B-splines for Purely Vision-based Localization and Mapping on Non-holonomic Ground Vehicles

Kun Huang[1], Yifu Wang[2] and Laurent Kneip[1]

***Abstract*— Purely vision-based localization and mapping is a cost-effective and thus attractive solution to localization and mapping on smart ground vehicles. However, the accuracy and especially robustness of vision-only solutions remain rivalled by more expensive, lidar-based multi-sensor alternatives. We show that a significant increase in robustness can be achieved if taking non-holonomic kinematic constraints on the vehicle motion into account. Rather than using approximate planar motion models or simple, pair-wise regularization terms, we demonstrate the use of B-splines for an exact imposition of smooth, non-holonomic trajectories inside the 6 DoF bundle adjustment. We introduce both hard and soft formulations and compare their computational efficiency and accuracy against traditional solutions. Through results on both simulated and real data, we demonstrate a significant improvement in robustness and accuracy in degrading visual conditions.**

I. Introduction

Visual Simultaneous Localization And Mapping (SLAM) is a long-standing problem within the computer vision and robotics communities. Nonetheless, pure vision-based solutions lack the level of robustness found in laser-based solutions, and are thus often complemented by additional sensors such as—on ground vehicles—encoders measuring the rotational velocity of the wheels. The installation of wheel encoders on existing platforms is however difficult, and accessing the signals of existing encoders may be prevented by the manufacturer. As a result, the development of robust, purely vision-based (or inertial-supported) SLAM solutions remains a relevant topic in the development of self-driving vehicles. The present paper presents such a solution.

The trajectory in visual SLAM frameworks is commonly represented by a discrete set of camera poses each associated with one of the captured images. It is well known that this representation is too general and does in fact not respect the kinematic motion constraints of ground vehicles. The core idea of the present paper consists of increasing the robustness of a purely vision-based SLAM framework by employing a more restrictive but exact geometric representation for the kind of smooth trajectory that we have on drift-less, non-holonomic ground vehicles.

We make the following contributions:

- We use the B-spline based, smooth, continuous-time trajectory representation introduced by Furgale et al. [1] to represent the motion of ground vehicles.
- Rather than just being employed as a smooth trajectory model, we illustrate how the representation can be altered in order to incorporate the kinematic constraints on non-holonomic, drift-less ground vehicles.
- We introduce different hard and soft variants of the additional constraints, and compare the resulting frameworks against conventional solutions.
- We demonstrate a significant advantage in robustness and accuracy on both simulated and real data.

The paper is organized as follows. Additional related work is discussed in Section II. Section III provides a brief review of B-splines and kinematic motion constraints on drift-less non-holonomic platforms. Section IV introduces different realizations of the objective, and Section V finally concludes with our results on both simulated and real data.

II. Related work

The present work considers the improvement of a pure monocular SLAM solution by including vehicle kinematics related constraints into the optimization framework. This technique is already commonly applied to vision-based multi-sensor solutions that make additional use of odometers measuring the rotational velocity of each wheel. There have been EKF filter [2], particle filter [3], and optimization-based [4], [5] solutions, all relying on a drift-less planar motion model derived from a dual-drive or Ackermann steering platform. They perform relatively high-frequent integration of wheel odometry to come up with adequate priors on the relative displacement between subsequent views. Censi et al. [6] furthermore consider simultaneous extrinsic calibration between cameras and odometers. A closely related vehicle motion model that has also been used in filtering and optimization-based frameworks appears for skid-steering platforms [7], [8], [9]. Although slippage occurs, non-holonomic models relying on the Instantaneous Centre of Rotation (ICR) still explain the motion of skid-steering platforms relatively well [10], which is why our work may also be applied to such platforms. A very closely related work to ours is given by Zhang et al. [11], who still rely on a drift-less non-holonomic motion model, but extend the estimation to non-planar environments by introducing the motion-manifold and manifold-based integration of wheel odometry signals.

For pure vision-based solutions, the non-holonomic constraints need to be enforced purely by the model, which is more difficult. Scaramuzza [12], [13] successfully introduced the Ackermann motion model into relative camera displacement estimation, thus leading to highly robust solutions based on 1-point RANSAC or 1D histogram voting. Huang et al. [14] recently extended the method to an n-frame solver, while

[1]ShanghaiTech University; L. Kneip is also with the Shanghai Engineering Research Center of Intelligent Vision and Imaging. [2]Australian National University; The authors would like to thank the funding sponsored by Natural Science Foundation of Shanghai (grant number: 19ZR1434000).

Lee et al. [15] successfully applied it to a multi-camera array. Long et al. [16] and Li et al. [17] have included similar constraints into windowed optimization frameworks, which essentially penalize trajectory deviations from an approximate piece-wise circular arc model.

From a purely geometric point of view, a drift-less, non-holonomic ground vehicle moves along smooth trajectories in space, and—more importantly—heads toward the vehicle motion direction. This motivates our use of the continuous-time trajectory model as proposed by Furgale et al. [1]. While parametrizing a smooth vehicle trajectory, the representation and in particular its first-order differential is easily used to additionally enforce the vehicle heading to remain tangential to the trajectory.

III. Preliminaries

Continuous-time parametrizations have shown great value in motion estimation when dealing with smooth trajectories or temporally dense sampling sensors. There are various alternatives for the basis functions, such as FFTs, discrete cosine transforms, polynomial kernels, or Bézier splines. In this paper, we will use the efficient and smooth B-spline parametrization [18] as already illustrated by Furgale et al. [1]. We start by reviewing B-splines and conclude the section by looking at preliminaries on the non-holonomic motion.

A. B-splines

We represent the smooth motion with a p-th degree B-spline curve

$$\mathbf{c}(u) = \sum_{i=0}^{n} N_{i,p}(u)\mathbf{p}_i, \qquad a \leq u \leq b, \tag{1}$$

where u is the continuous-time parameter, $\{\mathbf{p}_i\}$ are the $n+1$ control points that control the smooth trajectory shape, and $\{N_{i,p}(u)\}$ are the $n+1$ pth-degree B-spline basis functions. Note that the form of a B-spline and the basis functions are generally fixed, the shape of the curve is influenced by the control points only. Trajectory splines are initialized from a set of discrete vehicle poses for each image and the image time-stamps. We use the spline curve approximation algorithm presented in Piegl and Tiller [18] for the initialization. The reader is invited to see more detailed foundations of B-splines and an example application in [18] and [1].

B. Non-holonomic motion

Ground vehicles commonly have a non-steering two-wheel axis, which causes the motion to be non-holonomic. This kinematic constraint is reflected in the Ackermann steering model. Infinitesimal motion is a rotation about an Instantaneous Centre of Rotation (ICR) which lies on the extended non-steering two-wheel axis. In other words, the instantaneous heading of the vehicle is parallel to its velocity. The constraint has already been exploited in purely vision-based algorithms, however only based on the approximation of a piece-wise constant steering angle:

- Front-end: Scaramuzza et al. [12] approximate the motion to be on a plane and the platform to have a locally constant steering angle. The trajectory between subsequent views is hence approximated by an arc of a circle, and the heading remains tangential to this arc. A minimal parameterization of the motion is given by the inscribed arc-angle θ as well as the radius of this circle r, and both the relative rotation and translation are expressed as functions of these parameters. The matter is illustrated in Figure 1.

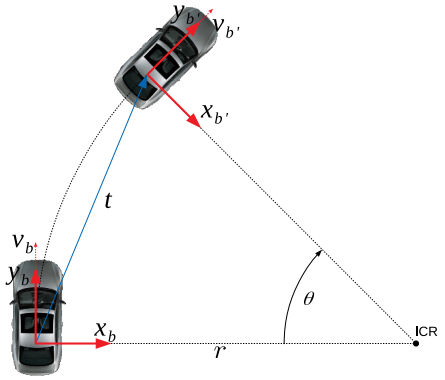


Fig. 1. Kinematic constraints of the Ackermann steering model.

- Back-end: As proposed by Peng et al. [19], relative rotations $\mathbf{R}$ and translations $\mathbf{t}$ under the approximation of a piece-wise constant steering angle or circular arc model need to satisfy the constraint

$$\left((\mathbf{I}+\mathbf{R})\begin{bmatrix}0 & 1 & 0\end{bmatrix}^T\right) \times \mathbf{t} = 0, \tag{2}$$

which can be added as a regularization term in a common bundle adjustment framework. We call this the *R-t constraint*.

The above models are only an approximation of the original infinitesimal constraints on the velocity and the position of the ICR. In the continuation, we will introduce the use of continuous-time parametrizations to continuously enforce identity between the body's velocity direction $\frac{\mathbf{v}_b}{\|\mathbf{v}_b\|}$ and the vehicle's forward axis y_b, which is the original infinitesimal constraint. We call this the *R-v constraint*.

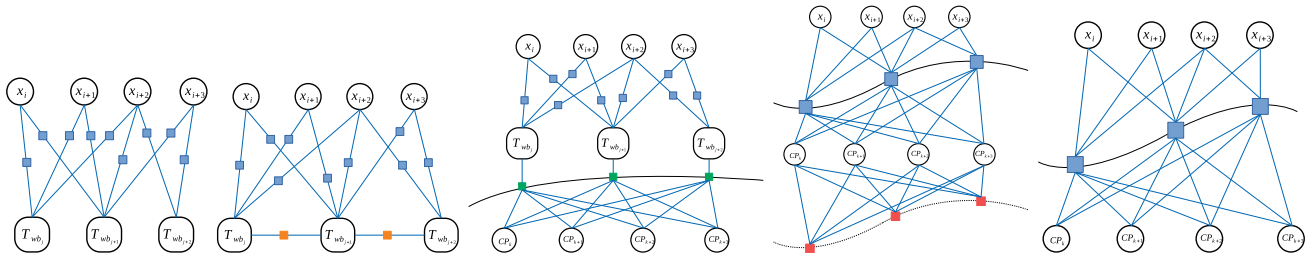
IV. Optimization of non-holonomic trajectories

We enforce the R-v constraint by using a spline to represent the non-holonomic vehicle trajectory in continuous time. The first-order differential of the spline gives us the instantaneous velocity of the vehicle, which we can then use to either directly express the vehicle heading (i.e. as a hard constraint), or otherwise form a regularization term on the vehicle orientation (i.e. as a soft constraint). Imposing the kinematic constraints as a hard or soft constraint may impact on both accuracy and computational performance, which is why we introduce and compare multiple formulations starting from conventional bundle adjustment.

A. Conventional Bundle Adjustment (CBA)

Conventional Bundle Adjustment (CBA) consists of minimizing reprojection errors over directly parametrized poses and landmarks. The non-linear objective is given by

$$\min_{\substack{\{\mathbf{t}_{b_j}\}\\ \{\mathbf{q}_{b_j}\}\\ \{\mathbf{x}_i\}}} \underbrace{\sum_{i,j} \rho\left(\left\|f_p\left(\mathbf{T}_{sb}\begin{bmatrix}\mathbf{R}(\mathbf{q}_{b_j}) & \mathbf{t}_{b_j}\\ 0 & 1\end{bmatrix}^{-1}\mathbf{x}_i\right) - \mathbf{m}_{ij}\right\|^2\right)}_{\text{conventional bundle adjustment (CBA)}}, \tag{3}$$



(a) (b) (c) (d) (e)

Fig. 2. Graphical models of the different methods: (a) CBA; (b) CBARt; (c) CBASpRv; (d) SSBARv; (e) FSBA.

where $\mathbf{R}(\mathbf{q})$ is the rotation matrix constructed from quaternion $\mathbf{q}$; $\mathbf{t}_{b_j}$ and $\mathbf{q}_{b_j}$ are the optimised pose parameters; $\{\mathbf{x}_i\}$ the landmarks (in homogeneous representation); $\mathbf{m}_{ij}$ the measurement of landmark i in frame j; $\rho(\cdot)$ is a loss function (e.g. Huber loss) to mitigate the influence of outliers; and $f_p(\cdot)$ is a general camera measurement function that depends on intrinsic parameters and transforms points from the camera frame (in homogeneous form) to the image plane. Note that $\mathbf{T}_{sb}$ represents the extrinsic parameters that transform points from the vehicle to the camera frame.

The graphical model of this problem is shown in Figure 2(a). Blue nodes are re-projection error residual blocks.

B. CBA with R-t constraints (CBARt)

The traditional way consists of adding the R-t constraint (2) as a pairwise, soft regularization constraint to bundle adjustment (cf. [16] and [17]). We obtain

$$\min_{\substack{\{\mathbf{t}_{b_j}\}\\ \{\mathbf{q}_{b_j}\}\\ \{\mathbf{x}_i\}}} \underbrace{\sum_{i,j} \rho\left(\left\|f_p\left(\mathbf{T}_{sb}\begin{bmatrix}\mathbf{R}(\mathbf{q}_{b_j}) & \mathbf{t}_{b_j}\\ 0 & 1\end{bmatrix}^{-1}\mathbf{x}_i\right)-\mathbf{m}_{ij}\right\|^2\right)}_{\text{conventional bundle adjustment (CBA)}} + \underbrace{\sum_j w_r\left\|\left(\left(\mathbf{I}+\mathbf{R}_{b_{j-1}b_j}\right)\begin{bmatrix}0\\1\\0\end{bmatrix}\right)\times\mathbf{t}_{b_{j-1}b_j}\right\|^2}_{\text{R-t constraint}}, \tag{4}$$

where w_r is a scalar weight for the R-t constraints, and the latter depend on the relative rotation $\mathbf{R}_{b_{j-1}b_j} = \mathbf{R}(\mathbf{q}_{b_{j-1}})^T\mathbf{R}(\mathbf{q}_{b_j})$ and the relative translation $\mathbf{t}_{b_{j-1}b_j} = \mathbf{R}(\mathbf{q}_{b_{j-1}})^T(\mathbf{t}_{b_j}-\mathbf{t}_{b_{j-1}})$ between subsequent views. The objective still optimises a discrete set of poses, and regularizes it against a piece-wise circular arc model.

The graphical model of this problem is shown in Figure 2(b). Yellow nodes indicate the R-t constraints.

C. CBA with spline regression (CBASpRv)

We proceed to our first utilization of a continuous time model, where we still use CBA to optimise individual poses, but interleavingly regress a 3D spline to the optimised positions $\mathbf{t}_b$ which we then use in a soft, regularising R-v constraint that replaces the R-t constraint. Denoting the alternatingly updated spline by $\mathbf{c}_1(t)$, the objective now becomes

$$\min_{\substack{\{\mathbf{t}_{b_j}\}\{\mathbf{q}_{b_j}\}\\ \{\mathbf{x}_i\},\mathcal{P}}} \underbrace{\sum_{i,j} \rho\left(\left\|f_p\left(\mathbf{T}_{sb}\begin{bmatrix}\mathbf{R}(\mathbf{q}_{b_j}) & \mathbf{t}_{b_j}\\ 0 & 1\end{bmatrix}^{-1}\mathbf{x}_i\right)-\mathbf{m}_{ij}\right\|^2\right)}_{\text{conventional bundle adjustment (CBA)}} + \underbrace{\sum_j w_s\|\mathbf{t}_{b_j}-\mathbf{c}_1(t_j)\|^2}_{\text{smoothness constraint}} + \underbrace{\sum_j w_c\|\mathbf{R}(\mathbf{q}_{b_j})\begin{bmatrix}0\\1\\0\end{bmatrix}-\eta(\mathbf{c}_1'(t_j))\|^2}_{\text{R-v constraint}}, \tag{5}$$

where $\mathcal{P}$ is the set of control points of $\mathbf{c}_1(t)$; t_j is the timestamp for jth frame; w_s, w_c are scalar weights; $\eta(\mathbf{a}) = \frac{\mathbf{a}}{\|\mathbf{a}\|}$; and $\mathbf{c}'(t)$ denotes the first-order derivative of $\mathbf{c}(t)$. Note that the latter is readily given as a spline that sums over products between control points and the fixed, first-order derivatives of the basis functions.

The graphical model of this problem is shown in Figure 2(c). The green nodes are the combined smoothness and R-v constraints. CP are the control points.

D. Soft spline bundle adjustment (SSBARv)

In our next formulation, the spline is used directly to represent the pose. Only the kinematic R-v constraint remains as side-constraint. We use the 7D spline $\mathbf{c}_2(t) = \begin{bmatrix}\mathbf{c}_2^{\mathbf{t}}(t)\\ \mathbf{c}_2^{\mathbf{q}}(t)\end{bmatrix}$ which represents the position in its first three entries and the quaternion orientation in its remaining four, see [20] for the detail about unit quternion B-spline. We obtain

$$\min_{\{\mathbf{x}_i\},\mathcal{P}} \underbrace{\sum_{i,j} \rho\left(\left\|f_p\left(\mathbf{T}_{sb}\begin{bmatrix}\mathbf{R}(\mathbf{c}_2^{\mathbf{q}}(t_j)) & \mathbf{c}_2^{\mathbf{t}}(t_j)\\ 0 & 1\end{bmatrix}^{-1}\mathbf{x}_i\right)-\mathbf{m}_{ij}\right\|^2\right)}_{\text{7D spline bundle adjustment(SSBA)}} + \underbrace{\sum_j w_c\|\mathbf{R}(\mathbf{c}_2^{\mathbf{q}}(t_j))\begin{bmatrix}0\\1\\0\end{bmatrix}-\eta(\mathbf{c}_2^{\mathbf{t}'}(t_j))\|^2}_{\text{R-v constraint}}, \tag{6}$$

where $\mathcal{P}$ is the set of control points of $\mathbf{c}_2(t)$. The graphical model of this problem is shown in Figure 2(d). The red nodes are the R-v constraints.

E. Hard spline bundle adjustment (FSBA)

Our final objective consists of directly using the derivative of the continuous position to express part of the vehicle orientation (i.e. the heading). As a result, we employ the four-dimensional spline $\mathbf{c}_3(t) = \begin{bmatrix}\mathbf{c}_3^{\mathbf{t}}(t)\\ \alpha(t)\end{bmatrix}$, where the first three entries denote the vehicle position as before, its derivative will be used to obtain the heading, and $\alpha(t)$ is a 1D spline that models the remaining roll angle about the heading

direction. Let's denote the vehicle orientation at time t as $\mathbf{U}(\mathbf{c}_3(t))$. It is defined as

$$\mathbf{U}(\mathbf{c}_3(t)) = \mathbf{Q}(\mathbf{c}_3^{\mathbf{t}}(t)) \begin{bmatrix} \cos\alpha(t) & 0 & \sin\alpha(t) \\ 0 & 1 & 0 \\ -\sin\alpha(t) & 0 & \cos\alpha(t) \end{bmatrix}, \quad (7)$$

where $\mathbf{Q}(\mathbf{c}_3^{\mathbf{t}}(t))$ is the base orientation and defined as

$$\mathbf{Q}(\mathbf{c}_3^{\mathbf{t}}(t)) = \begin{bmatrix} \eta\left(\eta(\mathbf{c}_3^{\mathbf{t}'}(t)) \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\right)^T \\ \eta(\mathbf{c}_3^{\mathbf{t}'}(t))^T \\ \left(\eta\left(\eta(\mathbf{c}_3^{\mathbf{t}'}(t)) \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\right) \times \eta(\mathbf{c}_3^{\mathbf{t}'}(t))\right)^T \end{bmatrix}^T. \quad (8)$$

The base orientation is defined such that the heading—the second column of $\mathbf{Q}(\mathbf{c}_3^{\mathbf{t}}(t))$—is aligned with the first-order differential of the vehicle trajectory, and the side-ways direction—the first column of $\mathbf{Q}(\mathbf{c}_3^{\mathbf{t}}(t))$—is orthogonal to the vertical direction $[0\ 0\ 1]^T$. The objective becomes

$$\min_{\{\mathbf{x}_i\},\mathcal{P}} \underbrace{\sum_{i,j} \rho\left(\left\|f_p\left(\mathbf{T}_{sb}\begin{bmatrix}\mathbf{U}(\mathbf{c}_3(t_j)) & \mathbf{c}_3^{\mathbf{t}}(t_j) \\ 0 & 1\end{bmatrix}^{-1}\mathbf{x}_i\right) - \mathbf{m}_{ij}\right\|^2\right)}_{\text{4D spline bundle adjustment(FSBA)}} \quad (9)$$

where $\mathcal{P}$ now denotes the set of control points of $\mathbf{c}_3(t)$. The graphical model of this problem is shown in Figure 2(e).

V. Experimental results

We mainly use the KITTI benchmark datasets [21], which are fully calibrated and contain images captured by a forward-looking camera mounted on a passenger vehicle driving through different environments. The datasets contain signals from high-end GPS/IMU sensors, which allow us to compare our results against ground truth. We use several different sequences that provide a mix of motion characteristics reaching from significant turns and height variations to simple forward motion. What's more, we add a few well-chosen, synthetic sequences to provide further analysis. We test all aforementioned methods plus CBASp and SSBA, which are similar to CBASpRv and SSBARv, respectively, but do not contain the kinematic R-v constraint. All our experiments are conducted on a laptop with 8GB RAM and an Intel Core i7 2.4 GHz CPU, and the C++ implementations use OpenCV [22], Eigen [23], and the Ceres [24] optimization toolbox with automatic differentiation.

The main purpose of our experiments is to demonstrate the ability of our method to handle degrading visual conditions. Besides the commonly analysed influence of noise on the image measurements $\mathbf{m}_{ij}$, we additionally analyse the influence of the connectivity of the graph by varying number of landmarks and number of observations. For each analysis and noise or connectivity setting, we calculate the mean and standard deviation of the sliding pair-wise Relative Pose Error (RPE) with respect to ground truth, which individually analyses rotation and translation errors. The rotation error is calculated using (2.15) in [25]. For the translation error, our evaluation differs from the one in [26] in that we ignore the scale of the relative translations which are unobservable in a monocular setting.

A. Results on synthetic data

We start by defining realistic trajectories, which we take straight from the ground truth trajectories from KITTI sequences [21]. We also adopt the intrinsic and extrinsic parameters $f_p(\cdot)$ and $\mathbf{T}_{sb}$ from the KITTI platform, respectively. However, rather than using the original image information, we generate synthetic correspondences by defining uniformly distributed random image points in each view. The number of points denotes the local connectivity. We define random depths for these points by sampling from a uniform distribution between 6 and 30 meters. The corresponding world points (landmarks) are finally projected into all nearby views to generate all possible correspondences in the graph. Note that the number of observations per landmark is however capped by the global connectivity setting. We also perform a boundary check to make sure that reprojected points are visible in the virtual views. Finally, we add zero-mean normally distributed noise to the observations.

Results are indicated in the first row of Figure 3. The detailed settings and resulting observations are as follows:

- **Noise level**: The noise level is controlled by setting the standard deviation of the normally distributed noise in unit pixels. As shown in Figures 3(a) and 3(b), adding kinematic constraints leads to a large reduction of errors; the proposed methods using continuous-time parametrizations perform better than CBA in most cases, especially in terms of the translational error. Although CBASpRv and CBARt are generally less stable, they perform best in low noise scenarios. FSBA and SSBARv in turn present high robustness against increasing noise levels.
- **Global connectivity**: As shown in Figures 3(c) and 3(d), the proposed kinematically consistent methods perform significantly better than their alternatives as the graph's global connectivity degrades. CBASpRv and CBARt again perform best for high connectivity, though also CBA is competitive at that setting.
- **Local connectivity**: As shown in Figures 3(e) and 3(f), the proposed kinematic methods perform significantly better as the number of observations per frame decreases, with SSBARv and FSBA outperforming other methods.

B. Results on artificial trajectories

The experiments of the previous section have indicated a reasonably good performance for CBARt. However, performance degrades as a function of another variable, which is the density of the frames (i.e. the R-t constraint is valid only locally). Given that the ICR is in fact a smoothly varying point, a lower frame density implies that the piece-wise circular arc based regularization of CBARt is less valid. To illustrate this matter, we conduct additional experiments in which the trajectory is formed by sinusoidal curves in the plane. In average, less than 10 keyframes are placed over the span of a single period, which simulates high dynamics. All other experimental settings are similar to the previous section, and results are shown in the second row of Fig. 3.

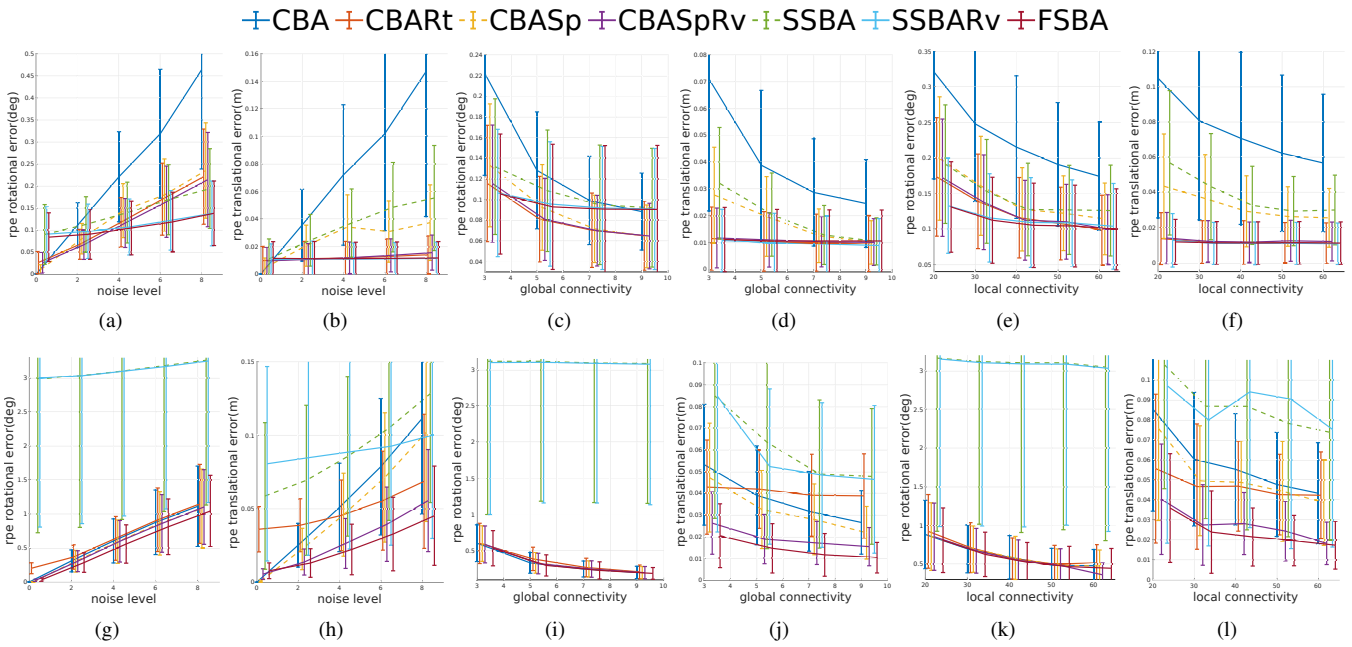


Fig. 3. Mean and standard deviation of RPE for different methods on synthetic data. The default noise level used in the experiments is 4, the default global connectivity is 3, and the default local connectivity is 40. The first row uses KITTI-VO-05 ground truth trajectories, while the second row summarizes results obtained on more sparsely sampled artificial trajectories. Columns one and two analyse rotational and translational errors for varying noise levels, columns 3 and 4 for varying maximum number of observations for each landmark, and columns 5 and 6 for varying number of landmarks per frame.

As expected, the performance of CBARt decreases and now performs similarly well to CBA and CBASp. Perhaps more surprisingly, SSBA and SSBARv are unable to handle this scenario well, and perform worst. An explanation is given by the fact that both SSBA and SSBARv use a 4D spline to model the orientation, and this representation seems unable to handle the fast orientation changes occurring in the present experiments. CBASpRv and FSBA in turn clearly outperform other methods.

C. Experiments on Real Data

On real images, we extract ORB [27] features and use the *flann* matcher from OpenCV. We furthermore use the 1-point RANSAC method by Scaramuzza et al. [13] to initialize the motion and identify inlier correspondences for triangulation, which lets us easily control the conditions for different experiments. The results for all methods are shown in Figure 4. We present individual results for six different KITTI sequences, which contain a mix of motion characteristics. KITTI-VO-01 and KITTI-VO-04 are an empty highway and a short straight road segment resulting in poor or simple graphical models, respectively. In order to measure the influence of degrading visual conditions, we again evaluate all results as a function of artificially added noise. The blue dotted line indicates the error of the initialization, which is generally improved upon after optimization. As can be observed, CBA generally performs worst, followed by CBASp, SSBA, and CBARt. The lowest errors are attained by SSBARv and FSBA, especially in terms of the translational error. Note that CBARt performs bad on KITTI-VO-00, which we trace back to a single difficult, badly modelled subpart of the trajectory between frames 250 and 300.

From a qualitative point of view, the advantage of our proposed methods is visualised in Figures 5(a), 5(b) and 5(c), which show the occasional failure of CBA to produce smooth results.

TABLE I

COMPARISON AGAINST ORB-SLAM. ERROR IN $\mathbf{t}$: [M] AND $\mathbf{R}$: [DEG].

Dataset	method	mean($\mathbf{t}$)	stddev($\mathbf{t}$)	mean($\mathbf{R}$)	stddev($\mathbf{R}$)
VO-01	ORB-SLAM	0.1293	0.1676	**0.3149**	0.4548
	CBA	0.0170	0.0413	0.3580	0.5445
	CBASpRv	0.0082	0.0046	0.3929	0.4717
	SSBARv	**0.0078**	**0.0033**	0.3606	**0.3684**
	FSBA	0.0080	0.0035	0.3711	0.4131
VO-04	ORB-SLAM	0.0073	0.0034	**0.0451**	**0.0312**
	CBA	0.0079	0.0039	0.0775	0.0392
	CBASpRv	**0.0050**	0.0032	0.0784	0.0392
	SSBARv	0.0050	**0.0032**	0.0806	0.0419
	FSBA	0.0051	0.0032	0.0829	0.0435
VO-06	ORB-SLAM	0.0076	0.0074	**0.0432**	**0.0277**
	CBA	0.0145	0.0411	0.1039	0.2494
	CBASpRv	**0.0057**	**0.0065**	0.0951	0.0821
	SSBARv	0.0057	0.0066	0.1013	0.0779
	FSBA	0.0058	0.0068	0.1074	0.0791

D. Comparison against ORB-SLAM

To conclude, we let our kinematically constrained optimization compete against an established alternative from the open-source community: ORB-SLAM [28]. RPE results are again indicated in Table I.

The results confirm that simple CBA is not able to compete with ORB-SLAM, while methods that impose kinematic constraints return comparable results and occasionally even outperform ORB-SLAM. We would like to emphasise that—although ORB-SLAM also uses CBA in the back-end—it is a heavily engineered framework that performs additional tasks to reinforce the health and quality of the underlying

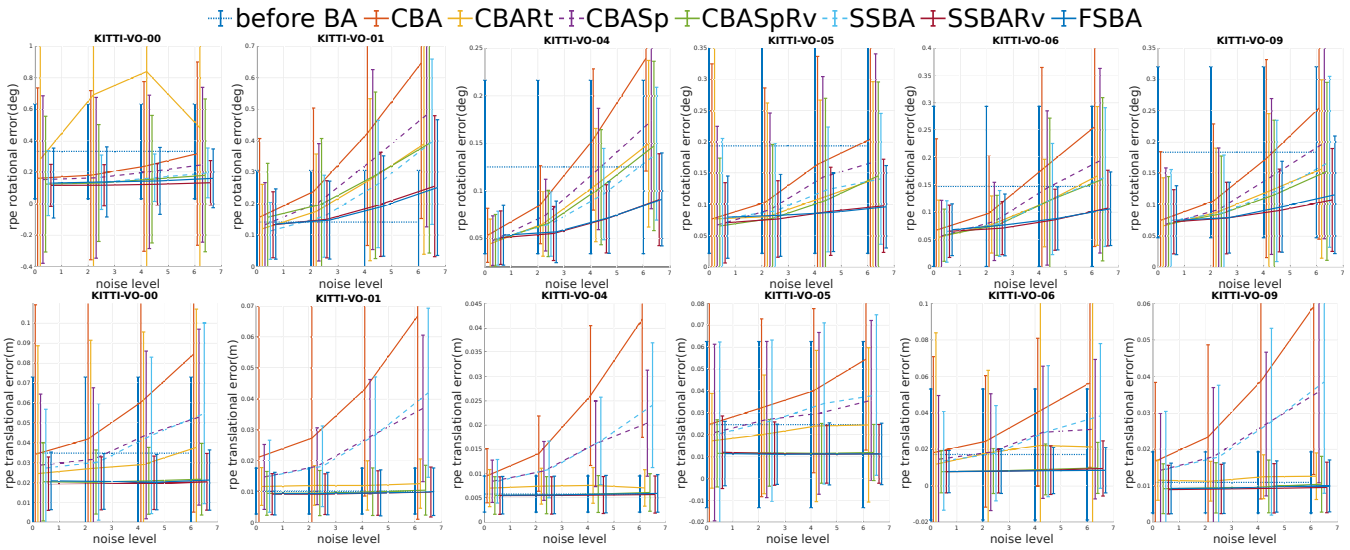


Fig. 4. Mean and standard deviation of RPE for different methods on real images from the KITTI benchmark. The first row shows rotational errors, while the second row shows translation errors. Each column presents results on a different dataset. *before BA* (blue dotted line) denotes the initial error before optimization.

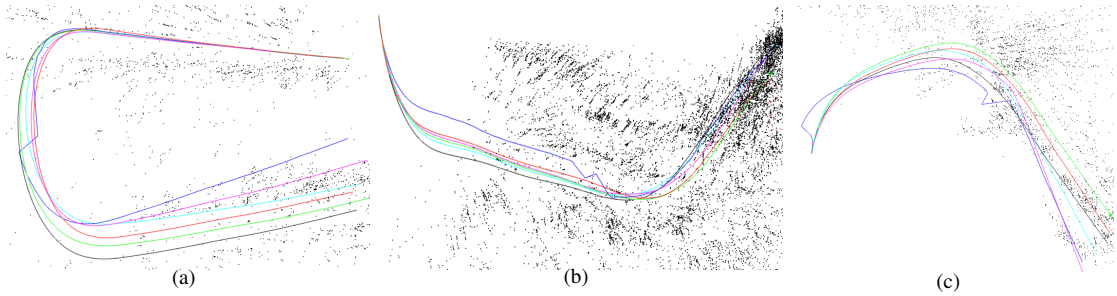


Fig. 5. Example trajectory segments for ground truth (Red), CBA (Blue), CBARt (Magenta), CBASpRv (Cyan), SSBARv (Black), and FSBA (Green). Left: U-turn on KITTI VO-06. Center: Uneven surface on KITTI VO-00. Right: Sharp turn on KITTI VO-10.

graphical model, while our methods simply use the map built from 1pt method [13]. We therefore again conclude that the addition of kinematic constraints generally models the motion well, and increases the ability to handle degrading visual measurements.

TABLE II
AVERAGE OPTIMIZATION TIME IN SECONDS PER 50 ITERATIONS.
(L.C. DENOTES LOCAL CONNECTIVITY).

L.C.	CBA	CBARt	CBASpRv	SSBARv	FSBA
20	9.294	8.963	22.225	23.824	18.301
30	13.887	15.661	30.790	35.662	27.270
40	18.162	19.775	57.784	46.733	31.150
50	22.194	22.555	62.120	56.652	43.387

E. Computational efficiency

To conclude, we compare the computational efficiency of the different methods. As indicated in Table II, the optimization in simulation is over 1000 frames using the KITTI-VO-05 trajectory. *l.c.* denotes the local connectivity, and thus the number of generated landmarks per frame. The noise level is set to 4, and the global connectivity to 3. There is no too significant difference between the various methods, with automatically differentiated B-spline-based implementations taking about double the time of conventional bundle adjustment. The fastest spline-based alternative is FSBA.

The number of control points used in the paper is about a third of the actual poses. We tested the effect of a varying number of control points. The most important insight is that—against the intuition—the computation time is not too much influenced by the number of control points, as the number of connections in the optimization graph is in fact left unchanged. The number of control points mainly affects the fitness of the spline.

VI. DISCUSSION

We introduce continuous-time trajectory parametrizations for an exact modelling of non-holonomic ground vehicle trajectories in bundle adjustment. Its addition strongly improves accuracy and robustness of monocular visual odometry, especially as the connectivity of the graph or the quality of the measurements degrades. For graphs with low connectivity, the hard-constrained four-dimensional spline formulation (FSBA) leads to the most stable and accurate results. For graphs with better connectivity, the best results are obtained by adding alternating spline regression and the R-v constraint to the optimization (CBASpRv). A possible explanation may be given by the fact that this representation still permits local pitch angle variations resulting from slight uneveness of the ground surface, and it may be less sensitive to errors in the extrinsic calibration parameters $\mathbf{T}_{sb}$. As for their sensitivity, we have $FSBA > SSBARv = CBASpRv$. FSBA is most sensitive due to the hard nature of the constraint.

REFERENCES

[1] P. Furgale, C. H. Tong, T. D. Barfoot, and G. Sibley, "Continuous-time batch trajectory estimation using temporal basis functions," *The International Journal of Robotics Research*, vol. 34, no. 14, pp. 1688–1710, 2015.

[2] K. J. Wu, C. X. Guo, G. Georgiou, and S. I. Roumeliotis, "VINS on wheels," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 5155–5162.

[3] T. Yap, M. Li, A. I. Mourikis, and C. R. Shelton, "A particle filter for monocular vision aided odometry," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5663–5669.

[4] M. Quan, S. Piao, M. Tan, and S.-S. Huang, "Tightly-coupled Monocular Visual-odometric SLAM using Wheels and a MEMS Gyroscope," *arXiv*, vol. 1804.04854, 2018.

[5] R. Kang, L. Xiong, M. Xu, J. Zhao, and P. Zhang, "Vins-vehicle: A tightly-coupled vehicle dynamics extension to visual-inertial state estimator," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 3593–3600.

[6] A. Censi, A. Franchi, L. Marchionni, and G. Oriolo, "Simultaneous calibration of odometry and sensor parameters for mobile robots," *IEEE Transactions on Robotics (T-RO)*, vol. 29, no. 2, pp. 475–492, 2013.

[7] J. Yi, H. Wang, J. Zhang, D. Song, S. Jayasuriya, and J. Liu, "Kinematic modelling and analysis of skid-steered mobile robots with applications to low-cost inertial-measurement unit-based motion estimation," *IEEE Transactions on Robotics (T-RO)*, vol. 25, no. 5, 2009.

[8] J. L. Martinez, J. Morales, A. Mandow, S. Pedraza, and A. Garcia-Cerezo, "Inertia-based ICR kinematic model for tracked skid-steer robots," in *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2017, pp. 166–171.

[9] W. Lv, Y. Kang, and J. Qin, "Indoor localization for skid-steering mobile robot by fusing encoder, gyroscope, and magnetometer," *IEEE Transactions on Systems, Man, and Cybernetics (SMC)*, vol. 99, pp. 1–13, 2017.

[10] J. L. Martinez, A. Mandow, J. Morales, S. Pedraza, and A. Garcia-Cerezo, "Approximating kinematics for tracked mobile robots," *International Journal of Robotics Research (IJRR)*, vol. 24, no. 10, pp. 867–878, 2005.

[11] M. Zhang, X. Zuo, Y. Chen, and M. Li, "Localization for ground robots: On manifold representation, integration, re-parameterization, and optimization," *arXiv*, vol. 1909.03423, 2019.

[12] D. Scaramuzza, F. Fraundorfer, and R. Siegwart, "Real-time monocular visual odometry for on-road vehicles with 1-point ransac," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. Ieee, 2009, pp. 4293–4299.

[13] D. Scaramuzza, "1-point-ransac structure from motion for vehicle-mounted cameras by exploiting non-holonomic constraints," *International Journal of Computer Vision (IJCV)*, vol. 95, no. 1, pp. 74–85, 2011.

[14] K. Huang, Y. Wang, and L. Kneip, "Motion estimation of non-holonomic ground vehicles from a single feature correspondence measured over n views," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, USA, 2019.

[15] G. H. Lee, F. Faundorfer, and M. Pollefeys, "Motion estimation for self-driving cars with a generalized camera," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 2746–2753.

[16] W. Zong, L. Chen, C. Zhang, Z. Wang, and Q. Chen, "Vehicle model based visual-tag monocular ORB-SLAM," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2017, pp. 1441–1446.

[17] P. Li, T. Qin, *et al.*, "Stereo vision-based semantic 3d object and ego-motion tracking for autonomous driving," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 646–661.

[18] L. Piegl and W. Tiller, *The NURBS book*. Springer Science & Business Media, 2012.

[19] X. Peng, J. Cui, L. Kneip, *et al.*, "Articulated multi-perspective cameras and their application to truck motion estimation." Institute of Electrical and Electronics Engineers Inc., 2019.

[20] I. Kang and F. Park, "Cubic spline algorithms for orientation interpolation," *International Journal for Numerical Methods in Engineering*, vol. 46, no. 1, pp. 45–64, 1999.

[21] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[22] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[23] G. Guennebaud, B. Jacob, *et al.*, "Eigen v3," http://eigen.tuxfamily.org, 2010.

[24] S. Agarwal, K. Mierle, and Others, "Ceres solver," http://ceres-solver.org.

[25] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An invitation to 3-d vision: from images to geometric models*. Springer Science & Business Media, 2012, vol. 26.

[26] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[27] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*. Ieee, 2011, pp. 2564–2571.

[28] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.