szeliski, R. (2008). Modeling the World from Internet Photo Collections. *International Journal of Computer Vision (IJCV)*, 80(2):189–210.

Szeliski, R. (2011). *Computer Vision: Applications and Algorithms*. Springer-Verlag, London, England, UK.

Torr, P. H. S. and Zisserman, A. (2000). MLESAC: A New Robust Estimator with Application to Estimating Image Geometry. *Computer Vision and Image Understanding (CVIU)*, 78(1):138–156.

Triggs, B., McLauchlan, P. F., Hartley, R. I., and Fitzgibbon, A. W. (1999). Bundle Adjustment — A Modern Synthesis. In *International Workshop on Vision Algorithms — Vision Algorithms: Theory and Practice*, pages 298–372, Corfu, Greece.

Valtonen Örnhag, M. and Heyden, A. (2018a). Generalization of Parameter Recovery in Binocular Vision for a Planar Scene. In *International Conference on Pattern Recognition and Artificial Intelligence*, pages 37–42, Montréal, Canada.

Valtonen Örnhag, M. and Heyden, A. (2018b). Relative Pose Estimation in Binocular Vision for a Planar Scene using Inter-Image Homographies. In *International Conference on Pattern Recognition Applications and Methods (ICPRAM)*, pages 568–575, Funchal, Madeira, Portugal.

Wadenbäck, M. and Heyden, A. (2013). Planar Motion and Hand-Eye Calibration Using Inter-Image Homographies from a Planar Scene. In *International Conference on Computer Vision Theory and Applications (VISAPP)*, pages 164–168, Barcelona, Spain.

Wadenbäck, M. and Heyden, A. (2014). Ego-Motion Recovery and Robust Tilt Estimation for Planar Motion Using Several Homographies. In *International Conference on Computer Vision Theory and Applications (VISAPP)*, pages 635–639, Lisbon, Portugal.

Wu, C., Agarwal, S., Curless, B., and Seitz, S. M. (2011). Multicore Bundle Adjustment. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3057–3064, Providence, RI, USA.

Zach, C. (2014). Robust Bundle Adjustment Revisited. In *European Conference on Computer Vision (ECCV)*, pages 772–787, Zurich, Switzerland.

Zhang, R., Zhu, S., Fang, T., and Quan, L. (2017). Distributed very large scale bundle adjustment by global camera consensus. In *International Conference on Computer Vision (ICCV)*, pages 29–38, Venice, Italy.

Zienkiewicz, J. and Davison, A. J. (2015). Extrinsics Autocalibration for Dense Planar Visual Odometry. *Journal of Field Robotics (JFR)*, 32(5):803–825.