Visual-based Lifelong Kinematics and Pose Estimation for Skid-Steering Robots

Xingxing Zuo¹, Mingming Zhang², Yiming Chen², Guoquan Huang³, Yong Liu¹*, and Mingyang Li²*

Abstract—To build commercial robots, skid-steering mechanical design is of increased popularity due to its manufacturing simplicity and unique mechanism. However, these also cause significant challenges on software and algorithm design, especially for pose estimation (i.e., determining the robot's rotation and position), which is the prerequisite of autonomous navigation. While the general localization algorithms have been extensively studied in research communities, there are still fundamental problems that need to be resolved for localizing skid-steering robots that change their orientation with a skid. To tackle this problem, we propose a probabilistic sliding-window estimator dedicated to skid-steering robots, using measurements from a monocular camera, the wheel encoders, and optionally an inertial measurement unit (IMU). Specifically, we explicitly model the kinematics of skid-steering robots by both track instantaneous centers of rotation (ICRs) and correction factors, which are capable of compensating for the complexity of track-to-terrain interaction, the imperfectness of mechanical design, terrain conditions and smoothness, and so on. To prevent performance reduction in robots’ lifelong missions, the time- and location-varying kinematic parameters are estimated online along with pose estimation states in a tightly-coupled manner. More importantly, we conduct in-depth observability analysis for different sensors and design configurations in this paper, which provides us with theoretical tools in making the correct choice when building real commercial robots. In our experiments, we validate the proposed method by both simulation tests and real-world experiments, which demonstrate that our method outperforms competing methods by wide margins.

Index Terms—Kinematics, pose estimation, visual odometry, inertial-aided navigation, Skid-steering robots, observability.

I. INTRODUCTION

In recent years, the robotic community has witnessed a growing ‘go-to-market’ trend, by not only building autonomous robots for scientific laboratory usage but also making commercial robots to create new business model and facilitate people’s daily lives. To date, a large amount of commercial outdoor robots, under either daily business usage or active trial operations and tests, are customized skid-steering robots [2]–[4]. Instead of having an explicit mechanism of steering control, skid-steering robots rely on adjusting the speed of the left and right tracks to turn around. The simplicity of the mechanical design and the property of being able to turn around with zero-radius make skid-steering robots widely used in both the scientific research community as well as the commercial robotic industry. However, the mechanical simplicity of skid-steering robots has significantly challenged the software and algorithm design in robotic artificial intelligence, especially in autonomous localization.

The localization system provides motion estimates, which is a key component for enabling any autonomous robot. To localize skid-steering robots, there is a large body of relevant literature [5]–[12]. Early work by Anousaki et al. [5] showed that the standard differential-drive two-wheel vehicle model could not be used to accurately model the motion of a skid-steering robot due to track and wheel slippage. To address this problem, Martínez et al. [6] proposed an approach to approximate the kinematics of skid-steering robots based on instantaneous centers of rotation (ICRs). Although some other kinematic models of skid-steering robots are also proposed [10], [11], [13], [14], ICR based kinematics is still popular due to its simplicity and feasibility [8], [9], [12], especially for real-time robotic applications. In [8], IMU readings and wheel encoder measurements are fused in an EKF-based motion-estimation system for skid-steered robots. The ICR-based kinematics is utilized to compute virtual velocity measurements for robot motion estimation. However, the easily changed ICR parameters are not estimated in the estimator, which may lead to performance reduction. In [9], ICR parameters and navigation states are estimated online in an EKF-based estimator for 3-DoF motion estimation of skid-steering robots. Wheel odometer measurements and GPS measurements are fused in the system. The work [12] leveraged ICR-based kinematics and fused the readings from wheel encoders and a GPS-compass integrated sensor to estimate the ICR parameters and 3-DoF poses of the robot. Images are only used for terrain classification.

In contrast to the above existing works, we estimate the kinematics (formulated by ICRs and correction factors), and full 6-DoF poses (3-DoF rotations and 3-DoF translations) of the skid-steering robots jointly in a sliding-window bundle adjustment (BA) based estimator. Extracted visual features from camera and wheel encoder readings, optional IMU readings, are fused to optimize the estimated states in a tightly-coupled way and track the states of the skid-steering robot during its lifelong mission. In skid-steering robots, the track-to-terrain interaction is exceptionally complicated, and the conversion between wheel encoder readings and robot’s motion depends on mechanical design, wheel inflation conditions, load and center of mass, terrain conditions, slippage, and so on. In the lifelong mission of the skid-steering robots, such as delivery, the kinematic parameters can be inevitably changed. Thus we estimate the kinematic parameters online to guarantee accurate...
Fig. 1: The skid-steering robotic platform used in our tests, as well as the corresponding kinematic model. (a) Our testing robot, built on the Clearpath Jackal Platform [1]. The equipped low-cost sensors (i.e., a monocular camera, an imu, and wheel encoders) are leveraged in the proposed system, while the others (i.e., LiDAR and RTK-GPS) are not required in our system. (b) The odometer measurements and the instantaneous center of rotation (ICR, denoted by $[ICR_v, ICR_l, ICR_r]$) of a skid-steering robot. $O_v$ represents the robot velocity in odometer frame, and $O_\omega_z$ is the angular velocity along the yaw direction.

pose estimation of the robots in complicated environments without performance reduction.

For a complicated estimator, it is critical to conduct observability analysis [9], [15]–[19] to study the identifiability of estimated states. Pentzer et al. [9] investigated the conditions that ICR parameters will be updated in a GPS-aided localization system, by demonstrating that the ICR parameters can be only updated when the robot is turning. However, this is just a glimpse of the observability property. The nature of the GPS measurements and the applicability of that algorithm are fundamentally different from our visual-based systems. Our previous work [19] performed observability analysis of localizing steering skid robot by using a monocular camera, wheel encoders, and an IMU, and showed that the skid-steering parameters are generally observable. In this work, we extensively extend [19] and fully explore the observability properties of the visual-based kinematics and pose estimation system, by explicitly identifying the identifiable and non-identifiable parameters with and without using the IMU. To be specific, we provide in-depth analysis for the following common sensor configurations in commercial robots: i) using a monocular camera and wheel encoders for estimation, ii) incorporating an extra IMU into the sensor system, and iii) performing online sensor extrinsic calibration. The detailed estimator observability analysis is another key contribution of this work, which allows to correctly estimate the kinematics and poses of skid-steering robots.

In summary, we focus on pose and kinematics estimation of steering-skid robots to enable their lifelong mission in complicated environment, by using measurements from a monocular camera, wheel encoders, and optionally an IMU. The main contributions are as follows:

- An efficient visual-based estimator dedicated to skid-steering robots, which jointly estimates the ICR-based kinematic parameters of the robotic platform and 6-DoF poses in a tight-coupled manner. The formulation, error state propagation, and initialization of the kinematic parameters are presented in detail.
- Detailed observability analysis of the estimator under different sensor configurations, and the key results are as follows: (i) by using a monocular camera and wheel encoders, only the three ICR kinematic parameters are observable; (ii) by introducing the additional IMU measurements, both the three ICR kinematic parameters and the two correction factors are observable under general motion; and (iii) the 3D translation and one dimension of the rotation in the extrinsics between the camera and odometer are unobservable with the online estimate of kinematic parameters, which prevents performing online sensor-to-sensor extrinsic calibration.
- Extensive experiments including both simulation tests and real-world experiments were conducted for evaluations. Ablation study is also investigated to standout the feasibility of the proposed method. In general, the proposed method i) shows high accuracy and great robustness under different environmental and mechanical conditions to enable the lifelong mission of the robots and (ii) outperforms competing methods that do not or inappropriately estimate the kinematics.

The rest of the paper is organized as follows. We introduce the kinematics model of skid-steering robots in Sec. II. In Sec. III, the framework of the tightly-coupled sliding-window estimator is introduced, and we illustrate the kinematics estimation in detail. The observability analysis of the estimator under different configurations is performed in Sec. IV. Experimental results are presented in Sec. V. Finally, the paper is concluded in Sec. VI.
II. PRELIMINARIES ON ICR-BASED KINEMATICS OF SKID-STEERING ROBOTS

In this section, we present our kinematic model for skid-steering robots. In our derivation, we have assumed that the two wheels of the robots are always in contact with the ground surface. In other words, the case of a robot moving forward with one wheel hanging in the air is not allowed. In addition, the rotational rates of the wheels on each side of the robot are always the same, which is one of the most common mechanical design choices in skid-steering robots.

A. Notations

In this paper, we consider a robotic platform navigating with respect to a global reference frame, \( \{G\} \). The platform is equipped with a camera, an IMU, and wheel odometers, whose frames are denoted by \( \{C\}, \{I\}, \{O\} \) respectively. To present transformation, we use \( \mathcal{A}_{PB} \) and \( \mathcal{A}_{BR} \) to denote position and rotation of frame \( \{B\} \) with respect to \( \{A\} \), and \( \mathcal{A}_{Q} \) is the corresponding unit quaternion of \( \mathcal{A}_{BR} \). In addition, \( I \) denotes the identity matrix, and \( 0 \) denotes the zero matrix. We use \( \hat{x} \) and \( \delta x \) to represent the current estimated value and error state for variable \( x \). Additionally, we reserve the symbol \( \breve{x} \) to denote the inferred measurement value of \( x \), which is widely used in observability analysis. For the rotation matrix \( \mathcal{G}_{R} \), we define the attitude error angle vector \( \delta \theta \) as follows [20]:

\[
\mathcal{G}_{R} = \mathcal{G}_{\hat{R}} (I + [\delta \theta]_v)
\]

where \([v] \) denotes the skew-symmetric matrix of the vector \( v \):

\[
[v] = \begin{bmatrix}
0 & -v_3 & v_2 \\
v_3 & 0 & -v_1 \\
v_2 & v_1 & 0
\end{bmatrix}, \quad v = \begin{bmatrix}
v_1 \\
v_2 \\
v_3
\end{bmatrix}
\]

B. ICR-based Kinematics

In order to design a general algorithm to localize skid-steering robots under different conditions, the corresponding kinematic models must be presented in a parametric format. In this work, we employ a model similar to the ones in [6], [9], which contains five kinematic parameters: three ICR parameters and two correction factors, as shown in Fig. 1. To describe the details, we denote \( \text{ICR}_v = (X_v, Y_v) \) the ICR position of the robot frame, and \( \text{ICR}_l = (X_l, Y_l) \) and \( \text{ICR}_r = (X_r, Y_r) \) the ones of the left and right wheels, respectively. The relation between the readings of wheel encoder measurements and the ICR parameters can be derived as follows:

\[
Y_l = -\frac{\alpha_l - \mathcal{O}v_x}{\mathcal{O}w_z}, \quad Y_r = -\frac{\alpha_r - \mathcal{O}v_x}{\mathcal{O}w_z}
\]

\[
X_v = \frac{\mathcal{O}v_x}{\mathcal{O}w_z}, \quad X_l = X_r = -\frac{\mathcal{O}v_y}{\mathcal{O}w_z}
\]

where \( \alpha_l \) and \( \alpha_r \) are linear velocities of left and right wheels, \( \mathcal{O}v_x \) and \( \mathcal{O}v_y \) are robot’s linear velocity along \( x \) and \( y \) axes represented in frame \( O \) respectively, and \( \mathcal{O}w_z \) denotes the rotational rate about yaw also in frame \( O \). Those variables are also visualized in Fig. 1, and we use \( \xi_{ICR} = [X_v, Y_l, Y_r]^T \) to represent the set of ICR parameters. Moreover, we have used two scale factors, \( \xi_{\alpha} = [\alpha_l, \alpha_r]^T \), to compensate for effects which might cause changes in scales of wheel encoder readings. Representative situations include tire inflation, changes of road roughness, varying load of the robot, and so on. With the ICR parameters and correction factors being defined, the skid-steering kinematic model can be written as:

\[
\begin{bmatrix}
\mathcal{O}v_x \\
\mathcal{O}v_y \\
\mathcal{O}\omega_z
\end{bmatrix} = g(\xi, \alpha_l, \alpha_r) = \frac{1}{\Delta Y} \begin{bmatrix}
-Y_r & Y_l \\
X_v & -X_l \\
-1 & 1
\end{bmatrix} \begin{bmatrix}
\alpha_l & 0 \\
0 & \alpha_r \\
0 & \alpha_r
\end{bmatrix} \begin{bmatrix}
\alpha_l \\
0 \\
\alpha_r
\end{bmatrix}
\]

(4)

where \( \xi \) is the entire set of kinematic parameters.

Interestingly, as a special configuration when

\[
\xi = [0, 0.5b, -0.5b, 1, 1]^T
\]

(6)

with \( b \) being the distance between left and right wheels, Eq. 4 can be simplified as:

\[
\mathcal{O}v_x = \frac{\alpha_l + \alpha_r}{2}, \quad \mathcal{O}\omega_z = \frac{\alpha_r - \alpha_l}{b}, \quad \mathcal{O}v_y = 0
\]

This is exactly the kinematic model for a wheeled robot moving without slippage (i.e., an ideal differential drive robot), and used by most existing work for localizing wheeled robots [21–23]. However, in the case of skid-steering robots, if Eq. 7 is employed directly in a localizer, the pose estimation accuracy will be significantly reduced due to the incorrect conversion between wheel encoder readings and robot’s motion estimates (also see experimental results in Sec. V). We also note that since the kinematic parameter \( \xi \) represents the environmental and mechanical conditions of a moving robot, \( \xi \) can not be modelled as a constant parameter vector. Instead, to allow high-precision localization, \( \xi \) must be estimated online along with other localization states. This is conceptually similar to performing online sensor extrinsic calibration [17], [18], [24]. However, it is practically possible to calibrate sensors’ extrinsic parameter offline, while infeasible to estimate all skid-steering kinematic parameters before deployment.

III. VISUAL-INERTIAL KINEMATICS AND POSE ESTIMATION

In this paper, we utilize a sliding-window bundle adjustment based estimator for localizing skid-steering robots using a monocular camera, wheel encoders, and optionally an IMU. For presentation simplicity, in this section, we describe our estimator by explicitly considering using the IMU. When the IMU is not included in the sensor system, our presented estimator can be straightforwardly modified by simply deleting the IMU related components.

The architecture of our sliding-window estimator closely follows the design of [25], [26], by iteratively optimizing sensor measurement constraints and probabilistically marginalizing old information. We also note that, compared to the articles that focus on estimator architecture novelty, this work
is to describe methods to systematically handle skid-steering effects via online calibration. Our goal is to consistently and accurately estimate the motion of a moving robot as well as necessary observability-guided calibration parameters.

A. Estimator Formulation

1) State Vector: To start with, we define the state vector of our estimator as:

\[
\mathbf{x} = \begin{bmatrix} \mathbf{x}_O \top & \mathbf{G}_{\mathbf{V}_k} \top & \mathbf{b}_{a_k} \top & \mathbf{b}_{w_k} \top & \mathbf{m}_k \top & \mathbf{e}_k \end{bmatrix} \]

(8)

where

\[
\mathbf{x}_O = \begin{bmatrix} \mathbf{G}_{\mathbf{V}_k} \top \mathbf{q} \top & \mathbf{G}_{\mathbf{P}_k} \top \mathbf{r} \top & \mathbf{G}_{\mathbf{P}_k} \top \mathbf{v} \top & \mathbf{G}_{\mathbf{P}_k} \top \mathbf{g} \top \end{bmatrix} \top
\]

(9)

denotes the sliding-window poses of odometry frame at times \(\{k-1, \ldots, k\}\) when keyframe images are captured. \(\mathbf{G}_{\mathbf{V}_k}, \mathbf{b}_{a_k}, \mathbf{b}_{w_k}\) are the IMU related states, including the IMU velocity in global frame, acceleration bias, and gyroscope bias. If IMU is not available in the system, \(\mathbf{G}_{\mathbf{V}_k}, \mathbf{b}_{a_k}, \mathbf{b}_{w_k}\) will be excluded from the state vector. In addition, \(\mathbf{m}_k\) denotes the parameters for modeling the local motion manifold of the skid-steering robots across current sliding window. This has been shown in [26] to improve the estimation performance for ground robots, and we also adopt this design in our work. For completeness of the estimator presentation, we also provide the details on \(\mathbf{m}_k\) in Appendix VI-C. Finally, \(\mathbf{e}_k\), as shown in Eq. 5, represents the skid-steering intrinsic parameter vector, which is explicitly included in the state vector and thus estimated online.

2) Bundle Adjustment Optimization: Our optimization process closely follows the design of [25], [26]. Specifically, as illustrated in Fig. 2, the sliding-window BA in our estimation algorithm seeks to iteratively minimize a cost function corresponding to a combination of sensor measurement constraints, motion kinematic constraints, and marginalized constraints.

\[
\mathcal{C} = \mathcal{C}_P + \mathcal{C}_V + \mathcal{C}_I + \mathcal{C}_O + \mathcal{C}_M
\]

(10)

In what follows, we describe each of the cost terms. Firstly, the marginalized term \(\mathcal{C}_P\) is critical to consistently keep the algorithm computational complexity bounded, by probabilistically removing the old states in the sliding window. For a constraint \(\mathcal{C}(\mathbf{x}_r, \mathbf{x}_m)\) involved with the old states needed to be marginalized \(\mathbf{x}_m\) and the remaining states \(\mathbf{x}_r\), we compute the Hessian and gradient matrices with respect to \([\mathbf{x}_m, \mathbf{x}_r]^\top\), which are denoted as:

\[
\begin{bmatrix} \mathbf{A}_{rr} & \mathbf{A}_{rm} \\ \mathbf{A}_{mr} & \mathbf{A}_{mm} \end{bmatrix}, \quad \begin{bmatrix} \mathbf{g}_r \\ \mathbf{g}_m \end{bmatrix}
\]

(11)

The marginalization can be conducted by computing the marginalized Hessian and gradient matrices, i.e., \(\mathbf{A}_{marg} = \mathbf{A}_{rr} - \mathbf{A}_{rm} \mathbf{A}_{mm}^{-1} \mathbf{A}_{mr}\) and \(\mathbf{g}_{marg} = \mathbf{g}_r - \mathbf{A}_{rm} \mathbf{A}_{mm}^{-1} \mathbf{g}_m\), which represent the uncertainty information for the remaining states \(\mathbf{x}_r\) in the current sliding window [25]. Once marginalization is performed, the prior cost function can be formulated to ensure the remaining states are characterized by the computed uncertainties:

\[
\mathcal{C}_P(\mathbf{x}_r) = \frac{1}{2} \left\| \mathbf{x}_r - \mathbf{\hat{x}}_r \right\|_{\mathbf{A}_{marg}}^2 + \mathbf{g}_{marg}^\top (\mathbf{x}_r - \mathbf{\hat{x}}_r)
\]

(12)

The “boxminus” operator \(\boxminus\) denotes the generalized minus operation, since we need to perform computations on the manifold. For the marginalization, it should be noted that, as shown in Fig. 2, for limiting the computational complexity, we only leverage the constraints from IMU \(\mathcal{C}_I\) and odometer \(\mathcal{C}_O\) between the latest frame \(k\) and the second latest frame \(k-1\). After \(\mathcal{C}_I\) and \(\mathcal{C}_O\) are minimized in the optimization, the information contained in them and the related states will be marginalized into the prior cost term.

The camera term \(\mathcal{C}_P\), IMU term \(\mathcal{C}_I\), and motion manifold term \(\mathcal{C}_M\) used in this work are similar to that of existing literature [17], [25], [26] but with dedicated design for ground robots. In general, the camera cost term models the geometrical reprojection error of point features in the keyframes, the IMU term computes the error of IMU states between two consecutive keyframes, and the manifold cost term characterizes the motion smoothness across the whole sliding window. To have this article self-contained, the exact cost terms we use are provided in the Appendix VI. Finally, \(\mathcal{C}_O\) denotes the error induced by wheel odometer measurements. This term is a function of robot pose, measurement input, as well as skid-steering intrinsic parameters, and is discussed in detail in the next section.

It should also be noted that in this work, we assume that the IMU, the wheel odometers, and the camera are synchronized by hardware. Integration of IMU and odometer measurements between the time instants of captured images are required in the constraints \(\mathcal{C}_I\) and \(\mathcal{C}_O\). However, since different types of measurements come at varying frequencies, it is unlikely to get IMU/odometer measurements at the exact time instants when capturing the images. Thus, we perform the linear interpolations of IMU and odometer measurements at the image capturing time for performing integration.

B. ICR-based Kinematic Constraints

This section provides details on formulating \(\mathcal{C}_O\). Specifically, by assuming the supporting manifold of the robot is locally planar between \(t_k\) and \(t_{k+1}\), the local linear and angular velocities, \(\mathbf{O}(t)\mathbf{v}\) and \(\mathbf{O}(t)\mathbf{w}\), are a function of the wheel encoders’ measurements of the left and right wheels \(\mathbf{o}_l(t)\) and \(\mathbf{o}_r(t)\) as well as the skid-steering kinematic parameters \(\mathbf{\xi}\) [see (4)]:

\[
\begin{bmatrix} \mathbf{O}(t)\mathbf{v} \\ \mathbf{O}(t)\mathbf{w} \end{bmatrix} = \mathbf{\Pi} g(\mathbf{\xi}(t), \mathbf{o}_l(t), \mathbf{o}_r(t))
\]

\[
= \mathbf{\Pi} g(\mathbf{\xi}(t), \mathbf{o}_l(t) - n_l(t), \mathbf{o}_r(t) - n_r(t))
\]

(13)

where \(\mathbf{\Pi} = \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & 0 & 0 & 0 & \mathbf{e}_3 \end{bmatrix}^\top\) is the selection matrix with \(\mathbf{e}_i\) being a \(3 \times 1\) unit vector with the \(i\)th element of 1, \(n_l(t)\) and \(n_r(t)\) are the odometry noise modeled as zero-mean white Gaussian. With slight abuse of notation, we define \(\mathbf{n}_o = \begin{bmatrix} n_l & n_r \end{bmatrix}^\top\).
the odometer-induced kinematic constraint can be generically 
\[ \hat{t} \] parameters at the newest keyframe time \( t \) 
the differential equations in Eq. 14 over the time interval 
velocities of the robot (see Eq. 13) are available, we integrate 
by using a random walk process, and \( \xi \) parameters, e.g., 
so on. It is important to point out that, unlike sensor extrinsic 
changes in road conditions, tire pressures, center of mass, and 
n by zero-mean white Gaussian noise. The motivation of using 
keyframe. Once the instantaneous local ve-

\[ \text{Motion Manifold} \]

By using \( O(t) v \) and \( O(t) \omega \), the wheel odometry based 
kineic equations are given by:
\[ G_{PO}(t) = G_{O}(t) R \cdot O(t) v \] (14a)
\[ G_{O}(t) \dot{R} = G_{O}(t) R \cdot [O(t) \omega] \] (14b)
\[ \dot{\xi}(t) = n_\xi(t) \] (14c)

where we model the noise of the ICR kinematic parameter \( \xi \) 
by using a random walk process, and \( n_\xi \) is characterized 
by zero-mean white Gaussian noise. The motivation of using 
\( n_\xi \) is to capture time-varying characteristics of \( \xi \), caused 
by changes in road conditions, tire pressures, center of mass, 
and so on. It is important to point out that, unlike sensor extrinsic 
calibration in which parameters can be modeled as constant 
parameters, e.g., \( C_p \) = 0 in [27], \( \xi \) must be modeled as a 
time-varying variable.

To propagate pose estimates in a stochastic estimator, we 
describe the process starting from the estimates \( x_{O_k-1} = 
\begin{bmatrix}
G_p^{T} & G_{O_{k-1}}^{T} & G_{\dot{O}_{k-1}}^{T} & \dot{\xi}_{T}^{T}
\end{bmatrix}^{T} \). 
Once the instantaneous local ve-

\[ \text{Motion Manifold} \]

where \( \Lambda_O \) represents the inverse covariance (information) 

\[ \delta^O p_O \simeq \frac{G}{O^O} \left( I + [\delta \theta] \right) \left( O^O \dot{v} + J_{\omega}^O \delta \xi + J_{\nu}^O n_\nu \right) - \frac{G}{O^O} \dot{O} \dot{v} \] (15)
\[ \simeq -\frac{G}{O^O} \dot{O} \dot{v} \delta \theta + \frac{G}{O^O} \dot{O} \dot{v} J_{\nu}^O \delta \xi + \frac{G}{O^O} \dot{O} \dot{v} J_{\nu}^O n_\nu \] (16)
\[ \delta \dot{\theta} \simeq -[O^O \omega] \delta \theta + J_{\omega}^O \delta \xi + J_{\omega}^O n_\omega \] (17)
\[ \delta \xi \simeq n_\xi \] (18)

We here point out that Eq. 17 can be obtained similar to Eq. 
156 of [20]. In above equations, \( J_{\nu} \), \( J_{\omega} \), \( J_{\nu} \omega \) are the 
linearized Jacobian matrices, originated from:
\[ O^O v = O^O v + J_{\nu}^O \delta \xi + J_{\nu}^O n_\nu \] (19)
\[ O^O \omega = O^O \omega + J_{\omega}^O \delta \xi + J_{\omega}^O n_\omega \] (20)

and
\[ J_{\nu} = \frac{\hat{\alpha}_o \alpha - \hat{\alpha}_r \alpha_r}{\Delta Y^2} \begin{bmatrix} 0 & \dot{Y}_r & -\dot{Y}_l & 0 & 0 \\ \Delta Y & -X_r & X_l & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \frac{1}{\Delta Y} \begin{bmatrix} 0 & 0 & 0 & -\dot{Y}_r o_l & \dot{Y}_l o_r \\ 0 & 0 & 0 & \dot{X}_r o_l & -\dot{X}_l o_r \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \] (21)
\[ J_{\nu} \omega = -\frac{1}{\Delta Y} \begin{bmatrix} \hat{\alpha}_r \dot{Y}_r & \hat{\alpha}_l \dot{Y}_l \\ \dot{X}_r o_l & \dot{X}_l o_r \end{bmatrix} \] (22)
\[ J_{\omega} = \frac{1}{\Delta Y^2} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\dot{\alpha}_o \alpha_r + \alpha_1 o_l - \dot{\alpha}_r \alpha_r - \alpha_1 o_l - \Delta Y o_l & \Delta Y o_r \end{bmatrix} \] (23)
\[ J_{\omega} = \frac{1}{\Delta Y} \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -\dot{\alpha}_l \alpha_r \end{bmatrix} \] (24)

Once continuous-time error-state equations are given, the 
discrete-time Jacobian matrices, e.g., \( \frac{\partial f(v)}{\partial x} \) and \( \frac{\partial f(\omega)}{\partial x} \) in 
Eq. 15, can be straightforwardly calculated, similar to the 
process described in [20]. As a result, Eq. 15 encapsulates 
all information related to the skid-steering effect and enables 
online estimation of the skid-steering parameters. More details 
can also be found in our technical report [28]. After the 
constraint in Eq. 15 is minimized, \( \xi \) will be marginalized 
immediately, ensuring low computational complexity of the 
system.

C. Initialization of Kinematic Parameters

To allow the estimation of skid-steering kinematic parameters 
online, an initial estimate of the parameter vector \( \xi \) is 
required. Generally, in a stochastic estimator, the initial 
estimate can either be computed purely from sensory data or 
from prior knowledge. The first type of method is typically 
used for variables that are independent over trials (e.g., robot 
initial velocity [29]), and the other type is used for the states
that may vary relatively slowly (e.g., sensor extrinsics [24]). The skid-steering kinematic parameter belongs to the second case, and thus we propose a prior-based method.

Specifically, we use a simply while effective method by setting:

\[
\xi_{\text{initial}} = \begin{bmatrix} 0, 0.5 b^1, -0.5 b^1, 1, 1 \end{bmatrix}^T
\]  

(25)

We emphasize that Eq. 25 is similar but different from Eq. 6. The parameter \( b \) represents wheel distance in Eq. 6, which can be correctly used for robot without slippage. However, skid-steering robots are designed to have slippery behaviors, and thus \( b^1 \) should not be simply the wheel distance. To compute \( b^1 \), we rotate the skid-steering robots and use the fact that rotational velocity reported by the IMU and odometry should be identical, which leads to the following equation:

\[
b^1 = \frac{1}{N} \sum_{i=1}^{N} \frac{||g_{im}(t_i) - o_{im}(t_i)||}{||\omega_{im}(t_i)||}
\]  

(26)

where \( \omega_{im}(t_i) \) is gyroscope measurement, and \( N \) is the number of measurements. Although this is not of high precision and the road condition of computing \( b^1 \) is different from that of the testing time, this simple initialization method in combination of the proposed online calibration algorithm is able to yield accurate localization results (see our experimental results).

IV. OBSERVABILITY ANALYSIS

A critical prerequisite condition for a well-formulated estimator is to only include locally observable (or identifiable\(^1\)) [30] sensor and kinematic parameters (i.e., intrinsic and extrinsic parameters) in the online optimization stage. In the skid-steering robot localization system, a subset of estimation parameters inevitably become unobservable under center circumstances, which will be analytically characterized in this section and avoided in a real-world deployment.

Specifically, in this section, we first conduct our analysis by assuming the extrinsic parameters between sensors are perfectly known, and analyze the observability properties of the skid-steering parameters in different sensor system setup. Specifically, we consider three cases: (i) monocular camera and odometer with the 3 ICR parameters and 2 correction factors; (ii) monocular camera and odometer with 3 ICR parameter only; (iii) monocular camera, odometer and an IMU with 3 ICR parameters and 2 correction factors. Subsequently, we perform the analysis under the case that extrinsic parameters between sensors are unknown. Since estimating extrinsic parameters online is a common estimator design choice in robotics community [31]-[33], we also investigate the possibility of doing that for skid-steering robots.

A. Methodology Overview

To investigate the observability properties, the analysis can be either conducted in the original nonlinear continuous-time system [17] or the corresponding linearized discrete-time system [34], [35]. As shown in [16], [17], the dimension of the nullspace of the observability matrix might subject to changes due to linearization and discretization, and thus we conduct our analysis in the nonlinear continuous-time space in this work.

To conduct the observability analysis, we follow the methodology in [18], to examine the existence of indistinguishable trajectories given the kinematic and sensor measurement models. Specifically, the observability analysis consists of three main steps: firstly, we investigate the information provided by each sensor, and derive inferred ‘abstract’ measurements from the raw measurements; Secondly, we use kinematic and measurement constraints to derive equations that indistinguishable trajectories must follow; Finally, the observability matrix is constructed by computing the derivatives of the previous derived equations with respect to the states of interests. The observability of the states can be determined by examining the rank and nullspace of the observability matrix [36].

B. Inferred Measurement Model

We first analyze the information provided by a monocular camera. It is well-known that a monocular camera is able to provide information on rotation and up-to-scale position with respect to the initial camera frame [18], [37] under general motion. Equivalently, the information characterized by a monocular camera can be given by: (i) camera’s angular velocity and (ii) its up-to-scale linear velocity:

\[
\hat{\omega}_C(t) = C^{(t)}_C \omega + n_{\omega}(t)
\]  

(27a)

\[
\hat{v}_C(t) = s^{-1} \cdot C^{(t)}_C \cdot G \cdot v_C(t) + n_v(t)
\]  

(27b)

where \( n_{\omega}(t) \) and \( n_v(t) \) are the measurement noises, \( C^{(t)}_C \omega \) denotes true local angular velocity expressed in camera frame, and \( Gv_C(t) \) is the linear velocity of camera with respect to global frame, and finally \( s \) is an unknown scale factor. Additionally, \( \hat{\omega}_C(t) \) and \( \hat{v}_C(t) \) denote the inferred rotational and linear velocity measurements. Moreover, to make our later derivation simpler, we also introduce the rotated inferred measurements as follows:

\[
\hat{\tilde{\omega}}(t) \triangleq O\omega \hat{\omega}_C(t), \quad \hat{\tilde{v}}(t) \triangleq O\omega \hat{v}_C(t)
\]  

(28)

It is important to point out that in the cases when extrinsic parameter calibration between sensors is not considered in the online estimation stage, \( \hat{\omega}(t) \) and \( \hat{v}(t) \) can be uniquely computed from the camera measurement and also treated as the inferred measurement.

C. Observability of \( \xi \) with Monocular Camera and Odometer

We first investigate the case when a system is equipped with a monocular camera and odometers, and their extrinsic parameters are known in advance. To perform observability analysis, we derive system equations that indistinguishable trajectories must satisfy. To start with, we note that the following geometric relationships hold for any camera-odometer system:

\[
O\omega = O\omega \cdot C\omega
\]  

(29)

\( ^\top \) Since the derivative of \( \xi \) is modeled by zero-mean Gaussian, we here use observability and identifiability interchangeably.
which allows us to derive the following equations:

\[
\begin{align*}
G_p = \tilde{G}_p \cdot R \cdot C_p + G_p^C & \quad (30a) \\
G_v = \tilde{G}_v \cdot R \cdot C_v & \quad (30b) \\
G_RG_v = \tilde{G}_R \cdot R \cdot G_v & \quad (30c) \\
o_v = -[\tilde{\omega}]^T \cdot R \cdot C_v + \tilde{G}_R^T \cdot G_v & \quad (30d)
\end{align*}
\]

Substituting Eq. 28 to Eq. 30d, we obtain the following equation:

\[
o_v = -[\tilde{\omega}]^T \cdot R \cdot C_v + s \cdot \tilde{v} \quad (31)
\]

where \(o_v\) is known and \(o_v\) is velocity expressed in the odometer frame. We also note that, during the observability analysis, the noise terms are ignored, following the standard procedure of performing the observability analysis.

On the other hand, as mentioned in Sec. II, odometer provides observations for the speed of left and right wheels, i.e., \(o_l\) and \(o_r\) respectively. By linking \(o_l\), \(o_r\), \(\omega(t)\), \(\tilde{v}(t)\), and kinematic parameter vector \(\xi\) together, specifically substituting Eq. 31 into Eq. 4, we obtain:

\[
\begin{bmatrix}
\tilde{\omega}^O y_C \\
\tilde{\omega}^O x_C
\end{bmatrix}
+ \begin{bmatrix}
\tilde{v}_x \\
\tilde{v}_y
\end{bmatrix}
= \frac{1}{\Delta Y} \begin{bmatrix}
-Y_r X_v & Y_l \\
X_v & -X_v
\end{bmatrix} \begin{bmatrix}
o_l & 0 \\
0 & o_r
\end{bmatrix} + \frac{1}{\Delta Y} \begin{bmatrix}
\tilde{\omega} Y_l \\
\tilde{\omega} X_v
\end{bmatrix}
\]

(32)

where \(\tilde{\omega}^O x_C\), \(\tilde{\omega}^O y_C\) are the first and second element of \(o_v\), and \(\tilde{v}_x\), \(\tilde{v}_y\) are the first and second element of \(\tilde{v}\). For brevity, we use \(\tilde{\omega}\) to denote the third element of \(\omega\). By defining \(\beta_r = \Delta Y^{-1} \cdot o_r\), and \(\beta_l = \Delta Y^{-1} \cdot o_l\), we can write

\[
\begin{bmatrix}
\tilde{\omega}^O y_C \\
\tilde{\omega}^O x_C
\end{bmatrix}
+ \begin{bmatrix}
\tilde{v}_x \\
\tilde{v}_y
\end{bmatrix}
= \begin{bmatrix}
\tilde{\omega} Y_l + \beta_l Y o_l \\
-\tilde{\omega} X_v + \beta_r o_r
\end{bmatrix}
\]

(33)

Note that, this equation only contains 1) sensor measurements, and 2) a combination of vision scale factor and skid-steering kinematics:

\[
\epsilon = \begin{bmatrix} X_v & Y_l & Y_r & o_l & o_r & s \end{bmatrix}^T
\]

which allows us to analyze whether indistinguishable sets of \(\epsilon\) exist subject to the provided measurement constraint equations.

The identifiability of \(\epsilon\) can be described as follows:

**Lemma 1.** By using measurements from a monocular camera and wheel odometers, \(\epsilon\) is not locally identifiable.

**Proof.** \(\epsilon\) is locally identifiable if and only if \(\bar{\epsilon}\) is locally identifiable:

\[
\bar{\epsilon} = \begin{bmatrix} Y_l & \Delta Y & X_v & \beta_l & \beta_r & s \end{bmatrix}^T
\]

By expanding Eq. 33, we can write the following constraints:

\[
\begin{align*}
c_x(\epsilon, t) = \tilde{\omega}(t) & Y_C + s \tilde{v}_x(t) - \tilde{\omega}(t) Y_l - \beta_l \Delta Y o_l(t) = 0 \quad (34a) \\
c_y(\epsilon, t) = -\tilde{\omega}(t) & X_C + s \tilde{v}_y(t) + \tilde{\omega}(t) X_v = 0 \quad (34b) \\
c_w(\epsilon, t) = \tilde{\omega}(t) & + \beta_l o_l(t) - \beta_r o_r(t) = 0 \quad (34c)
\end{align*}
\]

A necessary and sufficient condition of \(\epsilon\) to be locally identifiable is following observability matrix has full column rank, over a set of time instants \(S = \{t_0, t_1, \ldots, t_s\}\):

\[
\mathcal{O}_c = \begin{bmatrix} D(t_0)^T & D(t_1)^T & \cdots & D(t_s)^T \end{bmatrix}^T
\]

where

\[
D(t) = \begin{bmatrix}
\frac{\partial c_x(\epsilon, t)}{\partial \epsilon} & \frac{\partial c_y(\epsilon, t)}{\partial \epsilon} & \frac{\partial c_w(\epsilon, t)}{\partial \epsilon}
\end{bmatrix}^T
\]

(35)

Substituting Eq. 36 back into Eq. 35 leads to:

\[
\mathcal{O}_c = \begin{bmatrix}
-\tilde{\omega}(t_0) - \beta_l o_l(t_0) & 0 & -\Delta Y o_l(t_0) & 0 & \tilde{v}_x(t_0) \\
0 & 0 & \tilde{\omega}(t_0) & 0 & 0 & 0 & \tilde{v}_y(t_0) \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
-\tilde{\omega}(t_s) - \beta_l o_l(t_s) & 0 & -\Delta Y o_l(t_s) & 0 & \tilde{v}_x(t_s) \\
0 & 0 & \tilde{\omega}(t_s) & 0 & 0 & 0 & \tilde{v}_y(t_s) \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & -\beta_r o_r(t_s) & 0
\end{bmatrix}
\]

(37)

By defining \(\mathcal{O}_c(\cdot, i)\) the ith block columns of \(\mathcal{O}_c\), the following equation holds:

\[
(\tilde{\omega}^O y_C + Y_l) \cdot \mathcal{O}_c(\cdot, 1) + \Delta Y \cdot \mathcal{O}_c(\cdot, 2) \\
+ (X_v - \tilde{\omega}^O x_C) \cdot \mathcal{O}_c(\cdot, 3) + s \cdot \mathcal{O}_c(\cdot, 6) = 0
\]

The above equation demonstrates that \(\mathcal{O}_c\) is not of full column rank, indicating that \(\epsilon\) is not identifiable.

To further investigate the indistinguishable states that cause the unobservable situations, we note that for a vector \(\bar{\epsilon}_1 = [Y_l, \Delta Y, X_v, \beta_l, \beta_r, s]^T\) that satisfies Eq. 34, another vector

\[
\bar{\epsilon}_2 = [(1 + \lambda/s) Y_l - (\lambda/s) \tilde{\omega}^O y_C, (1 + \lambda/s) \Delta Y, \\
(1 + \lambda/s) X_v - (\lambda/s) \tilde{\omega}^O x_C, \beta_l, \beta_r, s + \lambda]^T
\]

for any \(\lambda \in \mathbb{R}\) is always valid for the constraints Eq. 34. Thus \(\bar{\epsilon}_1\) and \(\bar{\epsilon}_2\) are indistinguishable, and \(\epsilon\) is not locally identifiable. This completes the proof.

\(\square\)

**D. Observability of \(\xi_{ICR}\) with Monocular Camera and Odometer**

Since the full kinematic parameters \(\xi = [\xi_{ICR}^T, \xi_{GA}^T]^T\) with monocular camera and odometer are not locally identifiable, we look into the case of that only the 3 ICR parameters, i.e., \(\xi_{ICR}\), are estimated without the correction factors. Similar to Eq. 32, the following equation holds:

\[
\begin{bmatrix}
\tilde{\omega}^O y_C \\
\tilde{\omega}^O x_C
\end{bmatrix}
+ \begin{bmatrix}
\tilde{v}_x \\
\tilde{v}_y
\end{bmatrix}
= \begin{bmatrix}
\tilde{\omega} Y_l \\
\tilde{\omega} X_v
\end{bmatrix}
+ \begin{bmatrix}
o_l \\
o_r
\end{bmatrix}
\]

(38)

This completes the proof.
The above expression is a function of the odometry and inferred visual measurements \( \tilde{\omega}, \tilde{v}_x, \tilde{v}_y, o_l, o_r \), as well as the kinematic intrinsic parameters \( \xi_{ICR} \) and visual scale factor \( s \):

\[
\gamma = [\xi_{ICR}^T \ s]^T = [X_v \ Y_l \ Y_r \ s]^T
\]

The local identifiability of \( \gamma \) can be stated as follows:

**Lemma 2.** By using the monocular and odometry measurements, and the 3 ICR parameter vector \( \xi_{ICR} \) to model the kinematics, \( \gamma \) is locally identifiable except for the following degenerate cases: (i) the odometry linear velocity \( o_l(t) \) keeps zero; (ii) the angular velocity \( \tilde{\omega}(t) \) keeps zero; (iii) \( o_l(t), o_r(t) \), and \( \tilde{\omega}(t) \) are all constants; (iv) the linear velocities of two wheels \( o_l(t), o_r(t) \) keeps identical to each other; (v) the angular velocity \( \tilde{\omega}(t) \) is consistently proportional to \( o_l(t) \).

**Proof.** We first note that the local identifiability of \( \gamma \) is equivalent to that of \( \gamma' \),

\[
\gamma' = [Y_l \ \Delta Y \ X_v \ s]^T
\]

By expanding (38) and considering all the measurements at different time, we can derive the following system constraints:

\[
\begin{align*}
 c_x(\gamma, t) &= \tilde{\omega}(t)O^C y_C + s\tilde{v}_x(t) - \tilde{\omega}Y_l - o_l(t) = 0 \quad (39a) \\
 c_y(\gamma, t) &= -\tilde{\omega}(t)O^C x_C + s\tilde{v}_y(t) + \tilde{\omega}(t)X_v = 0 \quad (39b) \\
 c_\omega(\gamma, t) &= \tilde{\omega}(t) + \frac{o_l(t) - o_r(t)}{\Delta Y} = 0 \quad (39c)
\end{align*}
\]

Similar to the case of using full kinematic parameters \( \xi \) in Section IV-C, we derive the following observability matrix for \( \gamma' \) (using Eq. 35 and 36):

\[
O_c = \begin{bmatrix}
-\tilde{\omega}(t_0) & 0 & 0 & \tilde{v}_x(t_0) \\
0 & 0 & \tilde{\omega}(t_0) & \tilde{v}_y(t_0) \\
\vdots & \vdots & \vdots & \vdots \\
-\tilde{\omega}(t_s) & 0 & 0 & \tilde{v}_x(t_s) \\
0 & 0 & \tilde{\omega}(t_s) & \tilde{v}_y(t_s) \\
\vdots & \vdots & \vdots & \vdots \\
0 & 0 & \frac{o_l(t_s) - o_r(t_s)}{\Delta Y} & 0 \\
\end{bmatrix}
\]

(40)

To simplify the structure of the observability matrix, we apply the following linear operations without changing the observability properties:

\[
\begin{align*}
O_c(:, 2) &\leftarrow (\Delta Y)^2O_c(:, 2) \\
O_c(:, 4) &\leftarrow sO_c(:, 4) + (Y_l^{-1}O^C y_C)O_c(:, 1) + (X_v^{-1}O^C x_C)O_c(:, 3)
\end{align*}
\]

where \( (\cdot) \leftarrow (\cdot) \) represents the operator to replace the left side by the right side. As a result, Eq. 40 can be simplified as:

\[
O_c = \begin{bmatrix}
-\tilde{\omega}(t_0) & 0 & 0 & o_l(t_0) \\
0 & 0 & \tilde{\omega}(t_0) & 0 \\
\vdots & \vdots & \vdots & \vdots \\
0 & 0 & \frac{o_l(t_s) - o_r(t_s)}{\Delta Y} & 0 \\
\end{bmatrix}
\]

(41)

To investigate the observability of the matrix in Eq. 41, we inspect the existence of the non-zero vector \( k \) such that \( O_ck = 0 \). If such a vector \( k \) exists, all of the following conditions must be satisfied:

\[
-\tilde{\omega}(t)k_1 + o_l(t)k_4 = 0, (o_r(t) - o_l(t))k_2 = 0, \tilde{\omega}(t)k_3 = 0
\]

To allow the above equations to be true, one of the following conditions is required:

- \( o_l(t) \) keeps constantly zero, \( k = [0 \ 0 \ \rho]^T \);
- \( \tilde{\omega}(t) \) keeps constantly zero, \( k = [0 \ \rho \ 0 \ 0]^T \);
- \( o_l(t), o_r(t), \tilde{\omega}(t) \) are all constants, \( k = [0 \ \rho \ 0 \ 0]^T \);
- \( \tilde{\omega}(t) \) keeps proportional to \( o_l(t), \ k = [\rho o_l/\tilde{\omega} \ 0 \ \rho \ 0]^T \).

where \( \rho \) can be any non-zero value that is used to generate valid non-zero vector \( k \) such that \( O_ck = 0 \). We note that, all above cases are special conditions. When a robot moves under general motion, none of those conditions can be satisfied. Therefore, in a camera and odometers only skid-steering robot localization system, \( \xi_{ICR} \) is observable unless entering the specified special conditions listed above.

**E. Observability of \( \xi \) with a Monocular Camera, an IMU, and Odometer**

So far, we have shown that when a robotic system is equipped with a monocular camera and wheel odometer, estimating \( \xi \) is not feasible, and the alternative solution is to include \( \xi_{ICR} \) in the online stage only. However, it is not an ideal solution to calibrate \( \xi_{ICR} \) offline and fixed in the online stage since it is subject to the changes in road conditions and tire conditions and so on.

To tackle this problem, we investigate the observability of \( \xi \) when an IMU is added to the camera-odometer system. Once an IMU is used, similar to the previous analysis, we start by introducing the ‘inferred measurement’. Instead of focusing on visual measurement only, we provide ‘inferred’ measurement by considering the visual-inertial system together. As analyzed in rich existing literature, visual-inertial estimation provides: camera’s local (i) angular velocity and (ii) linear velocity, similar to vision only case (Eq. 28) without having the unknown scale factor [16], [18], [38]. Similarly to Eq. 34a, to simplify the analysis, we prove identifiability of \( \xi \) instead of \( \xi \), since properties of \( \xi \) and \( \xi \) are interchangeable:

\[
\xi = [Y_l \ \Delta Y \ X_v \ \beta_l \ \beta_r]^T
\]

**Lemma 3.** By using measurements from a monocular camera, an IMU, and wheel odometer, \( \xi \) is locally identifiable, except for following degenerate cases: (i) velocity of one of the wheels, \( o_l(t) \) or \( o_r(t) \), keeps zero; (ii) \( \tilde{\omega}(t) \) keeps zero; (iii) \( o_l(t), o_r(t) \), and \( \tilde{\omega}(t) \) are all constants; (iv) \( o_l(t) \) is always proportional to \( o_r(t) \); (v) \( \tilde{\omega}(t) \) is always proportional to \( o_l(t) \).
Proof. Similarly to Eq. 34, by removing the scale factor, the corresponding system constraints can be derived as:

\[ c_x(\xi, t) = \ddot{\omega}(t) O_y^C Y + \ddot{v}_x(t) - \ddot{\omega}(t) Y_1 - \beta_1 \Delta Y o(t) = 0 \quad (42a) \]
\[ c_y(\xi, t) = -\ddot{\omega}(t) O_x^C + \ddot{v}_y(t) + \ddot{\omega}(t) X_v = 0 \quad (42b) \]
\[ c_w(\xi, t) = \ddot{\omega}(t) + \beta_2 o(t) - \beta_2 r_o(t) = 0 \quad (42c) \]

Therefore, the observability matrix for \( \bar{\xi} \) can be computed by:

\[
O_c = \begin{bmatrix}
-\ddot{\omega}(t_0) & -\beta_1 o(t_0) & 0 & -\Delta Y o(t_0) & 0 \\
0 & 0 & \ddot{\omega}(t_0) & 0 & 0 \\
0 & 0 & 0 & o(t_0) & -a_r(t_0) \\
\vdots & \ddots & \ddots & \ddots & \ddots \\
0 & 0 & 0 & \ddot{\omega}(t_s) & 0 & -\Delta Y o(t_s) & 0 \\
0 & 0 & 0 & o(t_s) & -a_r(t_s)
\end{bmatrix}
\]

After the following linear operations:

\[
O_c(\cdot, 2) \leftarrow -O_c(\cdot, 2)/\beta_1 \\
O_c(\cdot, 4) \leftarrow O_c(\cdot, 4) + \Delta Y O_c(\cdot, 2)
\]

Eq. 43 can be simplified as:

\[
O_c = \begin{bmatrix}
-\ddot{\omega}(t_0) & o(t_0) & 0 & 0 & 0 \\
0 & 0 & \ddot{\omega}(t_0) & 0 & 0 \\
0 & 0 & 0 & o(t_0) & -a_r(t_0) \\
\ddots & \ddots & \ddots & \ddots & \ddots \\
0 & 0 & \ddot{\omega}(t_s) & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & o(t_s) & -a_r(t_s)
\end{bmatrix}
\]

Similarly, we investigate the existence of the non-zero vector \( k \) such that \( O_c k = 0 \). If such a vector \( k = [k_1 \ k_2 \ k_3 \ k_4 \ k_5] \) exists, all of the following must be satisfied:

\[ -\ddot{\omega}(t_0) k_1 + o(t_0) k_2 = 0, \ddot{\omega}(t_0) k_3 = 0, o_r(t_0) k_4 = o(t_0) k_5 = 0 \]

which requires one of the following conditions to be true:

- \( o(t) \) is constantly zero, \( k = [0 \ \rho_1 \ 0 \ \rho_2] \), or \( o_r(t) \) is constantly zero, \( k = [0 \ 0 \ 0 \ \rho_1 \ 0 \ \rho_2] \),
- \( \ddot{\omega}(t) \) is constantly zero, \( k = [\rho_1 \ 0 \ \rho_2 \ 0 \ \rho] \),
- \( o_r(t), o(t), \) and \( \ddot{\omega}(t) \) are all constants, \( k = [0 \ 0 \ \rho \ 0 \ 0] \),
- \( o(t) \) keeps proportional to \( o_r(t) \), \( k = [0 \ 0 \ \rho \ 0 \ 0] \),
- \( \ddot{\omega}(t) \) keeps proportional to \( o(t) \), \( k = [\rho o(t) / \ddot{\omega} \ \rho \ 0 \ 0 \ 0] \).

where \( \rho, \rho_1, \rho_2 \) can be any non-zero value that is used to generate valid non-zero vector \( k \) such that \( O_c k = 0 \). All the above cases are special conditions. Therefore, in a skid-steering robot localization system equipped with a camera, an IMU, and odometers, \( \bar{\xi} \) is observable unless entering the specified special conditions listed above. This completes the proof.

F. Observability of \( \xi \) with a Monocular Camera, an IMU, an Odometer, and with Online Extrinsic Calibration

It is essential to know the extrinsic transformations between different sensors in a multi-sensor system. Since IMU-camera extrinsic parameters are widely investigated in the existing literature, and practically the IMU-camera system is frequently manufactured as an integrated sensor suite, we here focus on camera-odometer extrinsic parameters.

Since the camera system and wheels of a mobile robot are different hardware components, extrinsic calibration between the corresponding frames is essential for sensor fusion. In this section, we investigate the possibility of performing online extrinsic parameter calibration by including them into the state vector and estimating along with other variables of interests [18], [24]. To allow this algorithm to function properly, the corresponding extrinsic parameters must be observable, which we investigate as follows.

We first define the parameter state when camera-odometer extrinsics are included:

\[ \eta = [X_v \ Y_r \ \alpha_l \ \alpha_r \ \alpha_\xi \ \alpha_y \ \alpha_c \ \alpha_y^C \ \alpha_c^C \ \delta \theta] \]

where \( \alpha_\xi \) is the translational and rotational part of the extrinsic transformation between odometer and camera. \( \alpha_y^C \) is error state (or lie algebra increment) of the 3D rotation matrix \( \bar{C} \bar{R} \).

Since extrinsic translation and rotation components might be subject to different observability properties, we also define state parameters that contain each of them separately:

\[ \eta_p = [X_v \ Y_r \ \alpha_l \ \alpha_r \ \alpha_x \ \alpha_y \ \alpha_z \ \alpha_x^C \ \alpha_y^C \ \alpha_z^C] \]

and

\[ \eta_0 = [X_v \ Y_r \ \alpha_l \ \alpha_r \ \delta \theta] \]

To summarize, the objective of this section is to demonstrate the observability properties of \( \eta, \eta_p, \) and \( \eta_0 \).

Lemma 4. By using measurements from a monocular camera, IMU and wheel odometers, \( \eta_p \) and \( \eta \) are not identifiable. Specifically, the vertical direction of translation in the extrinsics, \( \alpha_z^C \) is always unidentifiable for any type of ground robot, and \( \alpha_x^C \) and \( \alpha_y^C \) become unidentifiable if the skid-steering kinematic parameters are estimated online.

Lemma 5. By using measurements from a monocular camera, IMU and wheel odometers, \( \eta_0 \) is identifiable except the third dimension of the rotation between camera and odometer.

Proof. First of all, \( \eta \) is locally identifiable if and only if \( \tilde{\eta} \) is locally identifiable:

\[ \tilde{\eta} = [Y_r \ \Delta Y \ X_v \ \beta_l \ \beta_r \ \alpha_x \ \alpha_y \ \alpha_z \ \alpha_x^C \ \alpha_y^C \ \alpha_z^C \ \delta \theta] \]

be observable unless entering the specified special conditions listed above. This completes the proof. □
By substituting \( \ddot{v}(t) = \frac{\partial}{\partial C} \mathbf{R} \cdot \dot{v}_C(t) \) in Eq. 28, we are able to derive constraints similar to Eq. 42, as

\[
\begin{align*}
c_x (\eta, t) &= \dot{w}(t) O_{yc} e_1^T C R \dot{v}_C(t) - \dot{w}Y_t - \beta_1 \Delta Y t = 0 \quad (45a) \\
c_y (\eta, t) &= -\dot{w}(t) O_{xc} e_2^T C R \dot{v}_C(t) + \dot{w}(t) X_t = 0 \quad (45b) \\
c_{\omega} (\zeta, t) &= \dot{w}(t) + \beta_2 t - \beta_1 \theta t = 0 \quad (45c)
\end{align*}
\]

Considering the constraints in a set of time instants \( S = \{t_0, t_1, \ldots, t_s\} \), we compute the following observability matrices for the systems with calibrating \( C \) and \( \eta \), given by Eq. 46 and Eq. 47a, respectively. Eq. 47a can be converted to Eq. 47b by linear operations:

\[
\begin{align*}
O_c(:, 4) &\leftarrow O_c(:, 4) - \Delta Y / \beta_1 O_c(:, 2) \\
O_c(:, 2) &\leftarrow -O_c(:, 2) / \beta_1
\end{align*}
\]

Similar to previous proofs, to look into the properties of \( O_c \) in Eq. 46, we investigate non-zero vector \( k \) such that \( O_c k = 0 \). We can easily find the following non-zero solutions:

\[
\begin{align*}
k_1 &= \begin{bmatrix} \rho & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
k_2 &= \begin{bmatrix} 0 & 0 & \rho & 0 & 0 & 0 & 0 \end{bmatrix}^T \\
k_3 &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & \rho \end{bmatrix}^T
\end{align*}
\]

where \( \rho \) can be any non-zero value. We can find that \( k_1, k_2 \) are related with the kinematic parameters, while \( k_3 \) always holds, which results from no constraints on \( O_{zc} \) for ground robots and it has no matter with the kinematic parameters. Through the found null spaces, we can draw the following conclusions: (i) \( Y_t \) and \( O_{yc} \) are indistinguishable; (ii) \( X_t \) and \( O_{xc} \) are indistinguishable; (iii) the vertical direction of extrinsic parameters \( O_{zc} \) is always unidentifiable for skid-steering robot moving on ground, no matter whether the kinematic parameters are calibrated online.

However, \( O_c \) in Eq. 47b is under quite different properties. Similarly, we investigate the non-zero \( k \) that satisfies \( O_c k = 0 \), which requires the all of the following to be true:

\[
\begin{align*}
-\ddot{w}(t) k_1 + a_1(t) k_2 - e_1^T C R [\dot{v}_C(t)] \cdot k_1 e_1 = 0, k_3 = 0, k_5 = 0, \\
\ddot{w}(t) k_3 - e_2^T C R [\dot{v}_C(t)] \cdot k_7 e_2 = 0, a_2(t) k_4 - a_1(t) k_5 = 0
\end{align*}
\]

However, since \( \ddot{w}(t), \dot{v}_C(t), a_1(t), a_2(t) \) are time variant under general motion, we can only find such a non-zero vector \( k = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ \rho] \) where \( \rho \) can be any non-zero value. We can draw the conclusion: (iv) the first two dimensions of rotation between camera and odometer are identifiable under general motion.

Based on the derived observability properties, it is important to point out the following algorithm design issues: (i) Unlike online calibration algorithms in other literature [18], [38], [39], extrinsic parameters between camera and odometer are not observable and cannot be calibrated online. (ii) The first two dimensions of the extrinsic rotation between sensors remain observable and thus can be included in the state vector of an algorithm for online calibration.

V. EXPERIMENTAL RESULTS

In this section, we provide experimental results that support our claims in both algorithm design and theoretical analysis. Specifically, we conducted real-world experiments and simulation tests to demonstrate 1) the advantages and necessities of online estimating kinematic parameters in visual (inertial) localization systems and 2) the observability and convergence properties of the skid-steering kinematic parameters under different settings.

In our experiments, we used two testing skid-steering robots based on the commercially available Clearpath Jackal robot [1] (see Fig. 1), with both ‘localization’ sensors and ‘ground-truth sensors’ equipped. For ‘localization’ sensors, we used a 10Hz monocular global shutter camera at a resolution of 640×400, a 200Hz Bosch BMM160 IMU, and 100Hz wheel encoders. The ‘ground truth’ sensor mainly relies on RTK-GPS with centimeter-level precision. All sensors used in our experiment are synchronized by hardware and calibrated offline via [40]. We note that the offline calibration procedure is an important prerequisite in our experiments since the extrinsic position between the odometer and camera has shown to be constantly unobservable. All the experiments are conducted by first dataset collection and subsequently offline processing using an Intel Core i7-8700 @ 3.20GHz CPU, to allow repeatable comparison between different methods.

A. Real-world Experiment

In the first set of experiments, we focus on validating the effectiveness of the proposed skid-steering model as well as the localization algorithm. Specifically, we investigated the localization accuracy by estimating skid-steering kinematic parameters \( \xi \) (Eq. 4) online and compared that to the competing methods. To demonstrate the generality of our method, we conducted experiments under various environmental conditions. As shown in Fig 3, the environments involved in our robotic data collection include (a) lawn, (b) cement brick, (c) wooden bridge, (d) muddy road, (e) asphalt road, (f) ceramic tiles, (g) carpet, and (h) wooden floor. Also, Fig. 4 shows one representative trajectory and visual features estimated by the proposed method on a selected sequence, e.g., “SEQ20-CP01”, in which our robot traversed both outdoors and indoors.

We note that since GPS signal is not always available in all tests (e.g., indoor tests), we use both final drift and root-mean-squared error (RMSE) of absolute translational error (ATE) [41] as our metrics. To make this possible, we started and terminated each experiment in the same position. It is also important to point out that, in the research community, it is preferred to use publicly-available datasets to conduct experiments to facilitate comparison between different researchers. However, most localization datasets publicly available either utilize passenger cars (KITTI [42], Kaist Complex Urban [43], Oxford Robotcar [44]) or lack of one or multiple synchronized low cost sensors (NCLT [45], and Canadian 3DMap [46]).
To this end, we also plan to release a comprehensive dataset specifically with low-cost sensors, as our future work.

1) Localization Accuracy: We first conducted an experiment to show the benefits gained by modeling and estimating skid-steering parameters online. In this experiment, three sets of setup are compared, i.e., two provably observable methods and one baseline method. Specifically, those methods are 1) VIO W/ \(\xi\) : using measurements from a monocular camera, an IMU, and an odometer via the proposed estimator by estimating the full 5 skid-steering kinematic parameters \(\xi\) online; 2) VO W/ ICR : using monocular camera and odometry measurements (without an IMU), and performing localization by estimating the 3 ICR parameters \(\xi_{ICR}\) online; 3) VIO W/O \(\xi\) : using measurements from monocular camera, an IMU and odometry, and utilizing Eq. 7 for localization without explicitly modeling \(\xi\). We note that, in traditional methods when \(\xi\) is not modeled, Eq. 7 can be considered as one-parameter (i.e., b) approximation of skid-steering kinematics. We also point out that, compared to the third setup, the configuration VIO W/O ICR using a monocular camera, odometer measurements, and using Eq. 7 for localization without modeling \(\xi_{ICR}\) without IMU measurements is regarded as inferior to the third setup, so we do not show the results in this configuration.

In Table. I, we show the final drift errors on 23 representative sequences, which cover all the eight types of terrains (a)-(g) shown in Fig. 3. We also note that some sequences cover multiple types of terrains. Since the two robots were used for data collection, we use the notation “CP01, CP02” to denote the robot names in Table. I. In addition, we highlight the results with severe drift (error of norm is over 12m) by red color. The results clearly demonstrate that when skid-steering kinematic parameter \(\xi\) is estimated online, the localization accuracy can be significantly improved. This validates our claim that, in order to use odometer measurements of skidding robots, the complicated mechanism must be explicitly modelled to avoid accuracy loss. It should be noted that all the runs in Table. I started from the same initial guess for \(\xi\), to ensure fair comparisons. We also note that, the method of using an IMU and estimating the full 5 kinematic parameters performs best among those methods, by modeling the time-varying scale factors. In fact, the method of estimating only 3 ICR parameters with visual and odometer sensors works well for a portion of the dataset while fails in others (e.g., the datasets under (b,f) categories). This is due to the fact that those datasets involve terrain conditions changes, and the scale factor also changes. If those factors are not model, the performance will drop. Moreover, we note that, under those conditions (e.g., (b,f)), the best performing method still works not as good as the performance in other data sequences. This is due to the fact that we used ‘random walk’ process to model the ‘environmental condition’ changes, which is not the ‘best’ assumption when there are rapid road surface changes. We
Fig. 3: Skid-Steering robot traverses variable terrains: (a) lawn, (b) cement brick, (c) wooden bridge, (d) muddy road, (e) asphalt road, (f) ceramic tiles, (g) carpet, and (h) wooden floor.

Fig. 4: Skid-steering robots traversed outdoors and indoors. The left part shows the representative images with visual features recorded at positions marked by green circles respectively. The right part shows the estimated trajectory red curve, and estimated 3D landmarks by black dots.

The results demonstrate that estimating $\xi$ is beneficial for trajectory tracking. In order to provide insight into how the error of each algorithm grows with the trajectory length, we also calculate the calculated relative pose error (RPE) averaged over all the sequences when GPS measurements are available. The RPE results are shown in Table. II, which also support our algorithm claims.

2) Convergence of Kinematic Parameters: In this section, we show experimental results to demonstrate the convergence properties of $\xi$ and $\xi_{ICR}$, in systems that we theoretically claim observable. Unlike the experiments in the previous section, which utilized the method described in Sec. III-C for kinematic parameter initialization, we manually added extra errors to the kinematic parameter for the tests in this section, to better demonstrate the observability properties. Specifically, for the kinematics-constrained VIO system, we added the following extra error terms to initial kinematic parameters

$$\delta X_v = 0.08, \delta Y_l = 0.14, \delta Y_r = -0.1, \delta \alpha_l = 0.2, \delta \alpha_r = 0.2$$

For the kinematics-constrained VO system, we only add error terms to $\xi_{ICR}$. To show details in parameter convergence properties, We carried out experiments on representative indoor and outdoor sequences, “SEQ8-CP01, SEQ18-CP01, SEQ19-CP01”. In Fig. 7, the estimates of the full kinematic parameters $\xi$ in VIO are shown, along with the corresponding uncertainty envelopes. The convergence of $\xi_{ICR}$ in VO are also shown in Fig. 8. The results demonstrate that the kinematic parameters $\xi$ in the VIO quickly converge to stable values, and remains slow change rates for the rest of the trajectory. Similar behaviours will also leave the terrain detection as future work.

In some sequences where GPS signals were available across the entire data sequence, we also evaluated the root mean square errors (RMSE) [30] of absolute translational error (ATE) [41]. To compute that, we interpolated the estimated poses to get the ones corresponding to the timestamp of the GPS measurements. The RMSE errors are shown in Table. II, where we highlight the bad results (over 12m) by underlines.
can also be observed for $\xi_{ICR}$ when only a monocular camera and odometer sensors are used. The results exactly meet our theoretical expectations that $\xi$ in VIO and $\xi_{ICR}$ in VO are both locally identifiable under general motion. We also note that, since it is not feasible for obtaining high-precision ground truth for $\xi$, the correctness of those values cannot be 'directly' verified. Instead, they can be evaluated either based on the theoretical expectations that $\xi$ and odometer sensors are used. The results exactly meet our initial guess. Similar to the previous tests, three algorithms were conducted here by using the measurements from a camera, an IMU and odometer: 1) VIO with estimating $\xi$; 2) VO with estimating $\xi_{ICR}$ online; 3) VIO without estimating $\xi$.

### Table I: Final Drift for three different setups on 23 sequences which covers 8 types of terrain.

| Sequence   | Length(m) | Terrain | VIO W/ $\xi$ | VO W/ ICR | VIO W/O $\xi$ |
|------------|-----------|---------|-------------|-----------|---------------|
| SEQ1-CP02  | 232.30    | (b)     | 3.644       | 0.420     |               |
| SEQ2-CP01  | 193.63    | (f)     | 0.800       | 0.409     |               |
| SEQ3-CP01  | 632.64    | (h,f)   | 7.150       | 1.457     |               |
| SEQ4-CP01  | 629.96    | (f)     | 1.427       | 1.139     |               |
| SEQ5-CP01  | 626.83    | (h,f)   | 8.109       | 1.563     |               |
| SEQ6-CP01  | 212.59    | (g)     | 7.399       | 0.528     |               |
| SEQ7-CP01  | 51.44     | (a)     | 0.206       | 0.090     |               |
| SEQ8-CP01  | 204.81    | (c)     | 0.766       | 0.716     |               |
| SEQ9-CP01  | 77.63     | (c)     | 0.319       | -0.020    |               |
| SEQ10-CP01 | 27.09     | (a)     | 0.204       | 0.170     |               |
| SEQ11-CP01 | 270.41    | (c,b)   | 0.664       | 0.504     |               |
| SEQ12-CP01 | 436.19    | (c,e)   | 0.734       | 0.710     |               |
| SEQ13-CP01 | 28.64     | (d)     | 0.093       | 0.043     |               |
| SEQ14-CP01 | 372.15    | (b)     | 10.016      | 1.099     |               |
| SEQ15-CP02 | 81.03     | (h)     | 2.573       | 0.878     |               |
| SEQ16-CP02 | 53.49     | (h)     | 0.702       | 0.716     |               |
| SEQ17-CP01 | 110.55    | (b)     | 1.048       | 0.515     |               |
| SEQ18-CP01 | 104.63    | (h,b)   | 0.488       | 0.152     |               |
| SEQ19-CP01 | 214.66    | (h,b)   | 0.999       | 0.814     |               |
| SEQ20-CP01 | 254.30    | (h,b)   | 0.838       | 0.814     |               |
| SEQ21-CP01 | 629.16    | (h,f)   | 1.829       | 1.329     |               |
| SEQ22-CP01 | 633.53    | (h,f)   | 4.405       | 1.249     |               |
| SEQ23-CP01 | 651.94    | (b)     | 3.428       | 1.560     |               |

**Mean**

2.514  1.174  0.183  0.610  32.878  28.429  4.637  0.225  8.356  -1.917  -0.602  2.956
estimating $\xi_{ICR}$; 2) estimating $\xi$; 3) used fixed $\xi$ with a relatively good initial guess, obtained by the final estimate of running our online estimation algorithm.

We conducted experiments on 8 sequences named ABL-SEQ1 $\sim$ ABL-SEQ8, and each configuration corresponds to two sequences in ascending order (e.g., ABL-SEQ1 and ABL-SEQ2 correspond to the ‘normal’ condition). The evaluation methods used here are as same as the ones used in the previous section, which include both final drift and RMSE. In Table. IV, we show the final drift of three different localization methods. On the other hand, the RMSE of ATE is given in Table. V. Additionally, RPE was shown in Fig. 9 and Table. VI. Those results demonstrate that, when a robot is in normal mechanical condition, and the road condition is without large variance, there are minor differences between estimating the full 5 kinematic parameters $\xi$ and online estimating only the 3 parameters $\xi_{ICR}$, when good correction factors $\alpha_l, \alpha_r$ are given and kept constant. This is due to the fact that the correction factors reflect the transmission efficiency of the robot and are not subject to fast changes in the general case. However, if there are noticeable changes in the robotic mechanical condition, e.g., weight and center of mass change by carrying a large package or tire pressure changes after long-term usage, the correction factors $\alpha_l, \alpha_r$ will be changed significantly. In such cases, the overall estimation algorithm benefits significantly by online estimating $\xi$. We also show the estimated trajectories compared with RTK-GPS measurement in Fig. 10, for the representative runs.

B. Simulation Experiments

We also perform Monte-Carlo simulations to investigate our proposed method specifically for parameter calibration precision, since this cannot be verified in real-world tests. The synthetic trajectory is generated by simulating a real-world trajectory with a length of $205.4m$, using the method introduced in [47]. To generate noisy sensory measurements, we have used zero-mean Gaussian vector for all sensors with the following standard deviation (std) values. Pixel std for visual measurements is $0.6$ pixels, odometer stds for the left and right wheels are both $0.0245$ m/s, gyroscope and accelerometer measurement stds are $9 \cdot 10^{-4}$ rad/s and $1 \cdot 10^{-2}$ m/s$^2$, and finally the stds representing the random walk behavior of gyroscope and accelerometer biases are $1 \cdot 10^{-2}$ rad/s$^2$ and $1 \cdot 10^{-2}$ m/s$^3$ respectively. Additionally, since skid-steering kinematic parameters cannot be known in advance, we initialize $\xi$ in our simulation tests by adding an error vector to the ground truth values. The noise vector is sampled from zero-mean Gaussian distribution with std $8 \cdot 10^{-2}$ for all elements in $\xi$.

To collect algorithm statistics, we conducted 15 Monte-Carlo tests and compute parameter estimation results for $\xi$. Specifically, we computed the mean and std of calibration errors averaged over the second half of the trajectory are: $-0.0276 \pm 0.0192, -0.0108 \pm 0.0108, -0.0212 \pm 0.0054$ rad/s and $-0.0019 \pm 0.0026, 0.0219 \pm 0.0192, 0.0189 \pm 0.0157$. In this case, the calibration errors averaged over the second half of the trajectory are: $-0.0276 \pm 0.0067, 0.0199 \pm 0.0118, 0.0054 \pm 0.0026, 0.0192 \pm 0.0157, 0.0189 \pm 0.0157$. Since simulation tests provide absolute ground truth, it is also interesting to investigate the accuracy gain by estimating $\xi$ online. Fig. 11 demonstrates the estimated trajectory when $\xi$ is estimated online, or $\xi$ is fixed during estimation as well as the ground truth. This clearly demonstrates that, by the online estimation process, the localization accuracy can be significantly improved. The averaged RMSE of rotation and translation for those two competing methods in this Monte-Carls tests are $0.042 \pm 0.023 rad, 2.051 \pm 0.830 m$ and $0.154 \pm 0.0635 rad, 4.617 \pm 2.563 m$, respectively.
steering robots. To guide the estimator design, we conduct
prevent performance reduction in the lifelong mission of skid-
ICRs and correction factors, which are online estimated to
the kinematics of skid-steering robots by using both track
inflation changes, and terrain conditions, we explicitly model
fectness of mechanical design, mass center changes, tire
for the complicated track-to-terrain interactions, the imper-
sliding-window BA. In particular, in order to compensate
multi-modal measurements are fused in a tightly-coupled
estimation method specialized for skid-steering robots, where

\[ \xi \]

Fig. 7: While given bad initial values, the ICR parameters \( \xi_{ICR} \) are able to converge to the reasonable values in the kinematics-constrained VO system. The online estimated \( \xi_{ICR} \) and the associated \( \pm 3\sigma \) envelopes are shown on sequences “ SEQ8-CP01, SEQ18-CP01, SEQ19-CP01” from left to right.

| Sequence | Length(m) | Terrain | Config. | \( \xi \) | \( \sigma \) |
|----------|-----------|---------|---------|-------|-------|
| ABL-SEQ1 | 167.30    | (e)     | (i)     | 0.316 | 0.038 |
| ABL-SEQ2 | 147.76    | (e)     | (i)     | 0.349 | -0.109 |
| ABL-SEQ3 | 152.23    | (e)     | (ii)    | 0.318 | -0.193 |
| ABL-SEQ4 | 152.80    | (e)     | (ii)    | 0.406 | -0.240 |
| ABL-SEQ5 | 237.36    | (e)     | (iii)   | 8.700 | 0.046  |
| ABL-SEQ6 | 232.43    | (e)     | (iii)   | 8.102 | -0.161 |
| ABL-SEQ7 | 232.54    | (e)     | (iv)    | 9.509 | 0.189  |
| ABL-SEQ8 | 233.07    | (e)     | (iv)    | 10.055| 0.256 |

TABLE IV: Ablation Experiments Results: Final drift.

Fig. 8: While given bad initial values, the ICR parameters \( \xi_{ICR} \) are able to converge to the reasonable values in the kinematics-constrained VO system. The online estimated \( \xi_{ICR} \) and the associated \( \pm 3\sigma \) envelopes are shown on sequences “ SEQ8-CP01, SEQ18-CP01, SEQ19-CP01” from left to right.

VI. CONCLUSIONS

In this paper, we propose a novel kinematics and pose estimation method specialized for skid-steering robots, where multi-modal measurements are fused in a tightly-coupled sliding-window BA. In particular, in order to compensate for the complicated track-to-terrain interactions, the imper-
fectness of mechanical design, mass center changes, tire
inflation changes, and terrain conditions, we explicitly model
the kinematics of skid-steering robots by using both track
ICRs and correction factors, which are online estimated to
prevent performance reduction in the lifelong mission of skid-
steering robots. To guide the estimator design, we conduct
detailed observability analysis for the proposed algorithm
under different setup conditions. Specifically, we show that
the kinematic parameter vector \( \xi \) is observable under general
motion when measurements from an IMU are added and
odometer-to-camera extrinsic parameters are calibrated offline.
In other situations, degenerate cases might be entered and
reduced precision might be incurred. Extensive real-world
experiments including ablation study and simulation tests are
also provided, which demonstrate that the proposed method is
able to compute skid-steering kinematic parameters online and
yield accurate pose estimation results. Experimental results
also validate our observability analysis, showing that under
theoretically observable conditions the corresponding param-
TABLE V: Ablation Experiments Results: RMSE of ATE (m).

| Segment Length (m) | VIO W/ ICR | VIO W/ ξ | VIO W/ Fixed ξ |
|-------------------|------------|----------|----------------|
| 9.00m             | 0.24       | 0.23     | 0.28           |
| 18.00m            | 0.17       | 0.17     | 0.24           |
| 27.00m            | 0.14       | 0.14     | 0.27           |
| 36.00m            | 0.16       | 0.16     | 0.23           |
| 45.00m            | 2.20       | 1.85     | 2.49           |
|                  | 2.20       | 1.87     | 2.40           |
|                  | 2.54       | 2.13     | 2.82           |
|                  | 2.54       | 2.09     | 2.75           |

Fig. 9: Ablation Experiments Results: boxplot of the relative trajectory error statistics over all the sequences where RTK-GPS measurements are available. This plot will best seen in color.

TABLE VI: Ablation Experiments Results: Mean of RPE (m) for Different Segment Length.

| Segment Length (m) | VIO W/ ICR | VIO W/ ξ | VIO W/ Fixed ξ |
|-------------------|------------|----------|----------------|
| 9.00m             | 0.53       | 0.48     | 0.54           |
| 18.00m            | 1.00       | 0.91     | 1.01           |
| 27.00m            | 1.47       | 1.33     | 1.49           |
| 36.00m            | 1.92       | 1.75     | 1.95           |
| 45.00m            | 2.36       | 2.14     | 2.39           |

APPENDIX

To make this article self-contained, we also provide detailed formulation for each term in our cost function.

A. Camera Cost Function

In the sliding-window BA, only the keyframes are optimized for computational saving. We use a simple heuristic for keyframe selection: the odometer prediction has a translation or rotation over a certain threshold (in all the experiments, 0.2 meter and 3 degrees). Since the movement form of the ground robot is simple, and it can be well predicted by the odometer in a short period of time. Unlike existing methods [48], [49], which extract features and analyze the distribution of the features for keyframe selection, the non-keyframe will be dropped immediately without any extra operations in our framework. Among keyframes selected into the sliding-window, corner feature points are extracted in a fast way [50] and tacked with FREAK [51] descriptors.

The successfully tracked features across multiple keyframes will be initialized in the 3D space by triangulation. By denoting $z_{i,j}$ the visual measurement of a 3D feature $Gp_{fj}$ observed by the $C_i$th camera keyframe, the visual reprojection error [37] in normalized image coordinate is given by:

$$C_i(G_{R}, G_{p_{i,j}}) = \|z_{i,j} - \pi(G_{R}, G_{p_{i,j}}, G_{fj})\|_2^{2},$$

where $G_{R}$ is the intrinsic camera, $G_{p}$ is the extrinsic transformation between camera and odometer, and $\pi$ is the IMU integration function. Since the IMU integration requires odometer to IMU extrinsic parameters to transform.

B. IMU Constraints

The IMU provides readings of both accelerometer and gyroscope as follows:

$$\omega_m = \omega_I + b_\omega + n_\omega$$

$$a_m = a_I - \frac{1}{G_{R}}G_{g} + b_a + n_a$$

where $G_g$ is the known global gravity vector, $b_\omega$ and $b_a$ the time-varying gyroscope and accelerator bias vectors, and $n_\omega$ and $n_a$ denote white Gaussian measurement noise. The IMU integration process is characterized by:

$$\dot{x}_k = \left[ G_{g}, G_{a}, \dot{G}_{g}, \dot{G}_{a}, \dot{G}_{\omega} \right]^T = f(x_{k-1}, W_m, A_m)$$

where $W_m, A_m$ are the gyroscope and accelerometer measurements during the time interval $t \in (t_{k-1}, t_k)$, and $f(\cdot)$ is the IMU integration function. Since the IMU integration is widely investigated in research communities [17], [25], [52], and we here ignore the details on $f(\cdot)$. The associated uncertainty matrix (i.e., linearized noise information matrix) of the prediction process $A_f$ can also be obtained by linearizing the function $f(\cdot)$. As a result, the IMU cost term can be summarized by:

$$C_f(x_k, x_{k-1}) = \|x_k \oplus f(x_{k-1}, W_m, A_m)\|_2^{2}$$

which provides pose constraints between consecutive keyframes. We also note that, the IMU cost function requires odometer to IMU extrinsic parameters to transform.
states in odometer frame to IMU frame, which are also calibrated offline. After minimizing Eq. 51, the states \( \xi \) will be marginalized, and the contained information will be incorporated into the prior cost term.

C. Motion Mainfold constraints

Finally, since the skid-steer robot navigates on ground surfaces, its trajectories can also be constrained by the prior knowledge about the shape of surface manifold. Specifically, we utilize our method presented in [26] to approximate ground surfaces using quadratic polynomials, the following holds:

\[
\begin{align*}
\dot{\mathbf{m}}_p^{(R, G \mathbf{p}_o)} &= \frac{1}{2} \mathbf{G} \mathbf{p}_o \mathbf{p}_o^\top \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix} \mathbf{G} \mathbf{p}_o + \begin{bmatrix} b_1 & b_2 \end{bmatrix} \mathbf{G} \mathbf{p}_x \mathbf{p}_x^\top + \begin{bmatrix} c \end{bmatrix} \mathbf{G} \mathbf{p}_o + \begin{bmatrix} 0 \end{bmatrix} \\
\mathbf{m}_r^{(G \mathbf{R}_i, G \mathbf{p}_o)} &= \frac{\partial m_{p_x}}{\partial G \mathbf{p}_o} = 0, \text{ and } m_{p_x}^{(G \mathbf{p}_o)} = 0
\end{align*}
\]

where \( \mathbf{m} = [a_1, a_2, a_3, b_1, b_2, c]^\top \) to denote the manifold parameters. By utilizing the quadratic surface approximation, we are able to define the following cost function for both rotation and position terms [26]:

\[
C_M^{(G \mathbf{R}, G \mathbf{p}_o, m_k, m_{k-1})} = \left\| \begin{bmatrix} m_k - m_{k-1} \, m_{p_x}^{(G \mathbf{p}_o)} \end{bmatrix} \right\|_{\Lambda_m} (53)
\]

for all \( i \in [k - s + 1, k] \). \( m_{k-1} \) and \( m_k \) denotes the manifold parameters characterize the motion manifold across the last and the current sliding window, respectively. Moreover, \( \Lambda_m \) is the information matrix describing the uncertainties in both localization states and the surface manifold approximation itself, which is described in detail in [26].

Fig. 10: In the ablation study, the trajectories of RTK-GPS (ground truth) and the estimated trajectory by the localization method: 1) VIO with online estimating the full kinematic parameters \( \xi \); 2) VIO with online estimating \( \xi_{ICR} \) only; 3) VIO with fixed \( \xi \). The skid-steering robot is under four different conditions: (a) normal; (b) carrying a package with the weight around 3 kg; (c) under low tire pressure; (d) carrying a 3-Kg package and with low tire pressure.

Fig. 11: In simulation experiments, estimated Trajectories aligned with the ground truth trajectory.

REFERENCES

[1] Clearpath Robotics Inc., “Clearpath ground vehicle,” Available: https://www.clearpathrobotics.com/jackal-small-unmanned-ground-vehicle/, 2019.
[2] P. Newswire, “Skid steer loader market,” Available: https://finance.yahoo.com/news/skid-steer-loader-market-anticipated-102000939.html, 2019.
[3] J. Vincent, “Fedex unveils autonomous delivery robot,” Available: https://www.theverge.com/2019/2/27/18242834/delivery-robot-fedex-sameday-bot-autonomous-trials, 2019.
[4] B. F. Rubin, “Amazon’s scout robots,” https://www.cnet.com/news/amazons-scout-robots-thats-no-cooler-thats-your-prime-delivery/, 2019.
[5] G. Anousaki and K. J. Kyriakopoulos, “A dead-reckoning scheme for skid-steered vehicles in outdoor environments,” in IEEE International Conference on Robotics and Automation, vol. 1, 2004, pp. 580–585.
