2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

# State Estimation for Legged Robots on Unstable and Slippery Terrain

Michael Bloesch, Christian Gehring, Péter Fankhauser, Marco Hutter, Mark A. Hoepflinger, Roland Siegwart
Autonomous Systems Lab, ETH Zürich, Switzerland, bloeschm@ethz.ch

***Abstract*—This paper presents a state estimation approach for legged robots based on stochastic filtering. The key idea is to extract information from the kinematic constraints given through the intermittent contacts with the ground and to fuse this information with inertial measurements. To this end, we design an unscented Kalman filter based on a consistent formulation of the underlying stochastic model. To increase the robustness of the filter, an outliers rejection methodology is included into the update step. Furthermore, we present the nonlinear observability analysis of the system, where, by considering the special nature of 3D rotations, we obtain a relatively simple form of the corresponding observability matrix. This yields, that, except for the global position and the yaw angle, all states are in general observable. This also holds if only one foot is in contact with the ground. The presented filter is evaluated on a real quadruped robot trotting over an uneven and slippery terrain.**

## I. Introduction

As the research in legged robotic design and control is resulting in increasingly performing platforms, the aspect of state estimation and perception of such machines becomes more and more important as well. In order to be able to leave structured and controlled lab environments and go into more uncertain, rough and difficult terrain, it is indispensable to endow legged robots with precise state estimation and perception capabilities. Consequently, different research groups explored the integration of perception devices on legged platforms [14, 19, 21]. In the present paper however, focus is set on the proper extraction of information contained in the kinematics of the robot and obtained from inertial sensors. While for most legged robots such data is readily available from on-board sensor devices, it also represents a very valuable source of high-bandwidth information for state estimation. In our opinion, the exploitation of this information is a prerequisite for fast and elaborate control of legged robots in unstructured and difficult environments and represents an important foundation for the inclusion of further sensor modalities like vision or LIDAR.

Roston et al. [18] presented one of the earliest navigation system which extracts information from leg kinematics. By matching the foot positions between two consecutive timesteps they compute the incremental motion of the main body. Further, they introduce a slip detection method which relies on the invariance of the distance between feet that are in contact with the ground. Several groups extend this idea, e.g., Gassmann et al. [4] introduce fuzzy weights, based on different sensor measurements, in order to describe how well a certain foot is in contact with the ground and fuse the resulting legged odometer with GPS data. Along similar lines Lin et al. [13] present a leg strain-based odometer and use an inertial measurement unit (IMU) for handling flight phases of their hexapod robot. Again based on contact point matching, Görner et al. [5] present a legged odometer where joint torques are used to estimate roll and pitch of a fully actuated hexapod. A common drawback of these methods is that the associated legged odometer requires at least 3 non-colinear feet in contact with the ground.

Other approaches range from data-driven methods to model based observers. For example, using joint encoders, pressure sensors, and IMU data, Reinstein and Hoffmann [17] search for significant sensory based indicators in order to determine stride length. While it requires training of the state estimation for new locomotion scenarios, it enables the handling of cases with significant foot slippage. Based on a two dimensional dynamic model, Lebastard et al. [12] designed a high-order sliding-mode observer for estimating the 2D posture of their bipedal robot during a walking gait. Assuming planar spring-mass running, Gur and Saranli [6] propose a generic, model-based state estimation technique. The major issues of these approaches are the requirement for a precise dynamic system model and the possible restriction to a specific type of motion.

The detection of outliers in the context of legged robotic state estimation has only scarcely been studied. Most approaches use some additional force sensing on the foot level and compare desired and actual forces in order to detect slippage [11]. More recently, Okita and Sommer [16] considered slip events being anomalies which can be detected by employing appropriate filtering methods. In a simplified 2D stick-slip experiment they showed how to detect slippage using smoothed innovation in an Unscented Kalman filter (UKF) setup. Detecting anomalies or outliers in general filtering frameworks has been very widely analyzed. Ting et al. [22] as well as Agamennoni et al. [1] present outlier robust Kalman filtering by introducing more flexible noise models which allow the co-estimation of update noise parameters. Others investigated the use of non-Gaussian distributions which are less susceptible to outliers [20, 23].

The present paper is an extension to our prior work [2]. While following a similar overall approach, in the sense that accurate estimates of the full body pose are obtained by fusing information from an on-board IMU and kinematic measurements, the presented approach extends and improves different aspects of the previous methodology. By deriving velocity constraints from the feet that are in contact with

This research was supported by the Swiss National Science Foundation through the National Centre of Competence in Research Robotics.

the ground, simple measurement equations are obtained which reduce the size of the state and which are more suitable for slippage detection. Further, a robot-centric formulation of the state space is chosen in order to appropriately partition the filter states and avoid problems with unobservable states.

A thorough nonlinear observability analysis is provided for the presented filter. A novel method for handling rotational states is presented which significantly simplifies the analytical evaluation of the unobservable subspace and corresponding rank deficiency. Based on the nonlinear observability analysis of Hermann and Krener [7] we present a method for handling states which are elements of the special orthogonal group $SO(3)$ by exploiting the local homeomorphism to 3D real vector space. With this we show, that up to some singular robot motions, all states of the robotic platform are observable except for the yaw angle around the gravity axis and the global position (which are not essential for the local control of the robot). This also holds for the case where only one leg is in contact with the ground and thus the state estimator can be applied for dynamic locomotion as well.

The presented filter is implemented and evaluated on our quadruped robot StarlETH [9]. We show results from experiments where the robot is trotting over uneven and labile terrain with occurring foot slippage. For all experiments the control of the robot fully relies on the estimates from the filter. No previous information on the shape of the terrain is required and the external motion capture system is only used for groundtruth comparison.

The paper is structured as follows. In Section II we start with some brief prerequisites. Subsequently, Section III discusses the specific filter setup and the outliers detection. In Section IV the observability analysis is performed and in Section V the experimental setup and obtained results are presented.

## II. Prerequisites

For better readability we give a short overview on the employed notations and conventions. The coordinates, expressed in a frame $A$, of a vector from a point $P$ to a point $Q$ are denoted by ${}_A\boldsymbol{r}_{PQ}$. If $B$ is a second coordinate frame, then $\boldsymbol{C}_{BA}$ maps the coordinates expressed in $A$ to the corresponding coordinates in $B$. The rotation between both frames is generally parametrized by the unit quaternion $\boldsymbol{q}_{BA}$. Throughout the paper, we add a subscript $k$ to a quantity $v$, if we want to talk about its value at a time $t_k$, i.e., $v_k = v(t_k)$. Two coordinate frames are of interest: the world fixed coordinate frame $W$ and the main body frame $B$.

In a filter setup, mathematical operations are employed which are not defined for 3D rotations (especially addition and differentiation). In order to handle this issue we exploit the homeomorphism between the 3D manifold $SO(3)$ and 3D vector spaces. For a more thorough discussion on the topic please refer to the work of Hertzberg et al. [8]. In short we use the exponential mapping, $\boldsymbol{q} = \exp(\boldsymbol{\theta})$, between a 3D rotation vector, $\boldsymbol{\theta} \in \mathbb{R}^3$, and the corresponding quaternion $\boldsymbol{q} \in SO(3)$. This mapping is surjective and thus an inverse exists, $\boldsymbol{\theta} = \log(\boldsymbol{q})$, which is called the logarithm. These maps are used for introducing the boxplus and boxminus operators:

$$\begin{aligned} \boxplus &: SO(3) \times \mathbb{R}^3 \to SO(3), && (1)\\ &\quad \boldsymbol{q}, \boldsymbol{\theta} \mapsto \exp(\boldsymbol{\theta}) \otimes \boldsymbol{q}, \\ \boxminus &: SO(3) \times SO(3) \to \mathbb{R}^3, && (2)\\ &\quad \boldsymbol{q}_1, \boldsymbol{q}_2 \mapsto log(\boldsymbol{q}_1 \otimes \boldsymbol{q}_2^{-1}), \end{aligned}$$

where the boxminus operator expresses the difference between two quaternions by returning the error rotation vector between both, and where the boxplus operator applies a small rotation, expressed by a rotation vector, onto a unit quaternion.

Based on the above definitions we introduce special differentials on unit quaternions. Given a function $\boldsymbol{q} : \boldsymbol{x} \mapsto \boldsymbol{q}(\boldsymbol{x})$ which maps from some real vector space $\mathbb{R}^N$ to the set of unit quaternions, we define the differential

$$\left(\frac{\partial \boldsymbol{q}}{\partial \boldsymbol{x}}\right)_i := \lim_{\epsilon \to 0} \frac{\boldsymbol{q}(\boldsymbol{x} + \epsilon \boldsymbol{e}_i) \boxminus \boldsymbol{q}(\boldsymbol{x})}{\epsilon}, \quad i = 1, \ldots, N, \tag{3}$$

and if $\boldsymbol{f} : \boldsymbol{q} \mapsto \boldsymbol{f}(\boldsymbol{q})$ is a function which maps from the set of unit quaternions to some real vector space we define

$$\left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{q}}\right)_i := \lim_{\epsilon \to 0} \frac{\boldsymbol{f}(\boldsymbol{q} \boxplus \epsilon \boldsymbol{e}_i) - \boldsymbol{f}(\boldsymbol{q})}{\epsilon}, \quad i = 1, \ldots, 3. \tag{4}$$

Let $\boldsymbol{C}(\cdot)$ be the mapping between unit quaternions and corresponding rotation matrices, then following identities hold:

$$\begin{aligned} \partial / \partial \boldsymbol{q} \left(\boldsymbol{C}(\boldsymbol{q}) \boldsymbol{v}\right) &= -\left(\boldsymbol{C}(\boldsymbol{q}) \boldsymbol{v}\right)^{\times}, && (5)\\ \partial / \partial \boldsymbol{q} \left(\boldsymbol{q}^{-1}\right) &= -\boldsymbol{C}^T(\boldsymbol{q}), && (6)\\ \partial / \partial \boldsymbol{q}_1 \left(\boldsymbol{q}_1 \otimes \boldsymbol{q}_2\right) &= \boldsymbol{I}, && (7)\\ \partial / \partial \boldsymbol{q}_2 \left(\boldsymbol{q}_1 \otimes \boldsymbol{q}_2\right) &= \boldsymbol{C}(\boldsymbol{q}_1), && (8)\\ \partial / \partial \boldsymbol{q} \left(\log(\boldsymbol{q})\right) &= \boldsymbol{\Gamma}_1^{-1}(\log(\boldsymbol{q})), && (9)\\ \partial / \partial t \left(\boldsymbol{q}_{BA}(t)\right) &= {}_B\boldsymbol{\omega}_{BA}(t), && (10) \end{aligned}$$

where the subscript $^{\times}$ is used to denote the skew-symmetric matrix of a vector and where $\boldsymbol{\omega}_{BA}$ is the rotational rate vector of frame $B$ with respect to frame $A$. We also made use of the auxiliary quantity $\boldsymbol{\Gamma}_n(\boldsymbol{\theta}) := \sum_{i=0}^{\infty} \frac{\boldsymbol{\theta}^{\times^i}}{(i+n)!}$. It draws on the series expansion of the matrix exponential and, consequently, $\boldsymbol{\Gamma}_0(\boldsymbol{\theta})$ represents the rotation matrix corresponding to the rotation vector $\boldsymbol{\theta}$. There exists a closed form expression for $\boldsymbol{\Gamma}_n$ that can be efficiently numerically evaluated (similar to Rodrigues' rotation formula). The above special differentials strongly simplify the handling of analytical Jacobians, especially in the context of nonlinear observability analysis including rotational quantities. It can be proven that the chain rule is valid. Please note that the formulation of the identities can vary slightly depending on the employed conventions.

## III. Filter Setup

### A. Filter States and Measurement Models

The overall structure of a filter strongly depends on the choice of the underlying filter states. In our case we chose a set of robot-centric states in order to describe the motion of the robot's main body. The state includes the position of the world frame with respect to the body frame, ${}_B\boldsymbol{r}_{BW}$, the

negative velocity of the main body expressed in the frame $B$, $-_B\boldsymbol{v}_B$, the attitude of the main body parametrized by $\boldsymbol{q}_{WB}$, as well as the bias terms of the accelerometer and gyroscope, ${}_B\boldsymbol{b}_f$ and ${}_B\boldsymbol{b}_\omega$. In short, the state $\boldsymbol{x}$ will be defined as

$$\boldsymbol{x} := \left(\boldsymbol{r}, \boldsymbol{v}, \boldsymbol{q}, \boldsymbol{c}, \boldsymbol{d}\right) \tag{11}$$

$$:= \left({}_B\boldsymbol{r}_{BW}, -_B\boldsymbol{v}_B, \boldsymbol{q}_{WB}, {}_B\boldsymbol{b}_f, {}_B\boldsymbol{b}_\omega\right). \tag{12}$$

Building on this, process and measurement equations need to be formulated which properly capture the behavior and uncertainties of the underlying system. The choice of the models is a trade-off between simplicity and accuracy, whereby all stochastic quantities will be modeled as continuous white Gaussian noise or as discrete Gaussian noise processes. This is in accord with the prerequisites of most filtering methods and deviation from the real system can be handled to a certain extent by increasing the corresponding covariance matrices.

The proper acceleration measurement $\tilde{\boldsymbol{f}}$ and the rotational rate measurement $\tilde{\boldsymbol{\omega}}$ of the IMU are assumed to be affected by additive white Gaussian noise, $\boldsymbol{n}_f$ and $\boldsymbol{n}_\omega$, as well as by the additive biases $\boldsymbol{c}$ and $\boldsymbol{d}$:

$$\tilde{\boldsymbol{f}} = \boldsymbol{f} + \boldsymbol{c} + \boldsymbol{n}_f, \tag{13}$$

$$\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega} + \boldsymbol{d} + \boldsymbol{n}_\omega. \tag{14}$$

Both quantities do not directly depend on the states of the filter but rather measure the corresponding rates. Considering

$$\boldsymbol{f} = \boldsymbol{C}(\boldsymbol{q}_{BW})\left({}_W\dot{\boldsymbol{v}}_B - \boldsymbol{g}\right), \tag{15}$$

$$\boldsymbol{\omega} = \dot{\boldsymbol{q}}_{BW}, \tag{16}$$

where $\boldsymbol{g}$ is the gravity vector in $W$, the IMU measurements will later be directly included into the prediction step of the filter. For simplicity, we assume that all inertial measurements are obtained with respect to the body frame $B$.

Encoders in each of the robot's joints provide access to the corresponding angular measurements $\tilde{\boldsymbol{\alpha}}$ and their derivatives $\dot{\tilde{\boldsymbol{\alpha}}}$. Considering the forward kinematics ${}_B\boldsymbol{r}_{BF_i}(\tilde{\boldsymbol{\alpha}}) = \boldsymbol{s}_i(\tilde{\boldsymbol{\alpha}})$, we can compute the absolute location of the $i^{\text{th}}$ foot $F_i$:

$${}_W\boldsymbol{r}_{WF_i} = {}_W\boldsymbol{r}_{WB} + \boldsymbol{C}_{WBB}\boldsymbol{r}_{BF_i}(\tilde{\boldsymbol{\alpha}}) \tag{17}$$

$$= \boldsymbol{C}(\boldsymbol{q})(\boldsymbol{s}_i(\tilde{\boldsymbol{\alpha}}) - \boldsymbol{r}). \tag{18}$$

If foot $i$ is in contact with the ground and assuming that it remains stationary with respect to the world frame $W$, the differentiation of the above kinematic identity yields

$$0 = -\boldsymbol{v} + \boldsymbol{\omega}^\times \boldsymbol{s}_i(\tilde{\boldsymbol{\alpha}}) + \boldsymbol{J}_i(\tilde{\boldsymbol{\alpha}})\dot{\tilde{\boldsymbol{\alpha}}} + \boldsymbol{n}_s, \tag{19}$$

where $\boldsymbol{J}_i(\tilde{\boldsymbol{\alpha}}) = \frac{\partial}{\partial\tilde{\boldsymbol{\alpha}}}\boldsymbol{s}_i(\tilde{\boldsymbol{\alpha}})$ is the Jacobian of the forward kinematics. The discrete Gaussian noise term $\boldsymbol{n}_s \sim \mathcal{N}(0, \boldsymbol{R}_s)$ incorporates different sources of noise, including errors from the encoder measurements as well as imprecise kinematic modeling. This is mainly done because the noise on the encoder measurements causes only a minor part of the full measurement noise of (19), where modeling errors and foot contact effects are more important. In order to avoid the complex modeling of such effects, the covariance matrix $\boldsymbol{R}_s$ incorporates all stochastic errors together and represents one of the main tuning parameter of the filter.

As mentioned earlier, the IMU measurements are linked to the rates of the filter states and are thus included into the continuous time differential equations of the prediction model. Using equation (15) and (16) and carefully evaluating the total derivatives we can write:

$$\dot{\boldsymbol{r}} = -\left(\tilde{\boldsymbol{\omega}} - \boldsymbol{d} - \boldsymbol{n}_\omega\right)^\times \boldsymbol{r} + \boldsymbol{v}, \tag{20}$$

$$\dot{\boldsymbol{v}} = -\left(\tilde{\boldsymbol{\omega}} - \boldsymbol{d} - \boldsymbol{n}_\omega\right)^\times \boldsymbol{v} - \tilde{\boldsymbol{f}} + \boldsymbol{c} + \boldsymbol{n}_f - \boldsymbol{C}^T(\boldsymbol{q})\boldsymbol{g}, \tag{21}$$

$$\dot{\boldsymbol{q}} = \boldsymbol{C}(\boldsymbol{q})(\tilde{\boldsymbol{\omega}} - \boldsymbol{d} - \boldsymbol{n}_\omega), \tag{22}$$

$$\dot{\boldsymbol{c}} = \boldsymbol{n}_c, \tag{23}$$

$$\dot{\boldsymbol{d}} = \boldsymbol{n}_d. \tag{24}$$

The additional continuous white Gaussian noise processes $\boldsymbol{n}_c$ and $\boldsymbol{n}_d$ model a certain drift affecting the bias terms. For all white Gaussian noise processes, the corresponding covariance parameters, $\boldsymbol{R}_f$, $\boldsymbol{R}_\omega$, $\boldsymbol{R}_c$ and $\boldsymbol{R}_d$ describe the magnitude of the noise. The covariance parameters can be identified by considering the Allan plots of the IMU measurements [3].

### *B. Unscented Kalman Filter*

The different measurements are fused within an unscented Kalman filter framework. While the resulting computational costs are slightly higher than for a corresponding extended Kalman filter, the UKF is in general more robust against nonlinearities. However, for the case at hand, our choice was mainly motivated by the simplicity of handling correlated noise between prediction and correction step. The correlation can best be seen by considering the discretized filter equations.

Discretization of the stochastic differential equations (SDE) (20)-(24) is a difficult problem and is, in general, not analytically solvable without approximation. The most common approach is to linearize the equations and to integrate the linear SDE. Here, we discretize the deterministic and stochastic part of the SDE separately. This allows the analytical solution of the corresponding system of deterministic differential equations and thus keeps our rotational state in the 3D manifold $SO(3)$. Using the abbreviation $\Delta t_k = t_k - t_{k-1}$ and applying the method of variation of parameters we obtain:

$$\boldsymbol{r}_k = \boldsymbol{\Gamma}_{0,k}^T\Big(\boldsymbol{r}_{k-1} + \Delta t_k\boldsymbol{v}_{k-1} - \frac{\Delta t_k^2}{2}\Big(2\boldsymbol{\Gamma}_{2,k}(\tilde{\boldsymbol{f}}_k - \boldsymbol{c}_{k-1} - \boldsymbol{n}_{f,k}) + \boldsymbol{C}(\boldsymbol{q}_{k-1})\boldsymbol{g}\Big)\Big) + \boldsymbol{n}_{r,k}, \tag{25}$$

$$\boldsymbol{v}_k = \boldsymbol{\Gamma}_{0,k}^T\Big(\boldsymbol{v}_{k-1} - \Delta t_k\Big(\boldsymbol{\Gamma}_{1,k}(\tilde{\boldsymbol{f}}_k - \boldsymbol{c}_{k-1} - \boldsymbol{n}_{f,k}) + \boldsymbol{C}(\boldsymbol{q}_{k-1})\boldsymbol{g}\Big)\Big), \tag{26}$$

$$\boldsymbol{q}_k = \boldsymbol{q}_{k-1} \otimes \exp\Big(\Delta t_k(\tilde{\boldsymbol{\omega}}_k - \boldsymbol{d}_{k-1} - \boldsymbol{n}_{\omega,k})\Big), \tag{27}$$

$$\boldsymbol{c}_k = \boldsymbol{c}_{k-1} + \Delta t_k\boldsymbol{n}_{c,k}, \tag{28}$$

$$\boldsymbol{d}_k = \boldsymbol{d}_{k-1} + \Delta t_k\boldsymbol{n}_{d,k}, \tag{29}$$

with

$$\boldsymbol{\Gamma}_{n,k} = \boldsymbol{\Gamma}_n\Big(\Delta t_k(\tilde{\boldsymbol{\omega}}_k - \boldsymbol{d}_{k-1} - \boldsymbol{n}_{\omega,k})\Big). \tag{30}$$

The various discretized noise quantities are distributed with $\mathcal{N}(0, \boldsymbol{R}/\Delta t_k)$ where $\boldsymbol{R}$ is the corresponding continuous covariance parameter. The new discrete Gaussian noise term $\boldsymbol{n}_{r,k}$ is used to model errors that occurred during discretization.

While equations (25)-(29) are used for the prediction of the filter, the update step is based on the kinematic identity (19). This is applied to every leg $i$ that is in contact:

$$\begin{aligned} 0 = &- \boldsymbol{v}_k + (\tilde{\boldsymbol{\omega}}_k - \boldsymbol{d}_{k-1} - \boldsymbol{n}_{\omega,k})^{\times} \boldsymbol{s}_i(\tilde{\boldsymbol{\alpha}}_k) \\ &+ \boldsymbol{J}_i(\tilde{\boldsymbol{\alpha}}_k)\dot{\tilde{\boldsymbol{\alpha}}}_k + \boldsymbol{n}_{s,k}. \end{aligned} \tag{31}$$

The recurrence of the gyroscope measurement noise $\boldsymbol{n}_{\omega,k}$ in the update equation correlates the noise between prediction and update step. In an UKF setup this can be handled very easily. The basic outline of the filter looks as follows. Given the *a-posteriori* estimate $\boldsymbol{x}_{k-1}$ and its covariance matrix $\boldsymbol{P}_{k-1}$ at time $t_{k-1}$, sigma points are sampled in such a manner that they represent the joint distribution of the state estimate and all noise quantities. This results in a set of sigma points of the following form:

$$\mathcal{X}^i_{k-1} = \left(\boldsymbol{x}^i_{k-1}, \boldsymbol{n}^i_{r,k}, \boldsymbol{n}^i_{f,k}, \boldsymbol{n}^i_{\omega,k}, \boldsymbol{n}^i_{c,k}, \boldsymbol{n}^i_{d,k}, \boldsymbol{n}^i_{s,k}\right). \tag{32}$$

Whereby using the same sampled rotational rate noise during prediction and update automatically handles the stochastic correlation between both steps. For a more detailed discussion on the employed UKF please refer to [10]. Also, please note that throughout the filter the boxplus (1) and boxminus (2) operators have to be employed where appropriate.

### C. Outliers Detection

Kalman filters have the drawback that they can be very sensitive to outliers. While outliers are often caused by non-modeled effects or other anomalies, their appearance is in most cases only difficultly predictable and the corresponding observations draw generally from a significantly different probability distribution. The sensitivity is caused by the light-tailed underlying Gaussian distribution which leads to the minimization of squared error terms. In order to handle outliers caused by foot slippage we propose to employ a simple thresholding based on the Mahalanobis distance of the innovation. This employs the predicted covariance of the innovation and classifies a measurement as an outlier if the Mahalanobis distance exceeds a certain threshold. This has the drawback that the threshold needs to be hand-tuned, however, if it is appropriately chosen this leads to near-optimal filtering [22].

Let $\boldsymbol{y}_{i,k}$ be the innovation induced by the kinematic constraints of the $i^{\text{th}}$ leg at timestep $k$ (31) and $\boldsymbol{S}_{i,k}$ the corresponding predicted covariance matrix. We classify the observation as an outlier if the Mahalanobis distance is larger than a certain threshold parameter $p$, i.e., if $\boldsymbol{y}^T_{i,k}\boldsymbol{S}^{-1}_{i,k}\boldsymbol{y}_{i,k} > p$ is met. Under the assumption of Gaussian distribution the left hand side of the inequality will be $\chi^2$ distributed with 3 degrees of freedom. In our case the threshold $p = 16.27$ was chosen in order to obtain a rejection rate of $0.1\%$ for inliers. If the above threshold is exceeded, the kinematic constraints are ignored and not taken into account during the update step (like for all legs that are not currently in contact with the ground). An analogous approach was employed by Mirzaei et al. [15] for rejecting visual feature measurements within a Kalman filter based IMU-camera calibration.

## IV. Nonlinear Observability Analysis

Similarly to Hermann and Krener [7], we employ the notion of locally weakly observability which qualifies whether each point of a system can be instantaneously distinguished from its neighbors. As a slight technical difference we consider our system to have no external control input and interpret the rotational rate as well as the proper acceleration as system parameters. The subsequent nonlinear observability analysis should reflect the observability characteristics of the system in dependence of those parameters.

Lets consider the following state-space representation of a smooth nonlinear system:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}), \tag{33}$$

$$\boldsymbol{z} = \boldsymbol{h}(\boldsymbol{x}), \tag{34}$$

with process function $\boldsymbol{f}$ and measurement function $\boldsymbol{h}$. For a given state $\boldsymbol{x}$ and input parameters $\boldsymbol{u}$ we can now evaluate the observability matrix

$$\boldsymbol{\mathcal{O}}(\boldsymbol{x}, \boldsymbol{u}) = \begin{bmatrix} \nabla \mathcal{L}^0_{\boldsymbol{f}} \boldsymbol{h}(\boldsymbol{x}) \\ \nabla \mathcal{L}^1_{\boldsymbol{f}} \boldsymbol{h}(\boldsymbol{x}) \\ \vdots \end{bmatrix}, \tag{35}$$

based on the gradient operator $\nabla$ and Lie derivatives [7]. Informally it describes the effect of infinitesimal state perturbations $\boldsymbol{\delta x}$ on the instantaneous measurement $\boldsymbol{z}$ and its derivatives:

$$\begin{bmatrix} \boldsymbol{\delta z} \\ \boldsymbol{\delta \dot{z}} \\ \vdots \end{bmatrix} = \boldsymbol{\mathcal{O}}(\boldsymbol{x}, \boldsymbol{u}) \boldsymbol{\delta x}. \tag{36}$$

Perturbations $\boldsymbol{\delta x}$ which do not cause any change in the corresponding measurements are intrinsically not observable. Consequently, the nullspace of the observability matrix $\boldsymbol{\mathcal{O}}(\boldsymbol{x}, \boldsymbol{u})$ is equivalent to the unobservable subspace of the system at a state $\boldsymbol{x}$ and for a given input parameter $\boldsymbol{u}$.

The novelty in the presented nonlinear observability analysis is the seamless integration of rotational states into the observability analysis by means of the special derivatives introduced in equations (3) and (4). Using the identities (5)-(10) and applying the chain rule, the Lie derivatives can be easily evaluated, whereby the entries in the observability matrix corresponding to 3D rotational quantities will exhibit the proper number of dimension (which should be 3) and accurately reflect the observability characteristics of the system. This is best explained at hand of a concrete example: for the filter presented in this paper the sequence of Lie derivatives and corresponding gradients together with the observability matrix will be evaluated for the case of a single foot contact with the ground. For the sake of readability the indexes are omitted where possible and the noise terms are left out (they do not influence the observability analysis). In short we will also use

$\tilde{\boldsymbol{s}} = \boldsymbol{s}(\tilde{\boldsymbol{\alpha}})$, $\hat{\boldsymbol{\omega}} = \tilde{\boldsymbol{\omega}} - \boldsymbol{d}$, $\hat{\boldsymbol{f}} = \tilde{\boldsymbol{f}} - \boldsymbol{c}$ and $\boldsymbol{C} = \boldsymbol{C}(\boldsymbol{q})$. The process function (equations (20)-(24)) can be written as:

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) = \begin{pmatrix} -\hat{\boldsymbol{\omega}}^{\times}\boldsymbol{r} + \boldsymbol{v} \\ -\hat{\boldsymbol{\omega}}^{\times}\boldsymbol{v} - \hat{\boldsymbol{f}} - \boldsymbol{C}^T\boldsymbol{g} \\ \boldsymbol{C}\hat{\boldsymbol{\omega}} \\ 0 \\ 0 \end{pmatrix}. \tag{37}$$

The sequence of Lie derivatives and corresponding gradients can be evaluated to:

$$\mathcal{L}^0_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}) = -\boldsymbol{v} + (\tilde{\boldsymbol{\omega}} - \boldsymbol{d})^{\times}\tilde{\boldsymbol{s}} + \boldsymbol{J}(\tilde{\boldsymbol{\alpha}})\dot{\tilde{\boldsymbol{\alpha}}}, \tag{38}$$

$$\nabla\mathcal{L}^0_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}) = \begin{bmatrix} 0 & -\boldsymbol{I} & 0 & 0 & \tilde{\boldsymbol{s}}^{\times} \end{bmatrix}, \tag{39}$$

$$\mathcal{L}^1_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}) = \hat{\boldsymbol{\omega}}^{\times}\boldsymbol{v} + \hat{\boldsymbol{f}} + \boldsymbol{C}^T\boldsymbol{g}, \tag{40}$$

$$\nabla\mathcal{L}^1_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}) = \begin{bmatrix} 0 & \hat{\boldsymbol{\omega}}^{\times} & \boldsymbol{C}^T\boldsymbol{g}^{\times} & -\boldsymbol{I} & \boldsymbol{v}^{\times} \end{bmatrix}, \tag{41}$$

$$\vdots$$

$$\mathcal{L}^n_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}) = -\hat{\boldsymbol{\omega}}^{\times^n}\boldsymbol{v} - \hat{\boldsymbol{\omega}}^{\times^{n-1}}\hat{\boldsymbol{f}} - n\hat{\boldsymbol{\omega}}^{\times^{n-1}}\boldsymbol{C}^T\boldsymbol{g}, \tag{42}$$

$$\nabla\mathcal{L}^n_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}) = \Big[0 \quad -\hat{\boldsymbol{\omega}}^{\times^n} \quad -n\hat{\boldsymbol{\omega}}^{\times^{n-1}}\boldsymbol{C}^T\boldsymbol{g}^{\times} \quad \hat{\boldsymbol{\omega}}^{\times^{n-1}} \quad \partial\mathcal{L}^n_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x})/\partial\boldsymbol{d}\Big]. \tag{43}$$

With this, the Observability matrix (35) can be constructed and simplified in order to obtain the following term:

$$\boldsymbol{\mathcal{O}}(\boldsymbol{x}, \boldsymbol{u}) = \begin{bmatrix} 0 & -\boldsymbol{I} & 0 & 0 & \tilde{\boldsymbol{s}}^{\times} \\ 0 & 0 & \boldsymbol{C}^T\boldsymbol{g}^{\times} & -\boldsymbol{I} & \boldsymbol{v}^{\times} + \hat{\boldsymbol{\omega}}^{\times}\tilde{\boldsymbol{s}}^{\times} \\ 0 & 0 & -\hat{\boldsymbol{\omega}}^{\times}\boldsymbol{C}^T\boldsymbol{g}^{\times} & 0 & (\hat{\boldsymbol{\omega}}^{\times}\boldsymbol{v} + \hat{\boldsymbol{f}} + 2\boldsymbol{C}^T\boldsymbol{g})^{\times} \\ 0 & 0 & 0 & 0 & (\hat{\boldsymbol{\omega}}^{\times}\boldsymbol{v} + \hat{\boldsymbol{f}} + \boldsymbol{C}^T\boldsymbol{g})^{\times}\hat{\boldsymbol{\omega}}^{\times} \\ 0 & 0 & 0 & 0 & (\hat{\boldsymbol{\omega}}^{\times}\boldsymbol{v} + \hat{\boldsymbol{f}} + \boldsymbol{C}^T\boldsymbol{g})^{\times}\hat{\boldsymbol{\omega}}^{\times 2} \\ 0 & 0 & 0 & 0 & (\hat{\boldsymbol{\omega}}^{\times}\boldsymbol{g})^{\times} \\ 0 & 0 & 0 & 0 & (\hat{\boldsymbol{\omega}}^{\times 2}\boldsymbol{g})^{\times} \end{bmatrix}$$

In this example, the input parameter $\boldsymbol{u}$ is given by the rotational rate $\hat{\boldsymbol{\omega}}$ and the proper acceleration $\hat{\boldsymbol{f}}$ which describe the motion of the robot main body. Our goal is to obtain the observability characteristic in dependence of those parameters, rather than asking the question whether there exists some input parameter which make our system observable.

As mentioned above, the nullspace of the observability matrix corresponds to the directions of disturbances which can not be observed at the output of the system. Up to a few singular cases, the rank of the nullspace is 4 and is spanned by the following matrix:

$$\boldsymbol{\mathcal{U}}(\boldsymbol{x}, \boldsymbol{u}) = \begin{bmatrix} \boldsymbol{I} & 0 & 0 & 0 & 0 \\ 0 & 0 & \boldsymbol{g}^T & 0 & 0 \end{bmatrix}^T, \tag{44}$$

where the first row describes unobservable disturbances on the robot position and where the second row represents rotation around the gravity axis (yaw angle). The emergence of those unobservable modes could have been predicted as we do not use any global positioning system. However, there are singular cases where more directions become unobservable. Those can also be evaluated analytically based on the observability matrix. In the scope of this paper the singular cases are listed in table I together with a brief discussion (if a cross is set the above equality is fulfilled). As can be observed, the rank loss depends on the relation between gravity vector and rotational rate vector. If there is no rotational motion in the system, the filter can not distinguish between inclination angles (pitch and roll) and a bias on the proper acceleration measurement. Furthermore it will not be able to estimate the gyroscope bias around the gravity and we thus get a total rank deficiency of 3 for this case.

| $\hat{\boldsymbol{\omega}} = 0$ | $\hat{\boldsymbol{\omega}} \perp \boldsymbol{C}^T\boldsymbol{g}$ | $\hat{\boldsymbol{\omega}} \parallel \boldsymbol{C}^T\boldsymbol{g}$ | Rank deficiency |
|---|---|---|---|
| x | x | x | 5 $(\hat{\boldsymbol{f}} = -2\boldsymbol{C}^T\boldsymbol{g})$<br>3 $(\hat{\boldsymbol{f}} \neq -2\boldsymbol{C}^T\boldsymbol{g})$ |
| | x | | 1 |
| | | x | 1 |
| | | | 0 |

TABLE I
RANK DEFICIENCY IN DEPENDENCE OF INPUT PARAMETERS.

In a less intuitive way the system loses two further ranks if it does not exhibit any rotational motion and, at the same time, accelerates with $-\boldsymbol{g}$ in the world frame ($\hat{\boldsymbol{f}} = -2\boldsymbol{C}^T\boldsymbol{g}$). This represents a rather unrealistic situation our robot might find itself in. In the general case the system will rarely be perfectly at a singular point and thus the corresponding filter should be able to observe all state except for the globally unobservable position and yaw angle. Also, please remember that the above table describes the case where only a single foot is in contact with the ground and that the rank deficiency tends to be smaller if more contacts are available.

## V. RESULTS AND DISCUSSION

The presented filter was implemented and evaluated on our quadruped platform StarlETH [9]. For the experiments the output of the state estimation was used to stabilize and control the robot. We illustrate the filter performance at hand of an experiment where the robot trots over uneven and highly slippery terrain. Figure 1 shows a sequence of images depicting the trajectory of the robot. It covers a distance of approximately 3 m in roughly 15 s.

In Figure 2 a detailed sequence of snapshots shows a slip situation towards the end of the experiment (around the last image of Figure 1). For this sequence we plotted the results of the outlier detection algorithm of Section III-C in Figure 3. The three distinct blocks in the figure correspond to the stance phases of three subsequent steps of the left hind leg. While the light gray surface represents the contact detected by the contact sensor, the dark gray surfaces represent the detection of outliers. The first block corresponds to the slip situation of Figure 2, where the dark gray phase towards the beginning of the stance phase represents detected slip (the contact is also lost for a very short instant). There are a few unexpected outlier detections throughout the dataset. They often occur at the beginning or towards the end of stance phase where the foot is not well in contact with the ground and where oscillations can occur due to the compliance of the foot. In contrast to our previous work [2], where an estimate of the

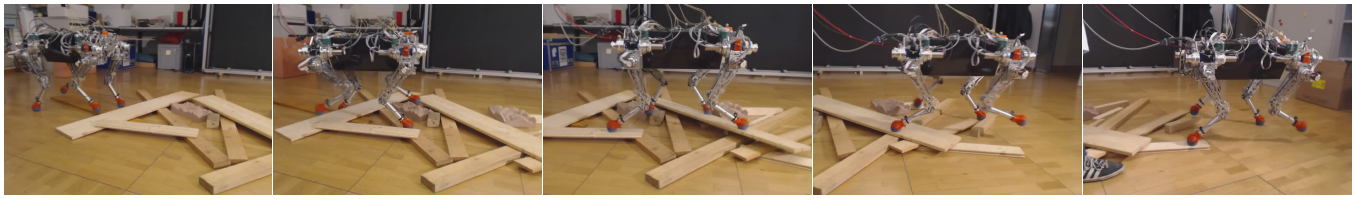

Fig. 1. Trotting sequence over uneven and slippery terrain. The robot requires about 15 s for traversing the 3 m long area covered with loose wooden planks.

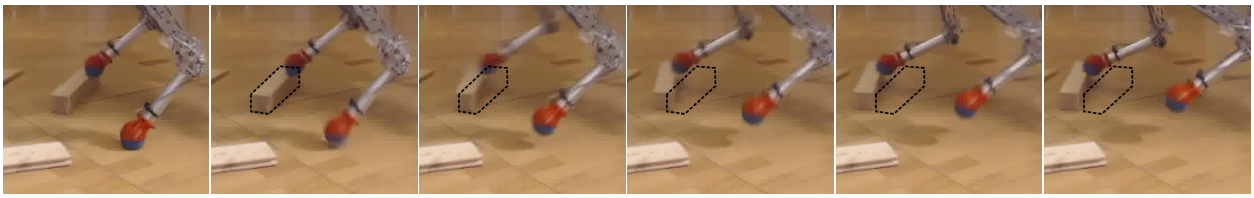

Fig. 2. A sequence of snapshots illustrating the substantial slip that is occurring during the experiment. If looking at the plank beneath the left foot one can observe that it is moved by approximately 10 cm. Time between snapshots: 32 ms.

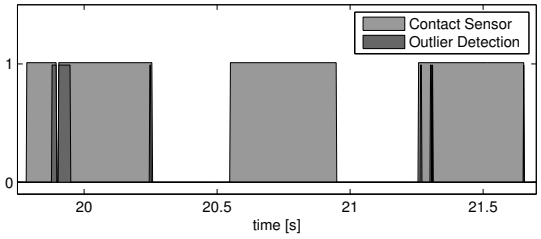


Fig. 3. Binary outputs from contact sensor and outlier detection of the left hind leg. Light gray: flag of contact sensor (1 = contact). Dark grey: outlier detection (only detect outliers if the contact sensor flag is true). Three stance phases are displayed. In the first stance phase slippage is detected which corresponding to the slip event illustrated in Figure 2.

foothold is initialized at each new step, the present filter is much less susceptible to fast switching foot contacts.

Figure 4 and Figure 5 show the resulting estimates for the attitude and the velocity of the robot main body. From the point of view of the local controller those quantities are of high importance in order to enable the stabilization of the main body. As pointed out in Section IV the angle around the gravity axis (yaw) is not observable and consequently the filter estimate will drift away. However, for the remaining two degrees of freedom (pitch and roll) very precise results are obtained with RMS values below 0.01 rad if compared to the motion capture data. The plotted $3\sigma$ covariance bounds of the attitude estimates roughly captures the uncertainty of the system and the motion capture attitude remains between the bounds for most of the time (there are some outliers in the motion capture data).

The velocity estimates are more difficult to evaluate due to noisy numerical differentials of the motion capture system. Still, one can observe a nice overlay between both trajectories. Here, all three quantities are observable and after a very quick initial convergence the covariance estimates remain more or less constant. The obtained RMS values are around 0.05 m/s, whereas a large amount is caused by the noisy motion capture estimates. If compared to the filter presented in [2] the RMS errors for the velocity estimates as well as for the roll and pitch angles are roughly halved for this experiment (for pitch there

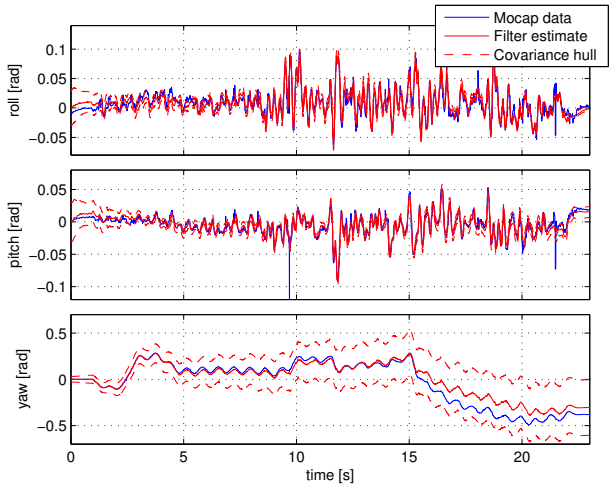


Fig. 4. Roll, pitch, and yaw angles of the main body for the sequence depicted in Figure 1. Red: estimated values. Red dashes: $3\sigma$ covariance bounds. Blue: motion capture data. The RMS values for the roll, pitch, and yaw estimates are: 0.0086 rad, 0.0056 rad, 0.0693 rad.

is even a factor 10). This comes at costs of accuracy on the position and yaw angle. However, as mentioned earlier those quantities are of secondary interest and their estimation could be improved by integrating more suitable sensor modalities like vision or LIDAR.

## VI. Conclusion and Future Work

In this paper we presented a novel state estimation approach for legged robots based on kinematic velocity measurements at the ground contacts. The obtained information is fused with measurements from an on-board IMU by means of an unscented Kalman filter. The provided nonlinear observability analysis shows that, for general robot motions, all states are observable except for the global position and the yaw angle. This results in a filter which accurately estimates the inclination angles (roll and pitch) as well as the velocities of the robot. It also avoids unnecessary assumptions on the shape of the floor or on the employed gait pattern and is robust to a certain amount of foot slippage. Implemented on our legged

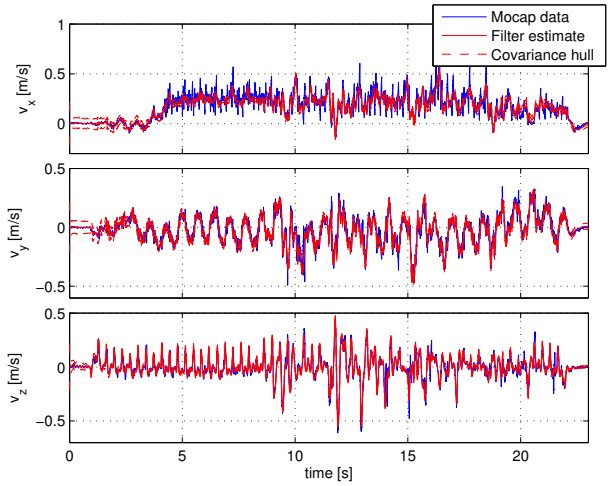


Fig. 5. Velocity estimates of the main body for the sequence depicted in Figure 1. Red: estimated velocity values. Red dashes: $3\sigma$ covariance bounds. Blue: motion capture data. The RMS values for the three velocity estimates are: 0.0546 m/s, 0.0406 m/s, 0.0348 m/s.

robot StarlETH, it enables dynamic locomotion over uneven and labile terrain.

While the position and the yaw angle of the robot are quantities which are less critical for a local stabilization of force controlled legged robots, they are important for global navigation. Future work will thus include evaluating different methods for integrating further sensor modalities which are more suited for navigation and terrain perception.

## REFERENCES


[1] G. Agamennoni, J. I. Nieto, and E. M. Nebot. An outlier-robust Kalman filter. In *IEEE Int. Conf. on Robotics and Automation*, May 2011.

[2] 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. In *Proc. of Robotics: Science and Systems*, July 2012.

[3] N. El-Sheimy, H. Hou, and X. Niu. Analysis and Modeling of Inertial Sensors Using Allan Variance. *IEEE Trans. on Instrumentation and Measurement*, 57(1):140–149, 2008.

[4] B. Gassmann, F. Zacharias, J. M. Zöllner, and R. Dillmann. Localization of Walking Robots. In *IEEE Int. Conf. on Robotics and Automation*, April 2005.

[5] M. Görner and A. Stelzer. A leg proprioception based 6 DOF odometry for statically stable walking robots. *Autonomous Robots*, pages 1–16, 2013.

[6] O. Gur and U. Saranli. Model-Based Proprioceptive State Estimation for Spring-Mass Running. In *Proc. of Robotics: Science and Systems*, June 2011.

[7] R. Hermann and A. Krener. Nonlinear controllability and observability. *IEEE Trans. on Automatic Control*, 22(5): 728–740, October 1977.

[8] C. Hertzberg, R. Wagner, U. Frese, and L. Schröder. Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds. *Information Fusion*, 14(1):57–77, 2011.

[9] M. Hutter, C. Gehring, M. Bloesch, M. A. Hoepflinger, C. D. Remy, and R. Siegwart. StarlETH: A compliant quadrupedal robot for fast, efficient, and versatile locomotion. In *Int. Conf. on Climbing and Walking Robots*, July 2012.

[10] S. J. Julier. The scaled unscented transformation. In *Proc. of the American Control Conference*, May 2002.

[11] K. Kaneko, F. Kanehiro, S. Kajita, M. Morisawa, K. Fujiwara, K. Harada, and H. Hirukawa. Slip observer for walking on a low friction floor. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, August 2005.

[12] V. Lebastard, Y. Aoustin, and F. Plestan. Estimation of Absolute Orientation for a Bipedal Robot: Experimental Results. *IEEE Trans. on Robotics*, 27(1):170–174, 2011.

[13] P. C. Lin, H. Komsuoglu, and D. E. Koditschek. Sensor data fusion for body state estimation in a hexapod robot with dynamical gaits. *IEEE Trans. on Robotics*, 22(5): 932–943, 2006.

[14] J. Ma, S. Susca, M. Bajracharya, L. Matthies, M. Malchano, and D. Wooden. Robust multi-sensor, day/night 6-DOF pose estimation for a dynamic legged vehicle in GPS-denied environments. In *IEEE Int. Conf. on Robotics and Automation*, May 2012.

[15] F. M. Mirzaei and S. I. Roumeliotis. A Kalman filter-based algorithm for IMU-camera calibration: Observability analysis and performance evaluation. *IEEE Trans. on Robotics*, 24(5):1143–1156, 2008.

[16] N. Okita and H. J. Sommer. A novel foot slip detection algorithm using unscented Kalman Filter innovation. In *Proc. of the American Control Conference*, June 2012.

[17] M. Reinstein and M. Hoffmann. Dead Reckoning in a Dynamic Quadruped Robot Based on Multimodal Proprioceptive Sensory Information. *IEEE Trans. on Robotics*, PP(99):1–8, 2012.

[18] G. P. Roston and E. P. Krotkov. Dead Reckoning Navigation For Walking Robots. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, July 1992.

[19] S. Singh, P. Csonka, and K. Waldron. Optical Flow Aided Motion Estimation for Legged Locomotion. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, October 2006.

[20] H. W. Sorenson and D. L. Alspach. Recursive bayesian estimation using gaussian sums. *Automatica*, 7(4):465–479, 1971.

[21] A. Stelzer, H. Hirschmüller, and M. Gorner. Stereo-vision-based navigation of a six-legged walking robot in unknown rough terrain. *The International Journal of Robotics Research*, 31(4):381–402, February 2012.

[22] J.-A. Ting, E. Theodorou, and S. Schaal. A Kalman filter for robust outlier detection. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, October 2007.

[23] M. West. Robust sequential approximate Bayesian estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, 43(2):157–166, 1981.