MUSE: A Real-Time Multi-Sensor State Estimator for Quadruped Robots

Ylenia Nisticò[1,2,*], João Carlos Virgolino Soares[1], Lorenzo Amatucci[1,2], Geoff Fink[1,3], and Claudio Semini[1]

***Abstract*—This paper introduces an innovative state estimator, MUSE (MUlti-sensor State Estimator), designed to enhance state estimation's accuracy and real-time performance in quadruped robot navigation. The proposed state estimator builds upon our previous work presented in [1]. It integrates data from a range of onboard sensors, including IMUs, encoders, cameras, and LiDARs, to deliver a comprehensive and reliable estimation of the robot's pose and motion, even in slippery scenarios. We tested MUSE on a Unitree Aliengo robot, successfully closing the locomotion control loop in difficult scenarios, including slippery and uneven terrain. Benchmarking against Pronto [2] and VILENS [3] showed 67.6% and 26.7% reductions in translational errors, respectively. Additionally, MUSE outperformed DLIO [4], a LiDAR-inertial odometry system in rotational errors and frequency, while the proprioceptive version of MUSE (P-MUSE) outperformed TSIF [5], with a 45.9% reduction in absolute trajectory error (ATE).**

***Index Terms*—state estimation, localization, sensor fusion, quadruped robots.**

I. Introduction

QUADRUPED robots are renowned for their ability to traverse difficult terrains and, for this reason, they have gained increasing importance in fields such as academic research, inspection, and monitoring. Perceptive information is crucial in unstructured environments [6], where accurate state estimation enables robust locomotion, real-time feedback, and stable gait control.

This work introduces **MUSE**, a **MU**lti-sensor **S**tate **E**stimator for quadruped robots, emphasizing coordinated sensing, real-time data processing, and refined fusion algorithms, critical for dynamic movement control and autonomous task execution.

Early research in state estimation has focused on combining proprioceptive (e.g., IMU, encoders, torque sensors) and exteroceptive (e.g., cameras, LiDARs) sensors. Exteroceptive sensors provide accurate, low-drift pose estimates, essential for autonomous navigation and SLAM [7], but may fail under adverse conditions or introduce time delays due to limited frequencies. Hence, they are often paired with high-frequency proprioceptive measurements, which remain reliable in settings where exteroceptive sensors struggle (e.g., poor lighting or limited features). Notable sensor-fusion methods include ORB-SLAM3 [8], a versatile framework for visual-inertial and multi-map SLAM, and DLIO [4], known for its computationally efficient LiDAR-inertial odometry.

Specifically for legged robots, state-of-the-art methods take into account also leg kinematics, which provides detailed information about the movement and positioning of each leg, and can help to improve the estimate. Part of the work in legged robots state estimation has focused solely on proprioception. For instance, the study in [9] employs an Observability Constrained Extended Kalman filter to estimate foothold positions and overall robot pose without assuming fixed terrain geometry, [5] propose TSIF, a residual-based recursive filter for state estimation in dynamic systems without requiring explicit process models, while [10] uses an Invariant Extended Kalman Filter (InEKF), to fuse contact-inertial dynamics with forward kinematic corrections.

However, proprioceptive-only approaches often suffer from drift, especially on soft or slippery terrains. A study done in [11] examining the impact of soft terrain revealed that relying solely on proprioception, with an estimator assuming rigid contacts, led to significant drift compared to navigating rigid terrain. Techniques such as [12] and [13] address this by filtering out outliers and modeling slip, while [14] uses factor graphs to account for uncertainty in contact points. In another work [15], the authors introduced an innovative InEKF designed specifically for legged robots, relying solely on proprioceptive sensors and incorporating robust cost functions in the measurement update. Although the use of these robust cost functions significantly reduced drift, they were not able to completely eliminate it.

These studies investigating the impact of terrains on legged robots' state estimation reveal the limitations of assuming non-slip conditions in the state estimator. While avoiding this assumption can lead to improved results, proprioceptive state estimation alone falls short of providing a drift-free pose, emphasizing the necessity of incorporating exteroceptive sensor data for enhanced accuracy.

To achieve low-drift estimates, many works fuse leg kinematics and inertial measurements with exteroceptive sensors. An example of a multi-sensor state estimator is Pronto [2], which integrates stereo vision and LiDAR data into an EKF that also combines IMU and leg kinematics. Although effective, Pronto is built with a non-slip assumption, meaning it operates under the premise that the robot maintains constant contact with the ground without experiencing slippage or falling. This assumption simplifies the state estimation process

Manuscript received: November, 3, 2024; Revised January, 3, 2025; Accepted March, 2, 2025.

This paper was recommended for publication by Editor Pascal Vasseur upon evaluation of the Associate Editor and Reviewers' comments.

This work was supported by the ASI PEGASUS Project.

[1] Dynamic Legged Systems (DLS), Istituto Italiano di Tecnologia (IIT), Genoa, Italy. {first_name.first_surname}@iit.it

[2] Dipartimento di Informatica, Bioingegneria, Robotica e Ingegneria dei Sistemi (DIBRIS), University of Genoa, Genoa, Italy

[3] Department of Engineering, Thompson Rivers University, Kamloops, BC, Canada. gefink@tru.ca

*Corresponding author: ylenia.nistico@iit.it

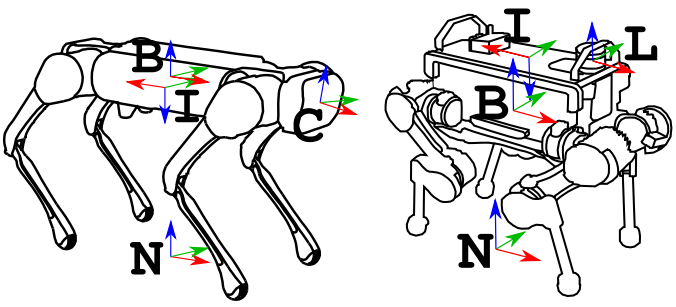
Digital Object Identifier (DOI): 10.1109/LRA.2025.3553047

but overlooks the possibility of slippage or loss of contact, which can significantly impact the accuracy and reliability of the estimated state, particularly in challenging terrains. In [16] an InEKF is employed for state estimation in a bipedal robot navigating slippery terrain. This method fuses Realsense T265 vision with inertial and leg-kinematic data, using an online noise parameter to adapt to measurement noise. STEP [17], instead, adopts a stereo camera for speed estimation and pre-integrated foot velocity factors, bypassing explicit contact detection and non-slip assumption. However, it is worth noting that these state estimators heavily depend on camera inputs, which may sometimes be unreliable, potentially affecting the accuracy and robustness of the estimated states. VILENS, proposed in [3] combines IMU, kinematics, LiDAR, and camera data using factor graphs to ensure reliable estimation, even when individual sensors may fail. Nevertheless, it is important to note that VILENS has not demonstrated real-time feedback control, and thus remains untested in fully closed-loop operations. More recently, Leg-KILO [18] combines LiDAR odometry with kinematic and inertial measurements to estimate the robot's pose, heavily relying on loop closure to minimize the drift. Loop closures significantly enhance state estimation by reducing drift. However, they might introduce sudden changes in the estimated trajectory. For this reason, state estimators performing loop closures are typically not used to provide feedback directly to controllers that rely on smooth, continuous feedback for real-time applications. Additionally, in situations where loop closures are infrequently, such as in a long corridor, the system will experience significant drift over an extended period.

A. Contribution and Outline

In this letter, we introduce MUSE, an innovative state estimation framework for legged robots that builds upon our earlier research in [1]. In that prior work, we deployed a nonlinear observer for attitude estimation and derived leg odometry from a quadruped model, which was then fused using a Kalman Filter (KF). With MUSE, we extend this approach into a comprehensive state estimation pipeline, incorporating exteroceptive sensors and integrating the slip detection module we presented in [19], wherein a kinematics-based strategy identifies slippage in one or more legs concurrently. This enhancement enables MUSE to deliver low-drift estimates while maintaining robustness against sensor failures and operating effectively in uneven, unstructured environments. In this context, we make the following contributions:

- Integration of a slip detection module in state estimation: to the best of our knowledge, this is the first instance of a multi-sensor state estimation pipeline featuring a module specifically designed for slip detection, crucial when walking on uneven or unstructured terrain.
- Real-time operation: unlike previous works (e.g. VILENS, and STEP [3], [17]), we used MUSE to provide real-time feedback to the locomotion controller during an experiment conducted on the Aliengo robot.
- Online and offline evaluation on different platforms and scenarios: our work was validated on the Aliengo robot in indoor environments on difficult scenarios, and on the ANYmal B300 robot, in the Fire Service College (FSC) Dataset [3] (Fig. 1). We demonstrate improvements of 67.6% and 26.67% in translational errors compared to two state-of-the-art algorithms for quadruped robots, Pronto and VILENS respectively, along with a 45.9% reduction in absolute trajectory error compared to TSIF. Additionally, MUSE shows superior performance in both rotational error and frequency compared to the LiDAR-inertial odometry system, DLIO.



(a) Aliengo frames (b) ANYmal B300 frames. (adapted from [20])

Fig. 1: Robot Reference Frames: the navigation frame $\mathcal{N}$, the body frame $\mathcal{B}$, the IMU sensor frame $\mathcal{I}$, the camera frame $\mathcal{C}$, and the LiDAR frame $\mathcal{L}$ for ANYmal.

For the benefit of the community, we released MUSE's code under an open-source license. The code is available at **https://github.com/iit-DLSLab/muse**. The remainder of this article is presented as follows: Section II describes the formulation of the proposed state estimator; Section III presents the experimental results; Section V concludes with final remarks.

II. State Estimator Formulation

Our objective is to estimate the pose and twist with respect to an arbitrary inertial navigation frame, of a quadruped robot equipped with a combination of proprioceptive and exteroceptive sensors, including IMUs, force sensors, joint sensors (encoders and torque sensors), cameras, and LiDARs. The state estimator consists of five major components, as shown in Fig. 2: an exteroceptive (camera or LiDAR) odometry module, an attitude observer, slip detection, leg odometry, and a sensor fusion algorithm.

In this work, we used the dynamic and kinematic models of Aliengo and ANYmal B300. However, the state estimator modules are general and can be applied to any legged robot with the proper sensors. The following reference frames (Fig. 1) are introduced: the navigation frame $\mathcal{N}$, which is assumed inertial, the body frame $\mathcal{B}$ which is located at the geometric center of the trunk, the IMU sensor frame $\mathcal{I}$, which is located at the origin of the accelerometer of the IMU mounted onto the trunk of the robot, the camera frame $\mathcal{C}$ for Aliengo (Fig. 1a), located at the optical center of the camera mounted in the front of the robot, and the LiDAR frame $\mathcal{L}$ for ANYmal (Fig. 1b), located at the center of the sensor mounted on top of the robot. The basis of the body frame is orientated forward, left, and up. We denote the reference frame of a variable using a right

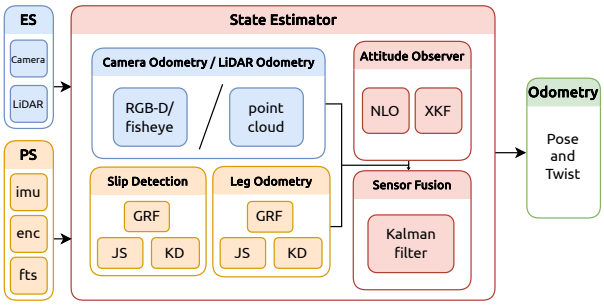


Fig. 2: MUSE utilizes two exteroceptive sensors (ES): Camera and LiDAR, and three proprioceptive sensors (PS): IMU, encoders, force/torque sensors. It comprises five main components: camera odometry or LiDAR odometry, an attitude observer (AO), slip detection (SD), leg odometry (LO), and a sensor fusion algorithm (SF). The AO includes a nonlinear observer (NLO) and an eXogenous Kalman Filter (XKF). The SD and LO include joint state (JS), robot kinematics/dynamics (KD), ground reaction forces (GRF), and leg odometry models. The SF utilizes a Kalman filter to estimate odometry.

superscript, i.e., x^n, x^b, x^i, x^c, and x^l denote x in $\mathcal{N}$, $\mathcal{B}$, $\mathcal{I}$, $\mathcal{C}$, and $\mathcal{L}$ respectively.

The robots are equipped with a six-axis IMU on the trunk, and every joint contains an absolute encoder. Aliengo has cameras at the front, while ANYmal is additionally equipped with torque sensors and a LiDAR. The sensors measure $\tilde{x} = x + b_x + n_x$, where b_x, and n_x are the bias and noise of x, respectively. All of the biases are assumed to be constant or slowly time-varying, and all noise variables have a Gaussian distribution with zero mean. The sensors are described as follows: 1) Camera: for the indoor lab experiment on Aliengo, we used a lightweight tracking camera, specifically the Intel Realsense T265. It features an IMU, two fisheye lenses with 163° of field of view, and the capability to provide camera odometry at up to 200 Hz. 2) LiDAR: for the FSC-Dataset on ANYmal B300, we used only the Velodyne VLP16 LiDAR as an external sensor, whose frequency is approximately 10 Hz. 3) IMU: the IMU consists of a 3-DoF gyroscope and 3-DoF accelerometer. The accelerometer measures a specific force $f_i^i = a^i + g^i \in \mathbb{R}^3$: where $a^i \in \mathbb{R}^3$ is the acceleration of the body in $\mathcal{I}$ and $g^i \in \mathbb{R}^3$ is the acceleration due to gravity in $\mathcal{I}$. The gyroscope measures angular velocity $\omega^i \in \mathbb{R}^3$ in $\mathcal{I}$. 4) Encoders and Torque sensors: the absolute encoders are used to measure the joint position $q_i \in \mathbb{R}$ and joint speed $\dot{q}_i \in \mathbb{R}$, respectively. ANYmal is equipped with torque sensors that directly measure $\tau_i \in \mathbb{R}^3$, while for Aliengo, the joint torque is estimated based on the motor current.

A. Camera Odometry and LiDAR Odometry

In the lab experiments with Aliengo, odometry data is obtained using a T265 tracking camera. Conversely, for the FSC-Dataset, LiDAR odometry is employed using the KISS-ICP algorithm [21].

B. Contact Estimation

To estimate the foot contact with the ground, the contact point is assumed to be on a fixed point at the center of the foot. The contact state $\alpha \in \mathbb{R}^4$ is estimated by computing the ground reaction forces (GRFs) from the dynamics equation of motion:

$$M(\bar{x})\ddot{\bar{x}} + h(\bar{x}, \dot{\bar{x}}) = \bar{\tau} + J^T F_{\text{grf}} \tag{1}$$

where $\bar{x} = [x^T \eta^T q^T]^T \in \mathbb{R}^{18}$ is the generalized robot state, given by the position and attitude of the base, and the joint angles. Then $\dot{\bar{x}} \in \mathbb{R}^{18}$ and $\ddot{\bar{x}} \in \mathbb{R}^{18}$ are the corresponding generalized velocities and accelerations, $M \in \mathbb{R}^{18\times18}$ is the joint-space inertia matrix, $h \in \mathbb{R}^{18}$ is the vector of Coriolis, centrifugal, and gravity forces, $\bar{\tau} = [\underline{0}\ \tau] \in \mathbb{R}^{18}$ where $\tau \in \mathbb{R}^{12}$ is the vector of joint torques, and finally $F_{\text{grf}} \in \mathbb{R}^{12}$ is the vector of GRFs, while $J \in \mathbb{R}^{18\times12}$ is the floating base Jacobian.

Then, assuming that all of the external forces are exerted on the feet during the *stance* phase, we first estimate the GRFs. Subsequently, the contact state α_i for every leg i is:

$$\alpha_i = \begin{cases} 1 & \|F_{\text{grf},i}\| > F_{\text{min}} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $F_{\text{min}} \in \mathbb{R}$ is the threshold value, and $F_{\text{grf},i} \in \mathbb{R}^3$ is the GRF of the leg i.

C. Leg Odometry and Slip Detection

Leg odometry estimates the incremental motion of the floating base from the forward kinematics of the legs in stable contact with the ground. This measurement can be formulated as either a relative pose or a velocity measurement. In our system, we formulate linear velocity measurements. If there is no slippage, then the contribution of each leg $\ell \in \mathbb{L}$ to the overall velocity of the base is:

$$x_\ell^b = -\alpha_\ell (J_\ell(q_\ell)\dot{q} - \omega^b \times x_\ell^b) \tag{3}$$

and the base velocity is:

$$\dot{x}^b = \frac{1}{n_s} \sum_{\ell}^{\mathbb{L}} \dot{x}_\ell^b \tag{4}$$

where $n_s = \sum_\ell^{\mathbb{L}} \alpha_\ell$ is the number of stance legs.

Leg odometry is prone to drift when the robot is walking on slippery ground. For this reason, we used a slip-detection algorithm, presented in our previous work [19], to compensate for this characteristic drift. The approach is based on kinematics and makes use of velocity and position measurements at the ground contacts. We define:

$$\overline{\Delta V} = \sqrt{\sum_{i=x,y,z} \left(\frac{{}_d\dot{x}_{f_i}^b - \dot{x}_{f_i}^b}{|{}_d\dot{x}_{f_i}^b| + m} \right)^2} \quad , \quad \Delta P = \left\| {}_d x_{f_i}^b \right\| - \left\| x_{f_i}^b \right\| \tag{5}$$

where ${}_d\dot{x}_f^b$ and $\dot{x}_f^b$ are the desired and measured linear velocities of the foot in $\mathcal{B}$, while ${}_d x_{f_i}^b$ and $x_{f_i}^b$ are the desired and measured positions of the foot in $\mathcal{B}$, and m is a tunable parameter used to avoid division by zero. If $\overline{\Delta V}$ and ΔP overcome their respective thresholds ϵ_v and ϵ_p then a slip is detected. We use the flag $\beta_i \in [0, 1]$ for each leg i, whose value is set to 1 if there is a slip detection, 0 otherwise.

Once a single slippage or multiple slippages are detected, we increase the leg odometry covariance R_1 in (8) so that the error in leg odometry does not negatively affect our base pose/velocity estimates.

D. Attitude Observers

To estimate the attitude, we implemented a cascaded structure composed of a nonlinear observer (NLO) and an eXogeneous Kalman Filter (XKF). The XKF linearizes about a globally stable exogenous signal from a NLO. The cascaded structure maintains the global stability properties from the NLO and the near-optimal properties from the KF. The proof of stability is explained in [22].

1) Nonlinear Observer: We use the non-linear observer in [23], an extension of the work introduced in [24], that makes use of symmetry properties for attitude estimation. The comprehensive equations have been elucidated in our previous publication [1].

2) eXogeneous Kalman Filter: The state of the filter is $x = [q^T b^T]^T \in \mathbb{R}^7$ where $q \in \mathbb{R}^4$ is a quaternion, while $b \in \mathbb{R}^3$ is the IMU's bias. The quaternion is used to represent the rotation as it does not contain singularities. The input is $u = \omega^b \in \mathbb{R}^3$ given by the IMU's 3-axis gyroscope, and the dynamics of the filter is:

$$\dot{q}_b^n = \frac{1}{2}\begin{bmatrix} 0 & -(\omega^b - b^b)^T \\ (\omega^b - b^b)^T & -S(\omega^b - b^b) \end{bmatrix} q_b^n$$
$$\dot{b}^b = 0$$

where $S(\cdot)$ is the skew-symmetric matrix function. We use a multiplicative error function, to respect the quaternion norm constraint $e_q = (q_b^n)^{-1} \otimes \hat{q}_b^n$, where $\otimes$ is quaternion multiplication [25].

In ideal conditions, a 3-axis accelerometer and a 3-axis magnetometer provide the measurements (feedback) to the filter. While the magnetometer can be used to estimate the orientation of an object relative to the Earth's magnetic field, there are some limitations when using a magnetometer for determining the orientation of a quadruped robot. Magnetometers are affected by local magnetic fields, due to the presence of high current (electric motors) and metallic objects (buildings) in the environment where the robots operate. For these reasons, we adopted another strategy to obtain the measurement vector. In a standard situation, the magnetometer will give a vector to a constant north. We implemented a "*pseudo-north*" strategy using the exteroceptive sensors (LiDAR or camera) for external odometry. This outputs the rotation from the sensor local frame ($\mathcal{S}$) to the sensor world frame ($\mathcal{SW}$). We used a constant vector that is constant in $\mathcal{N}$, rotated it by the amount given by the sensor orientation, and then we rotated this measurement in the body frame $\mathcal{B}$. The measurement vector is $z = [{f_b}^T {m_b}^T]^T \in \mathbb{R}^6$, where $f_b = R_i^b f_n \in \mathbb{R}^3$ is the acceleration given by the accelerometer rotated in $\mathcal{B}$, and $m_b = R_s^b R_{sw}^s [1\ 0\ 0]^T \in \mathbb{R}^3$ is the "pseudo-magnetometer measure", in which $[1\ 0\ 0]^T$ is a constant vector in $\mathcal{N}$ pointing to a "pseudo" North, rotated in $\mathcal{B}$.

Finally, the equations of the XKF are:

$$\dot{\hat{x}} = f_x + F(\hat{x} - \bar{x}) + K(z - h_x - H(\hat{x} - \bar{x})) \tag{6a}$$
$$\dot{P} = FP + PF^T - KHP + Q \tag{6b}$$
$$K = PH^T P^{-1} \tag{6c}$$

where $F = \frac{\partial f_x}{\partial x}|_{\bar{x},u}$, $H = \frac{\partial h_x}{\partial x}|_{\bar{x},u}$, $\bar{x} \in \mathbb{R}^n$ is the bounded estimate of x from the globally stable NLO.

E. Sensor Fusion

The inertial measurements are fused with the leg odometry and the camera or LiDAR odometry. Decoupling the attitude from position and linear velocity offers a key benefit: the resulting dynamics become linear time-varying (LTV), ensuring inherent stability properties. In other words, the filter will not diverge within a finite timeframe. The KF has the following dynamics:

$$\dot{\hat{x}} = f_x + K(z - h_x) \tag{7a}$$
$$\dot{P} = FP + PF^T - KHP + Q \tag{7b}$$
$$K = PH^T R^{-1} \tag{7c}$$

where the state $x = [{x^n}^T {v^n}^T]^T \in \mathbb{R}^6$ is the position and velocity of the base, the input $u = (R_b^n f_i^b - g^n) \in \mathbb{R}^3$ is the acceleration of the base, and the vector z is the vector of measurements. The dimensions of z vary depending on the measurements. In the case of indoor experiments on Aliengo, in which the T265 camera is used as the only external sensor, z is dimension 9, because the T265 has pose and twist as outputs. This means that, in this case, $z = [R_b^n {\dot{x}_\ell^b}^T\ R_b^n {\dot{x}_c^b}^T\ R_b^n {x_c^b}^T]^T \in \mathbb{R}^9$ is given by the leg odometry (base velocity: $R_b^n {\dot{x}_\ell^b}^T$), and the camera velocity and position rotated in $\mathcal{N}$: $R_b^n {\dot{x}_c^b}^T$ and $R_b^n {x_c^b}^T$. On the FSC Dataset, on the other hand, since KISS-ICP has only the pose as output, the vector of measurements is $z = [R_b^n {\dot{x}_l^b}^T\ R_b^n {x_l^b}^T]^T \in \mathbb{R}^6$, where x_l^b is the position of the LiDAR in $\mathcal{B}$. The Kalman gain K is a matrix $\in \mathbb{R}^{6\times9}$ when all the measurements are available, or $\in \mathbb{R}^{6\times6}$ when the sensor velocity is not available. $P \in \mathbb{R}^{6\times6}$ is the covariance matrix, and $Q \in \mathbb{R}^{6\times6}$ is the process noise. The measurement noise covariance matrix is a diagonal block matrix, assuming that the measurements are uncorrelated:

$$R = \begin{bmatrix} R_1 & 0_3 & 0_3 \\ 0_3 & R_2 & 0_3 \\ 0_3 & 0_3 & R_3 \end{bmatrix} \quad \text{or} \quad R = \begin{bmatrix} R_1 & 0_3 \\ 0_3 & R_2 \end{bmatrix} \tag{8}$$

where $R_1 \in \mathbb{R}^{3\times3}$ is the covariance of the leg odometry, and its values are updated in case of slippage. $R_2 \in \mathbb{R}^{3\times3}$ is the covariance of the exteroceptive sensor velocity measurement (when available), and $R_3 \in \mathbb{R}^{3\times3}$ is the covariance of the exteroceptive sensor position measurement. Then

$$f_x = \begin{bmatrix} v^n \\ u \end{bmatrix} \quad \text{and} \quad F = \begin{bmatrix} 0_3 & I_3 \\ 0_3 & 0_3 \end{bmatrix} \tag{9}$$

where $I_3 \in \mathbb{R}^{3\times3}$ and $0_3 \in \mathbb{R}^{3\times3}$ are the identity matrix and null matrix, respectively. For the same reason previously explained, the matrix $H \in \mathbb{R}^{6\times9}$ or $H \in \mathbb{R}^{6\times6}$ is:

$$H = \begin{bmatrix} 0_3 & I_3 \\ 0_3 & I_3 \\ I_3 & 0_3 \end{bmatrix} \quad \text{or} \quad H = \begin{bmatrix} 0_3 & I_3 \\ I_3 & 0_3 \end{bmatrix} \tag{10}$$

The final structure of the state estimator is shown in Fig. 2.

We emphasize that to maintain efficient computation despite the slower arrival of exteroceptive measurements, we rely on internal measurements (IMU and joint states) for attitude estimation (Section II-D) and sensor fusion (Section II-E), applying corrections only when exteroceptive data becomes available.

Fig. 3: During the closed-loop experiment, Aliengo walked up and down the stairs, then on rocks and slippery terrain, repeating these tasks three times.

III. Experimental Results

In this section, we show the results obtained on two different robotic platforms: Aliengo on an online lab experiment (Section III-A), and ANYmal B300 on a pre-recorded outdoor dataset (Section III-B). As our state estimator operates as an odometry system, no loop closures (intended to recognize previously visited locations to reduce drift) have been executed on the estimated trajectory.

A. Closing the loop with the controller

The first test is a closed-loop experiment with Aliengo. The robot walked on difficult terrain, using a crawl gait, on a trajectory of approximately 60 m, and was commanded by a joystick. During this experiment, the robot completed three laps around the lab, walking up and down stairs, then on rocks and slippery terrain (Fig. 3).

The controller is the Model Predictive Controller (MPC) described in [26], that receives base pose and velocity inputs from MUSE, and gives torque commands to the joint PD controllers of the robot. The MPC runs at 100 Hz, and the PD controller at 1000 Hz. We ran the pipeline on an Intel NUC i7 with 32 GB of memory. Additionally, the IMU has an acquisition frequency of 1000 Hz, as well as leg kinematics, while the camera odometry runs at 200 Hz, and MUSE runs at 1000 Hz. The average execution time of each module within MUSE is 0.05 milliseconds, ensuring efficient processing and real-time state updates.

For MUSE, we used camera orientation $R_c^n \in \mathbb{R}^{3\times3}$ and IMU acceleration ($f_b \in \mathbb{R}^3$) as inputs for the XKF, while the linear velocity $\dot{x}_b^n \in \mathbb{R}^3$ from LO, linear position $x_c^n \in \mathbb{R}^3$ and velocity $\dot{x}_c^n \in \mathbb{R}^3$ from the camera, and the estimated orientation $R_b^n \in \mathbb{R}^{3\times3}$ from the XKF were used as inputs for SF (Sections II-D and II-E). The ground truth was obtained using a *Vicon* motion capture system.

We compared our results with those obtained using the T265's binocular visual-inertial tracking camera, commonly used as a standalone state estimator in other works (for instance in [27]). As shown in Figs. 4a and 4b, both position and orientation estimated by MUSE closely match the ground truth. Exteroceptive measurements significantly improved the robot's state estimation, particularly in correcting drift, which is common in the vertical direction (z-axis) when relying only on proprioception. Notably, camera odometry compensates for drift when walking on uneven terrain (Fig. 4a), making the benefits of the slip detection (SD) module less evident. However, the significance of SD is highlighted when using the proprioceptive-only variant of MUSE (P-MUSE). As shown in Fig. 4a, the SD module partially compensates for position drift during slippage, which arises when leg odometry becomes unreliable. In this experiment, "slippage" encompasses not only instances when the robot traverses the designated slippery patch, but also any event in which a leg slips on a rock.

Additionally, since the MPC controller requires the robot's linear velocity as feedback, Fig. 5 demonstrates that the estimated linear velocity closely aligns with the ground truth, enabling the robot to follow the desired trajectory.

Table I presents the Absolute Trajectory Error (ATE) and Relative Pose Error (RPE) [28] statistics over 1 m. These results confirm that the MUSE pipeline provides accurate position estimates. The ATE is comparable to that of the T265 camera, but MUSE operates at a higher frequency with lower RPE in both translation and orientation. The advantage of the SD module is evident in P-MUSE, where the ATE is higher when SD is not used. Furthermore, it is important to note that the yaw angle is accurately estimated even using only proprioceptive sensors, owing to the globally stable Attitude Observer, which ensures bounded orientation error and prevents filter divergence in a finite time frame.

B. FSC Dataset with ANYmal B300

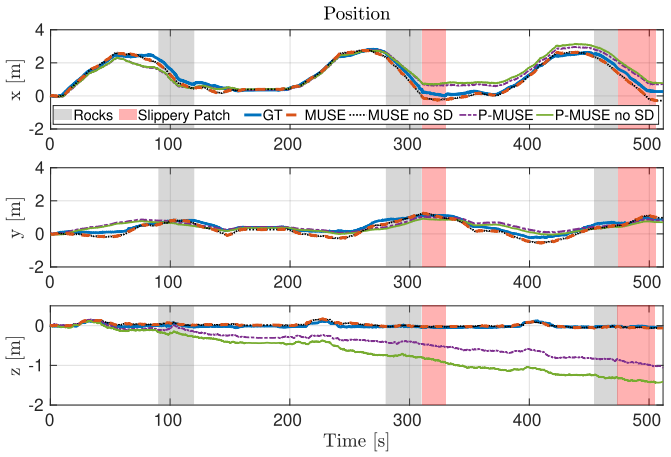
This section shows the results obtained by running MUSE on the Fire Service College Dataset [3]. The Fire Service College is a firefighting training facility located in the UK. One of the test areas represents a simulated, industrial oil rig with a total dimension of 32.5m $\times$ 42.5m. In the experiment, ANYmal trotted at 0.3 m/s, completing three loops before returning to the initial position, for a total of 240 m distance covered in 33 min. The environment was challenging due to the presence of standing water, oil residue, gravel, and mud. For this experiment, we show results using LiDAR as an exteroceptive sensor. The orientation from LiDAR odometry was used as an external measurement in the attitude estimate (Sec. II-D), and the position estimate was used as a measurement in the sensor fusion KF (Sec. II-E).

The ground truth (GT) trajectory was obtained with millimetric accuracy by combining the absolute sparse positions taken from a *Leica Total Station T16*, and a SLAM system based on ICP registration and IMU.

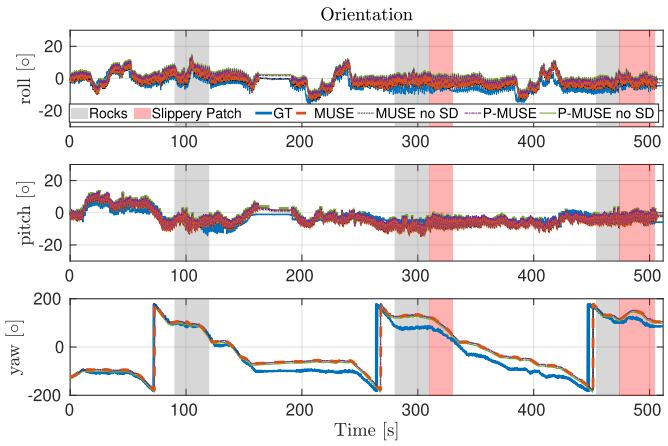
We computed the ATE and RPE over 10 m, on the entire trajectory (240 m). The performance in terms of ATE and

TABLE I: **Aliengo on uneven terrain**: ATE and RPE over 1 m ($\sim$ 60 m trajectory)

	T265	MUSE	MUSE no SD	P-MUSE	P-MUSE no SD
ATE [m]	0.24	0.24	0.25	0.57	0.67
RPE [m]	0.10	0.08	0.09	0.10	0.12
RPE [$\circ$]	0.35	0.25	0.26	0.27	0.27
Freq [Hz]	200	1000	1000	1000	1000



(a) Ground Truth (GT) vs. estimated position.



(b) Ground Truth (GT) vs. estimated orientation.

Fig. 4: **Aliengo on uneven terrain**: Comparison of position and orientation estimations between the GT and MUSE, MUSE without the SD module (MUSE with no SD), Proprioceptive MUSE (P-MUSE), and P-MUSE without the SD module (P-MUSE with no SD). The grey shaded areas indicate that the robot is walking on rocks, while the red ones indicate when the robot is walking on the slippery patch. The position plot (left) shows that the drift is higher when SD is not active.

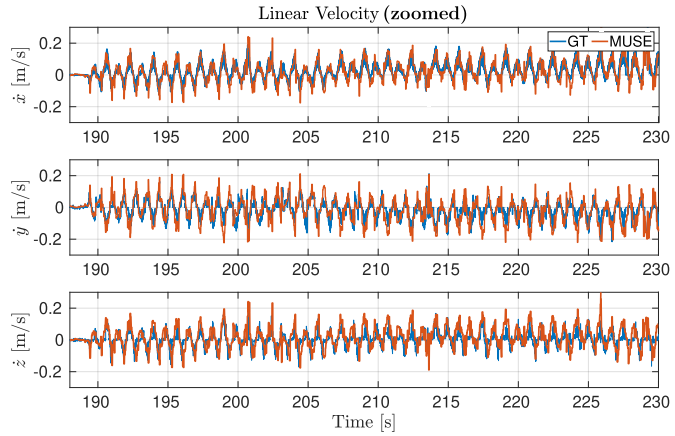


Fig. 5: **Aliengo on uneven terrain**: Ground Truth (GT) vs. Linear Velocity estimated by MUSE during the closed-loop experiment, zoom into the time interval [185-230] s.

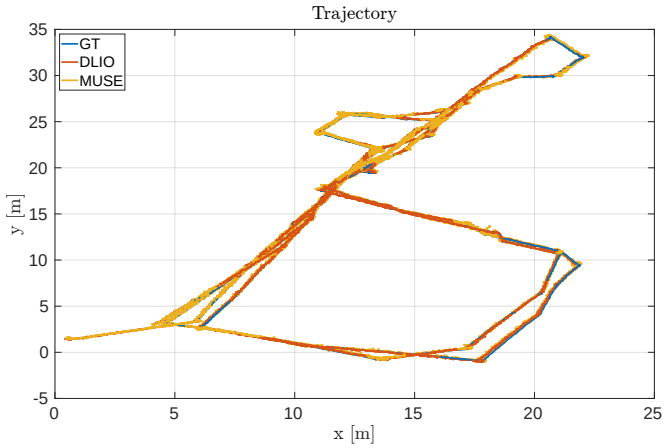
RPE was benchmarked against other state-of-the-art state estimators: DLIO [4], a LiDAR-inertial odometry algorithm, and three state estimators tailored for quadruped robots, Pronto [2], VILENS [3] and TSIF [5]. Pronto and VILENS fuse exteroceptive and proprioceptive measurements, while TSIF uses only proprioceptive data. All of these are odometry systems that do not utilize loop closures, with results shown in Table II.

Compared to DLIO, MUSE achieved a similar ATE and translational RPE, with a difference of only 3 cm and 2 cm, respectively. However, MUSE showed a lower rotational RPE and operated at a higher frequency. Specifically for this experiment, MUSE runs at 400 Hz, because leg kinematics and IMU run at 400 Hz, whereas DLIO operates at 100 Hz on average. While incorporating leg kinematics introduces slightly higher ATE due to noisy leg odometry, it makes the estimator more robust and faster in terms of frequency. Importantly, fusing different sensor modalities helps compensate for the limitations of individual sensors. Fusion does not always produce a more accurate estimate than using a single sensor, but it provides more robust estimation by allowing each sensor to compensate for potential failures of others. Furthermore, sensor fusion enables the estimator to reach higher frequencies by relying on high-frequency inputs.

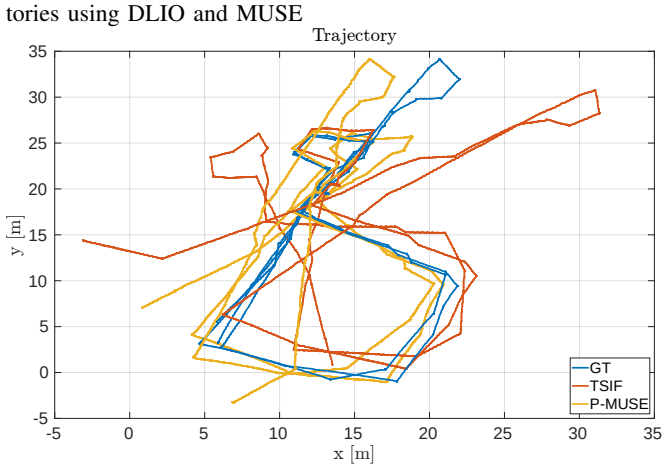
In comparison with Pronto and VILENS, MUSE is more accurate in terms of translational RPE, with improvements of 67.6% and 26.7%, respectively. The rotational RPE is similar to VILENS, although Pronto paper [2] does not provide this metric, nor does either system provide ATE data. When comparing proprioceptive-only state estimators, P-MUSE and TSIF, our algorithm proves to be more accurate in terms of ATE, reducing the mean error by nearly 50%. This is the most significant metric for evaluating overall trajectory discrepancy, reflecting global accuracy. The rotational RPE is similar between P-MUSE and TSIF, but P-MUSE shows a slightly higher translational RPE. This indicates that while TSIF captures short-term movements more precisely, small errors accumulate

TABLE II: **FSC Dataset**: ATE and RPE over 10 m ($\sim$ 240 m trajectory)

	DLIO	Pronto	VILENS	MUSE	MUSE no SD	TSIF	P-MUSE	P-MUSE no SD
ATE [m]	0.14	N.A	N.A.	0.17	0.18	4.40	2.38	2.57
RPE [m]	0.09	0.34	0.15	0.11	0.12	0.05	0.12	0.15
RPE [◦]	1.9	N.A.	1.14	1.78	1.85	1.96	1.93	1.96
Freq [Hz]	100	400	400	400	400	400	400	400



(a) FSC-Dataset: ground-truth trajectory (blue) vs. estimated trajectories using DLIO and MUSE



(b) FSC-Dataset: ground-truth trajectory (blue) vs. estimated trajectory using TSIF and P-MUSE

Fig. 6: **Trajectory of the FSC Dataset**: Comparison of the trajectory estimated using MUSE, P-MUSE, and two state-of-the-art state estimators: DLIO and TSIF.

over time, resulting in inferior global accuracy compared to P-MUSE. Figs. 6a and 6b provide visual comparisons between the ground truth and estimated trajectories. In Fig. 6a, we can see that DLIO and MUSE almost overlap with the ground truth, while in Fig. 6b, it is evident that our proprioceptive pipeline outperforms TSIF in terms of global accuracy.

IV. Discussion

MUSE is a modular state estimator for legged robots that combines proprioceptive and exteroceptive sensor data to provide accurate and robust state estimation across various environments. Its modularity enables the integration of multiple sensor modalities, compensating for individual sensor limitations and improving overall estimation accuracy. MUSE's real-time capability was demonstrated in the closed-loop experiment with the Aliengo robot, where it provided real-time feedback on linear velocity and orientation to the controller. Benchmarking on the FSC dataset further highlighted MUSE's superior global and local accuracy compared to other legged robot state estimators, demonstrating its effectiveness.

The results in Table II underscore MUSE's competitive performance among other estimators. Although DLIO may yield slightly lower ATE and RPE, MUSE's up-to-400-Hz operation remains crucial for high-frequency controllers such as MPC. Per well-established principles of cascaded controller design, the state estimation dynamics should converge significantly faster than the control loop to ensure nested-system stability [29], [30]. This makes MUSE's high bandwidth particularly valuable for legged robots, which require rapid and robust responses to dynamic changes in terrain, a necessity reinforced by the literature in [31]. Furthermore, the bandwidth consistency of MUSE aligns with legged robot state estimators such as Pronto, VILENS, and TSIF, underscoring the importance of high-frequency operation in this domain. MUSE's 400 Hz operation addresses these requirements by enabling quick, compliant reactions to disturbances while maintaining robust performance. Considering the trade-off between estimator frequency and error metrics, we argue that a higher bandwidth offers more substantial benefits for control stability than marginally improved error statistics. Future research could explore the correlation between estimator error and control stability to provide further insights.

Additionally, slip detection is a core component of MUSE, particularly for operation on slippery or uneven terrain. The effectiveness of this module, validated in our previous work [19], is central to the robustness of the overall pipeline. While threshold-based methods inherently involve a trade-off between false positives and false negatives, these were mitigated through parameter tuning and dynamic adjustments during locomotion. By incorporating slip detection, MUSE maintains accurate state estimation even when exteroceptive data is unavailable, offering more redundancy and robustness. This advantage is critical for dynamic and uneven terrains.

A. Limitations

Despite its strong performance, MUSE has some limitations that present opportunities for future work:

- Friction in the robot's joints occasionally affects the dynamics and reduces the accuracy of the contact estimation module, impacting both slip detection and leg odometry. Future work could address this by refining the contact estimation algorithm to better handle joint friction.
- While effective, the slip detection module does not capture all slippage events. Enhancing it to detect a broader range of scenarios, either through a probabilistic approach or machine learning techniques, is a promising avenue for improvement.
- MUSE assumes a fixed contact point at the center of the foot, which does not fully account for rolling motions or spherical foot geometries. This assumption can introduce errors in contact state estimation and ground reaction forces (GRFs). Developing methods to dynamically estimate and compensate for contact point variations could significantly improve robustness.

Addressing these limitations will enhance MUSE's applicability and reliability, broadening its deployment to a wider range of environments.

V. Conclusion

This paper presented MUSE, a state estimator designed to improve accuracy and real-time performance in quadruped robot navigation. By integrating camera and LiDAR odometry with foot-slip detection, MUSE fuses data from multiple sources, including IMU and joint encoders, to provide reliable pose and motion estimates, even in complex environments.

Ablation studies conducted on the Aliengo robot, along with benchmarking against other state-of-the-art estimators using the FSC Dataset of ANYmal B300 platform, validate the robustness and adaptability of MUSE across different scenarios. The results demonstrate the estimator's capability to handle dynamic and challenging conditions effectively, ensuring reliable performance during locomotion and navigation.

Future work includes refining the contact estimation algorithm to address inaccuracies, enhancing the slip detection module to capture a broader range of scenarios, and developing methods to account for dynamic contact point variations. Additionally, implementing camera and LiDAR odometry modules for dynamic environments will enable MUSE to handle challenges introduced by moving objects, people, or animals.

Acknowledgment

The authors would like to thank Professors Maurice Fallon (University of Oxford) and Marco Camurri (Free University of Bozen-Bolzano) for providing the FSC-Dataset.

References

[1] G. Fink and C. Semini, "Proprioceptive sensor fusion for quadruped robot state estimation," in *2020 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2020, pp. 10 914–10 920, DOI: 10.1109/IROS45743.2020.9341521.

[2] M. Camurri, M. Ramezani, S. Nobili, and M. Fallon, "Pronto: A multi-sensor state estimator for legged robots in real-world scenarios," *Front. Robot. AI*, vol. 7, 2020, DOI: 10.3389/frobt.2020.00068.

[3] D. Wisth, M. Camurri, and M. Fallon, "VILENS: Visual, inertial, lidar, and leg odometry for all-terrain legged robots," *IEEE Trans. Robot.*, 2022, DOI: 10.1109/TRO.2022.3193788.

[4] K. Chen, R. Nemiroff, and B. T. Lopez, "Direct lidar-inertial odometry: Lightweight LIO with continuous-time motion correction," in *2023 IEEE Int. Conf. Robot. Autom. (ICRA)*, 2023, pp. 3983–3989, DOI: 10.1109/ICRA48891.2023.10160508.

[5] M. Bloesch, M. Burri, H. Sommer, R. Siegwart, and M. Hutter, "The two-state implicit filter recursive estimation for mobile robots," *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, pp. 573–580, 2018, DOI: 10.1109/LRA.2017.2776340.

[6] R. Grandia, F. Jenelten, S. Yang, F. Farshidian, and M. Hutter, "Perceptive locomotion through nonlinear model-predictive control," *IEEE Trans. Robot.*, vol. 39, no. 5, pp. 3402–3421, 2023, DOI: 10.1109/TRO.2023.3275384.

[7] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, 2016, DOI: 10.1109/TRO.2016.2624754.

[8] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap SLAM," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1874–1890, 2021, DOI: 10.1109/tro.2021.3075644.

[9] M. Bloesch, M. Hutter, M. A. Hoepflinger, S. Leutenegger, C. Gehring, C. D. Remy, and R. Siegwart, "State estimation for legged robots: consistent fusion of leg kinematics and IMU," *Robotics*, vol. 17, pp. 17–24, Jul. 2013, DOI: 10.15607/RSS.2012.VIII.003.

[10] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided invariant extended Kalman filtering for robot state estimation," *Int. J. Robot. Res.*, vol. 39, no. 4, pp. 402–430, 2020, DOI: 10.1177/0278364919894385.

[11] S. Fahmi, G. Fink, and C. Semini, "On State Estimation for Legged Locomotion Over Soft Terrain," *IEEE Sens. Lett.*, vol. 5, no. 1, pp. 1–4, 2021, DOI: 10.1109/LSENS.2021.3049954.

[12] M. Bloesch, C. Gehring, P. Fankhauser, M. Hutter, M. A. Hoepflinger, and R. Siegwart, "State estimation for legged robots on unstable and slippery terrain," in *2013 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*. IEEE, 2013, pp. 6058–6064, DOI: 10.1109/IROS.2013.6697236.

[13] F. Jenelten, J. Hwangbo, F. Tresoldi, C. D. Bellicoso, and M. Hutter, "Dynamic locomotion on slippery ground," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 4170–4176, Oct. 2019, DOI: 10.1109/LRA.2019.2931284.

[14] D. Wisth, M. Camurri, and M. Fallon, "Preintegrated velocity bias estimation to overcome contact nonlinearities in legged robot odometry," in *2020 IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2020, pp. 392–398, DOI: 10.1109/ICRA40945.2020.9197214.

[15] H. M. S. Santana, J. C. V. Soares, Y. Nisticò, M. A. Meggiolaro, and C. Semini, "Proprioceptive state estimation for quadruped robots using invariant Kalman filtering and scale-variant robust cost functions," in *2024 IEEE-RAS Int. Conf. Humanoid Robots*, 2024, pp. 213–220, DOI: 10.1109/Humanoids58906.2024.10769911.

[16] S. Teng, M. W. Mueller, and K. Sreenath, "Legged robot state estimation in slippery environments using invariant extended Kalman filter with velocity update," *2021 IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 3104–3110, 2021, DOI: 10.1109/ICRA48506.2021.9561313.

[17] Y. Kim, B. Yu, E. M. Lee, J. Kim, H. Park, and H. Myung, "STEP: State estimator for legged robots using a preintegrated foot velocity factor," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4456–4463, 2022, DOI: 10.1109/LRA.2022.3150844.

[18] G. Ou, D. Li, and H. Li, "Leg-KILO: Robust kinematic-inertial-lidar odometry for dynamic legged robots," *IEEE Robot. Autom. Lett.*, vol. 9, no. 10, pp. 8194–8201, 2024, DOI: 10.1109/LRA.2024.3440730.

[19] Y. Nisticò, S. Fahmi, L. Pallottino, C. Semini, and G. Fink, "On slip detection for quadruped robots," *Sensors*, vol. 22, no. 8, 2022, DOI: 10.3390/s22082967.

[20] D. Wisth, M. Camurri, S. Das, and M. Fallon, "Unified multi-modal landmark tracking for tightly coupled lidar-visual-inertial odometry," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 1004–1011, 2021, DOI: 10.1109/LRA.2021.3056380.

[21] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, "KISS-ICP: In defense of point-to-point ICP – simple, accurate, and robust registration if done the right way," *IEEE Robot. Autom. Lett.*, vol. 8, no. 2, pp. 1029–1036, feb 2023, DOI: 10.1109/lra.2023.3236571.

[22] T. A. Johansen and T. I. Fossen, "The eXogenous Kalman filter (XKF)," *Int. J. Control*, vol. 90, no. 2, pp. 161–167, 2017, DOI: 10.1080/00207179.2016.1172390.

[23] H. F. Grip, T. I. Fossen, T. A. Johansen, and A. Saberi, "Globally exponentially stable attitude and gyro bias estimation with application to GNSS/INS integration," *Automatica*, vol. 51, pp. 158–166, 2015, DOI: 10.1016/j.automatica.2014.10.076.

[24] R. Mahony, J. Trumpf, and T. Hamel, "Observers for kinematic systems with symmetry," *IFAC Proceedings Volumes*, vol. 46, no. 23, pp. 617–633, 2013, DOI: 10.3182/20130904-3-FR-2041.00212.

[25] F. L. Markley and J. L. Crassidis, *Fundamentals of spacecraft attitude determination and control*. Springer, 2014, vol. 1286, DOI: 10.1007/978-1-4939-0802-8.

[26] L. Amatucci, G. Turrisi, A. Bratta, V. Barasuol, and C. Semini, "Accelerating model predictive control for legged robots through distributed optimization," in *2024 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2024, pp. 12 734–12 741, DOI: 10.1109/IROS58592.2024.10801676.

[27] J. Bayer and J. Faigl, "On autonomous spatial exploration with small hexapod walking robot using tracking camera Intel Realsense T265," in *2019 European Conference on Mobile Robots (ECMR)*, 2019, pp. 1–6, DOI: 10.1109/ECMR.2019.8870968.

[28] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual(-inertial) odometry," in *2018 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2018, pp. 7244–7251, DOI: 10.1109/IROS.2018.8593941.

[29] G. F. Franklin, J. D. Powell, A. Emami-Naeini, and J. D. Powell, *Feedback control of dynamic systems*. Prentice hall Upper Saddle River, 2002, vol. 4.

[30] M. Focchi, "Strategies to improve the impedance control performance of a quadruped robot," *Genoa: Istituto Italiano di Tecnologia*, 2013.

[31] S. Kleff, A. Meduri, R. Budhiraja, N. Mansard, and L. Righetti, "High-frequency nonlinear model predictive control of a manipulator," in *2021 IEEE Int. Conf. Robot. Autom. (ICRA)*, 2021, pp. 7330–7336, DOI: 10.1109/ICRA48506.2021.9560990.