2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Tactile Odometry in Aerial Physical Interaction

Micha Schuster[†,1], Anton Bredenbeck[†,2], Michael Beitelschmidt[1], Salua Hamaza[2]

Abstract— **Aerial robots are well-established technologies in environments characterized by reliable GNSS signals and favorable conditions for navigation based on cameras or LiDARs. However, their robustness is significantly challenged whenever ambient lighting is insufficient, GNSS signals are blocked, and range measurements are corrupted, for example, in underground, dark, or foggy environments. There, conventional navigation methods solely based on computer vision are very limited. This work proposes a completely novel approach to *Aerial Tactile Odometry* for pose estimation of aerial robots exploiting contact to precisely determine the system's pose. By employing a compliant end-effector design with onboard tactile information by means of a trackball, we infer the complete UAV's pose with respect to the environment, and the path traveled during contact. Through a large set of experiments, the proposed method shows centimeter-level accuracy for various relative orientations between the environment and the robot as well as for different trajectories. Akin to conventional dead-reckoning odometry methods in wheeled robotics, this method provides a valuable additional source of pose estimation, increasing the robustness of aerial robots – especially aerial manipulators – in the real world.**

I. INTRODUCTION

Aerial robots, commonly known as drones, have become evermore present in many applications and sectors. Their ease of use, availability, and low cost have enabled them to find use cases in various domains, such as inspection, agriculture, environmental monitoring, and search & rescue.

So far, the modalities exploited by drones for autonomous navigation include Global Navigation Satellite System (GNSS), LiDAR, or vision to achieve safe and reliable navigation in unknown environments. All these methods present their own advantages but also shortcomings, e.g., GNSS-based navigation with real-time kinematics extension is extensively used on drones and can reach sub-centimeter accuracy in the presence of a stable satellite signal and reference stations [1]; however, a high density of obstacles, proximity to buildings, or indoor environments block GNSS signals, leading to loss of accuracy or complete failure. LiDAR systems provide high-resolution maps at the cost of high power consumption, weight, size, and computational load. Lastly, cameras provide a compact, lightweight and self-sustained solution to navigation with centimeter accuracy [2], [3]; however, they require an environment rich of visually trackable features and are susceptible to low lighting, high dynamic range, and motion blur [4]. As an alternative approach, this work aims at introducing and validating the use of tactile feedback for odometry estimation on drones as a lightweight and self-sustained solution, immune to lighting conditions.

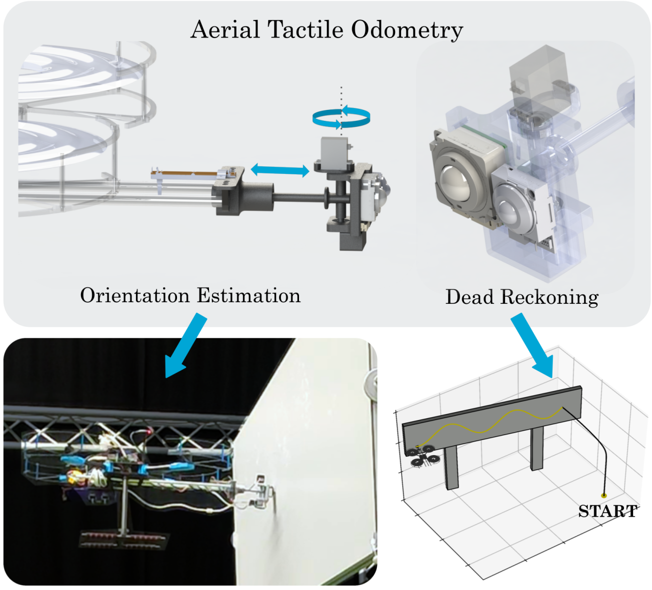

Fig. 1. Three key mechanisms enable the tactile sensing system on board a standard quadrotor: Linear compliance in the EE enables safe contact establishment and regulation. A passive rotational joint ensures the EE always aligns with the environment. A trackball informs the system about the distance traveled while in contact. These interactions yield an odometry estimate based on tactile information.

Drones capable of establishing contact with the environment to perform aerial physical interaction are known as Aerial Manipulators (AMs). Early works in this field focused on force exertion via *single-point-of-contact* for applications in Non-Descrutive Testing (NDT) tasks [5]–[8]. Later on, more dynamic force and/or position tracking on drones was accomplished via *moving-point* tracking on 2D and 3D surfaces [9]–[12]. Lastly, the latest works realize complex tasks with varying degree of freedom during operation like drilling or screwing [13], [14]. All the above works have progressed positioning accuracy of moving end-effectors on drones in the task space, serving the purpose of manipulation.

In the recent work [15], a drone with an anthropomorphic finger embodies mechanical compliance to infer information about the unknown environment in a *proprioceptive* way and uses it for tactile navigation over unknown surfaces, enabling wall following but not position estimation. As a follow-up to this concept, this work introduces tactile odometry during dynamic aerial contact tasks over flat surfaces, employing a trackball-based end-effector design. Notably,

---

[†] Equal contribution.
[1] Dynamics and Mechanism Design Group, TU Dresden.
[2] Biomorphic Intelligence Lab, Aerospace Engineering, TU Delft.
We abide by the FAIR-Principles of science. Project Page: `https://github.com/BioMorphic-Intelligence-Lab/tactile-odometry-drone`

979-8-3503-7769-9/24/$31.00 ©2024 IEEE

ground-wheeled robots are known for using wheel odometry for navigation purposes; e.g., the NASA Mars rover or the autonomous cleaning robots are prominent examples of classical dead-reckoning wheel odometry [16]. However, to the best of the authors' knowledge, navigation modalities using tactile odometry have not yet been investigated in aerial robotics. Hereby, we employ the end-effector as a tool for adding a tactile-based navigation modality, thereby using it for the drone's state estimation in Cartesian space.

Therefore, the main contributions of this work are:

- We propose a compliant mechanical design of an EE that enables alignment with and following of randomly oriented vertical surfaces, as illustrated in Fig. 1.
- We propose for the first time a tactile-based quantitative state estimation approach for drones based on the information extracted from contact: we estimate the position and yaw angle of an Unmanned Aerial Vehicle (UAV) w.r.t. the contact surface from IMU measurements, EE joint angles, and trackball-ticks only.
- We validate this approach in experiments on differently oriented vertical surfaces, where the system successfully reproduces the ground truth within centimeter accuracy.

II. TACTILE AERIAL ODOMETRY

We aim to introduce tactile aerial odometry as a novel concept in aerial robotics, building upon the idea of exploiting contact with the environment for improving navigation and autonomy of drones. In this preliminary study, we integrate a trackball as part of the EE's design to measure the traveled path over a flat vertical surface, inferring local position estimation. At contact, the UAV autonomously aligns itself to the wall, regulates its normal distance to the wall, and follows a predefined trajectory w.r.t. the wall surface. During contact, the odometry filter estimates the pose of the UAV body w.r.t. the wall frame. This output can then be used for closing the position control loop, as separately controlling the UAV's position on the wall surface and its normal distance to it, instead of its 3D position w.r.t. a world frame, increases the robustness of the control loop. Fig. 2C shows the schematic overview of the full system. Three key features comprise the overall system:

1) a mechanically compliant EE to ensure stable contact and to regulate the normal force magnitude,
2) the passive rotary joint in the EE ensures a proper alignment with the target surface,
3) the trackball provides measurement data for dead reckoning position estimation.

A. Mechanical Design

The tactile odometry setup is mounted on a conventional quadrotor (Holybro X500-V2). We propose a compliant EE that implements the previously mentioned key features, illustrated in Fig. 2A: the contact interface with trackballs and mounting plate, the revolute encoder and axle, and the linear axle with its potentiometer. The contact interface consists of one trackball as input for the odometry filter and a second trackball to enable two mechanically well-defined points of contact. This allows for self-alignment of the EE about the revolute axle and low friction while moving along the wall. The mounting plate features an Inertial Measurement Unit (IMU) for roll and pitch angle data to measure the current orientation of the trackballs, as using the onboard IMU of the flight controller would induce too much error due to the flexibility of the manipulator. The EE also has a stiff linear compliance, whose displacement is measured by a linear potentiometer, allowing estimation and regulation of the wall distance and contact force based on the spring's displacement. The angle ϑ and displacement d from the joints are combined in the vector $\boldsymbol{\xi} = \begin{pmatrix} \vartheta & d \end{pmatrix}$.

TABLE I

NOMENCLATURE: FRAMES AND TRANSFORMATIONS

Symbol	Description	Symbol	Description
$\mathcal{W}$	world frame	${}^{A}_{B}\mathbf{R}$	rotation matrix, changing coordinates of vector from frame $\mathcal{B}$ to $\mathcal{A}$
$\mathcal{S}$	wall-surface frame	$\mathbf{R}_{\square}$	elementary rotation matrix about axes $\square$
$\mathcal{B}$	UAV body frame	${}^{\mathcal{A}}\boldsymbol{p}_{\mathcal{BC}}$	vector from origin of $\mathcal{B}$ to origin of $\mathcal{C}$, denoted in $\mathcal{A}$
$\mathcal{E}$	end-effector pivot frame	$\square_{\text{true}}$	quantity from ground truth
$\mathcal{O}$	odometry frame (i.e. TCP)	$\square_{\text{des}}$	desired quantity

B. Reference Frames

To clearly describe the system, the used reference frames are defined in Table I and visualized in Fig. 2B. The wall-surface frame $\mathcal{S}$ is located on the wall at the point of first established contact between the Tool Center Point (TCP) and the wall. Its y-axis is the normal vector of the wall, and its z-axis points upwards. Its x-axis points to the left to obtain a right-handed reference frame. The body frame is located at the center of the UAV and is the reference frame for the pose control of the flight controller. Its axes follow the convention (x-y-z): left-back-up. The EE pivot frame $\mathcal{E}$ lies on top of the rotational axle of the manipulator and rotates about its z-axis. The odometry frame $\mathcal{O}$ is located in the TCP, which coincides with the tip of the trackball. If the rotary joint is in its neutral state, the orientation of the odometry frame and end-effector pivot frame are identical.

III. ODOMETRY FILTER

The odometry filter estimates the pose of the UAV w.r.t. the wall frame based on the trackball measurements $\boldsymbol{m}$, the angle and displacements $\boldsymbol{\xi}$ from the EE joints and the current heading (roll) of the trackball ϕ_{IMU} measured by the EE IMU. Its signal flow diagram is shown in Fig. 3. This work employs the mathematical nomenclature as defined in Table I.

A. Orientation Estimation

This paper aims to use only an IMU and the tactile odometry for pose estimation during contact. Based on IMU measurements, the roll and pitch angle of the UAV can be

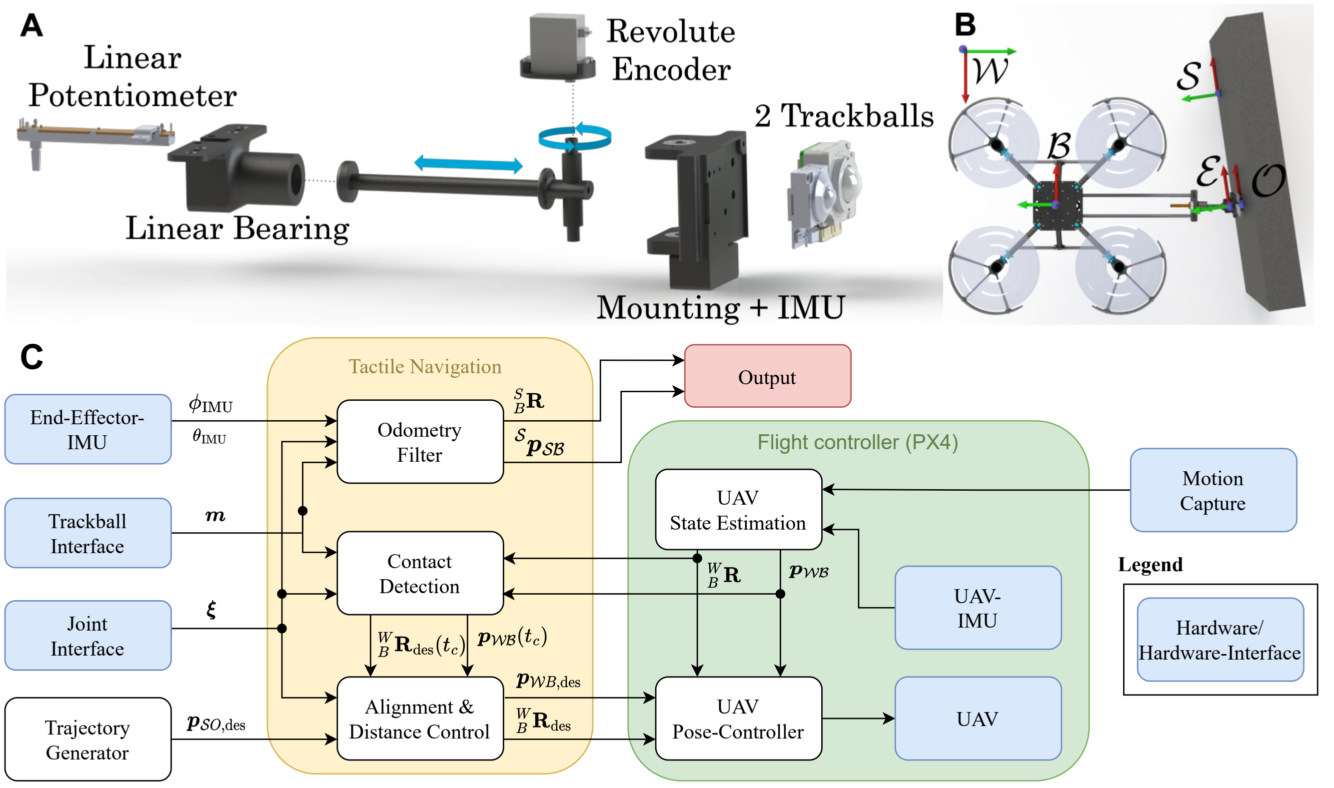

Fig. 2. A: Components of the mechanically compliant EE. It comprises two trackballs as the contact interface attached to a rotational and linearly compliant mounting plate. The displacement of the revolute and linear compliance displacement is measured using encoders to provide additional information about the odometry. B: Rendered CAD view of the UAV during contact with visualization of the used frames. C: Overview of the tactile navigation system.

estimated, whereas the IMU-based estimate of the yaw angle $\gamma_{\mathcal{SB}}$ has a high uncertainty. Thus, the estimation of the UAV yaw angle requires additional external information. For this purpose, the EE of the UAV can freely tilt about the UAV's z-axis to be able to align with any vertical wall in a way that the y-axis of the odometry frame is parallel to the normal vector of the wall (i.e., the y-axis of the wall frame). The orientation ${}^{S}_{B}\mathbf{R}$ of the UAV w.r.t. the wall frame is estimated based on the pitch θ_{IMU} and roll ϕ_{IMU} estimate from the IMU, and the yaw estimate based on the measured EE joint angle ϑ:

$$ {}^{S}_{B}\mathbf{R} = \mathbf{R}_z(-\vartheta) \cdot \mathbf{R}_y(\phi_{\text{IMU}}) \cdot \mathbf{R}_x(\theta_{\text{IMU}}). \qquad (1) $$

B. Position Estimation

1) Dead Reckoning from Trackball Rotation: The position increment of the local trackball motion ${}^{\mathcal{O}}\Delta\boldsymbol{p}_{\mathcal{SO}}$ is defined from the sensor ticks $m_{x,z}$ as

$$ {}^{\mathcal{O}}\Delta\boldsymbol{p}_{\mathcal{SO}} = k \cdot \begin{pmatrix} m_x & 0 & m_z \end{pmatrix}^{\top} = k \cdot \boldsymbol{m} \qquad (2) $$

with constant k, the conversion constant from ticks to meters. To obtain the position ${}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SO}}$ of the odometry frame $\mathcal{O}$ w.r.t. the wall frame $\mathcal{S}$, the position increment is summed up using the current heading of the odometry frame ${}^{S}_{O}\mathbf{R}$ on the wall

$$ {}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SO}}(t) = \sum_{t_i=t_c}^{t_i=t} {}^{S}_{O}\mathbf{R}(t_i) \cdot k \cdot \boldsymbol{m}(t_i). \qquad (3) $$

The heading of the odometry frame equals the roll angle ϕ_{IMU} of the EE-IMU, as the z-axes of the wall frame and world frame are parallel per definition, and the position estimate is only in the x-z-plane of the odometry frame, the relative orientation between the wall and the odometry frame can be simplified as

$$ {}^{S}_{O}\mathbf{R} = \mathbf{R}_y(\phi_{\text{IMU}}). \qquad (4) $$

2) Inverse Kinematics: The pose of body frame $\mathcal{B}$ relates to the estimated pose of the odometry frame $\mathcal{S}$ via:

$$ {}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SB}} = {}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SO}} - {}^{S}_{B}\mathbf{R} \cdot \left[{}^{\mathcal{B}}\boldsymbol{p}_{\mathcal{BE}}(d) + {}^{B}_{E}\mathbf{R}(\vartheta) \cdot {}^{\mathcal{E}}\boldsymbol{p}_{\mathcal{EO}} \right] \qquad (5) $$

with the EE's pose frame $\mathcal{E}$ w.r.t. the UAV body frame $\mathcal{B}$ as:

$$ \begin{aligned} {}^{B}_{E}\mathbf{R}(\vartheta) &= \mathbf{R}_z(\vartheta) && (6) \\ {}^{\mathcal{E}}\boldsymbol{p}_{\mathcal{BE}} &= \boldsymbol{p}_{\mathcal{BE},0} - \boldsymbol{e}_y \cdot d && (7) \end{aligned} $$

and the position vector ${}^{\mathcal{E}}\boldsymbol{p}_{\mathcal{EO}}$ from the EE pivot frame to the odometry frame.

IV. TACTILE WALL FOLLOWING

The odometry-based pose estimation works only while in contact; consequently, the UAV has to maintain contact between the wall and the TCP at all times. To establish contact with the wall, the desired target position in free flight is placed slightly "behind" the wall surface. After contact with the wall is detected by the UAV, it aligns itself

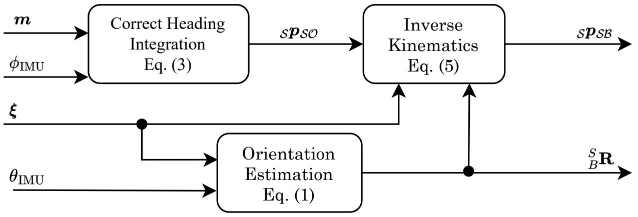

Fig. 3. Signal flow diagram of the odometry filter. Integrating the measured ticks of the trackball $\boldsymbol{m}$ and the EE heading ϕ_{IMU} yields its position on the surface of the environment ${}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SO}}$. In combination with the angle of the EE revolute joint ϑ this yields the observed UAV pose ${}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SB}}$.

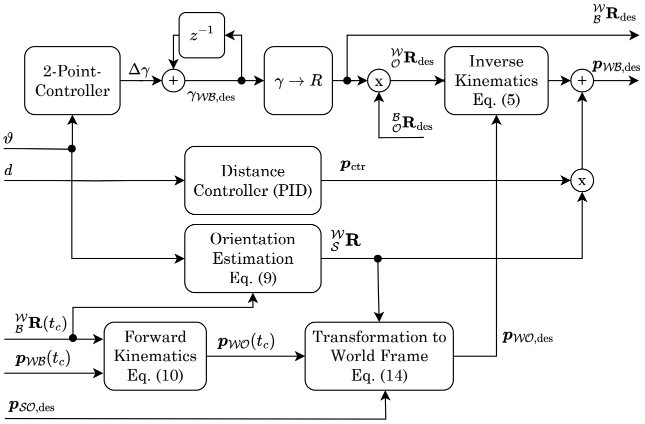

Fig. 4. Signal flow diagram for wall following. The angle relative to the environment is regulated using a 2-point controller that drives the angle of the passive revolute joint to a dead band close to zero. The distance to the environment is regulated via a PID controller that drives the linear compliance to a set point. Using the resulting aligned pose, we transform the nominal trajectory planned in the wall frame $\mathcal{S}$ in the world frame $\mathcal{W}$.

to the wall and ensures a constant distance, which enables following an arbitrarily oriented wall without losing contact. This allows it to receive a desired trajectory w.r.t. the wall frame and follow it based on its pose w.r.t. the world frame. Fig. 4 shows the overall signal flow diagram of the procedure.

A. Alignment

The alignment algorithm enables the UAV to align perpendicularly with a vertical surface of unknown orientation in the world frame $\mathcal{W}$, in conjunction with the underlying low-level flight controller. The orientation of the wall and the position of first contact must be determined at first in the world frame. For this we employ a Motion Capture System in free flight, until stable contact with the wall is established. We consider the established contact as *stable* if either the linear compliance exceeds a minimum displacement threshold $\epsilon_{\text{JS}} = 1\,\text{mm}$ or the trackballs have registered a minimum motion $\epsilon_{\text{TB}} = 3\,\text{cm}$ for a minimum amount of time $t_\epsilon = 3\,\text{s}$, with heuristically determined values. Subsequently, the orientation of the wall frame ${}^{W}_{S}\mathbf{R}$ w.r.t. the world frame is estimated based on the orientation of the body frame ${}^{W}_{B}\mathbf{R}_{\text{true}}(t_c)$ at first moment of stable contact and the evolution of the joint angle during contact, where t_c denotes the time of first stable contact. From this, the orientation of the odometry ${}^{W}_{O}\mathbf{R}$ can be derived as

$$ {}^{W}_{O}\mathbf{R} = {}^{W}_{B}\mathbf{R}_{\text{true}}(t_c) \cdot \mathbf{R}_z(\vartheta(t_c)). \tag{8} $$

Assuming that the odometry frame and the wall frame are aligned in contact regarding their y-axis (i.e., the normal axis of the wall) and the wall frame's z-axis pointing up, we can estimate the orientation of the wall frame ${}^{W}_{S}\mathbf{R}$ w.r.t. the world frame as

$$ {}^{W}_{S}\mathbf{R} = \mathbf{R}_z\left(\gamma_{\mathcal{WO}}\right). \tag{9} $$

Whereas the yaw angle $\gamma_{\mathcal{WO}}$ is defined as the angle between the y-axis of the odometry frame and the y-axis of the world frame. The absolute position of the odometry frame ${}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WO},\text{true}}(t_c)$ at first contact is determined using the forward kinematics as the inversion of Eq. (5)

$$ \begin{aligned} {}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WO},\text{true}}(t_c) = {}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WB},\text{true}}(t_c) + {}^{W}_{B}\mathbf{R}_{\text{true}}(t_c) \\ \left[{}^{\mathcal{B}}\boldsymbol{p}_{\mathcal{BE}}(d(t_c)) + {}^{B}_{E}\mathbf{R}(\vartheta(t_c))\, {}^{\mathcal{E}}\boldsymbol{p}_{\mathcal{EO}}\right]. \end{aligned} \tag{10} $$

For the alignment, the desired yaw angle for each time step is calculated based on the initial yaw angle $\gamma_{\mathcal{WB},\text{des}}(t_c) = \gamma_{\mathcal{WB}}(t_c)$ at first stable contact,

$$ \gamma_{\mathcal{WB},\text{des}}(t_i) = \gamma_{\mathcal{WB},\text{des}}(t_{i-1}) + \Delta\gamma \tag{11} $$

and is computed incrementally using a two-point-controller with the angle increment

$$ \Delta\gamma = \begin{cases} 0, & \text{abs}(\vartheta) < \vartheta_{\text{band}} \\ \dot{\gamma}_{\mathcal{WB},\text{des}} \cdot T_s \cdot \text{sign}(\vartheta), & \text{else}, \end{cases} \tag{12} $$

desired angular velocity $\dot{\gamma}_{\mathcal{WB},\text{des}}$, sampling time T_s and width of the hysteresis band ϑ_{band}. From this, the new desired orientation is defined as

$$ {}^{W}_{B}\mathbf{R}_{\text{des}} = \mathbf{R}_z(\gamma_{\mathcal{WB},\text{des}}(t_i)). \tag{13} $$

To obtain the desired position ${}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WB},\text{des}}$ of the UAV w.r.t. the world frame, the desired position of the odometry frame ${}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WO},\text{des}}$ is computed based on the desired trajectory ${}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SO},\text{des}}$ in the wall frame

$$ {}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WO},\text{des}} = {}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WO}}(t_c) + {}^{W}_{S}\mathbf{R} \cdot {}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{WO},\text{des}}. \tag{14} $$

Furthermore, from the desired position of the odometry frame ${}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WO},\text{des}}$ the desired position of the body frame ${}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WB},\text{des}}$ is obtained using the inverse kinematics defined in Eq. (5). This results in a position feasible of maintaining contact between the UAV and the wall during the whole alignment procedure and pre-planned trajectory. The alignment utilizes the absolute orientation of the wall ${}^{W}_{S}\mathbf{R}$ w.r.t. the world frame, which is estimated from the orientation of the UAV at first stable contact. However, a potentially low quality of this estimate does not affect the alignment and wall following performance, as the alignment algorithm functions entirely relative to the wall.

B. Distance Control

To compensate for errors in the alignment and to ensure a sufficient contact force to reduce slippage of the trackball, a distance controller is utilized to control the distance between the UAV and the wall using the measured linear displacement d of the manipulator. The distance control position offset ${}^{\mathcal{S}}\boldsymbol{p}_{ctrl}$ is computed w.r.t. $\mathcal{S}$-Frame by a standard PID-controller Subsequently, the position offset ${}^{\mathcal{S}}\boldsymbol{p}_{ctrl}$ is transformed to the world frame and added to the desired UAV position as shown in Fig. 4.

V. EXPERIMENTAL SETUP

This section introduces the experimental setup used to validate the approach. Each flight consisted of the UAV lifting off, approaching, and safely establishing contact with a flat wall, aligning automatically with the arbitrarily oriented wall, and executing a pre-computed trajectory. Overall, we validate the proposed methods over 32 flights with 26 straight-line trajectories flights and 6 sinusoidal altitude variation flights with various nominal angles to the wall as shown in Table II. Throughout the flights, the flight controller is fed UAV position measurements from a Motion-Capture system at $120\,\mathrm{Hz}$ for state estimation, which also provides the ground truth for evaluation. The trackball (ATEG X19 laser-optical) provides data with $20\,\mathrm{Hz}$ sampling rate and a resolution of 425 increments per revolution. The ball is made of epoxy resin with a diameter of $19\,\mathrm{mm}$. The wall is made of pressed wood with a smooth surface finish. This surface poses a challenge for the trackball and demonstrates the system's robustness, as the hard trackball, as well as the wood, have low friction coefficients, and therefore require sufficient normal force to avoid slipping. The EE joints are measured at $20\,\mathrm{Hz}$ by a teensy micro-controller with a resolution of $0.09°$ for the rotational encoder $0.06\,\mathrm{mm}$ for the linear potentiometer. The trajectories are pre-computed in the wall frame, consisting of a straight line along the surface of the wall at $0.1\,\mathrm{m\,s^{-1}}$, or a sinusoidal trajectory at the same lateral velocity with amplitude $a = 0.2\,\mathrm{m}$ and period $T = 1\,\mathrm{m^{-1}}$. Onboard the UAV, a Raspberry Pi4 computes the reference pose resulting from the introduced methodology and forwards it to a Pixhawk 6C running the PX4 flight controller [17]. For completeness, the fixed translations between the reference frames on the UAV, in cm, are:

$${}^{\mathcal{E}}\boldsymbol{p}_{\mathcal{EO}} = (\text{-2, -4.2, -1.7})^\top\,;\ {}^{\mathcal{B}}\boldsymbol{p}_{\mathcal{BE},0} = (0,\ \text{-43, -6.7})^\top \tag{15}$$

VI. RESULTS

A. Error-Metrics

To evaluate the accuracy of a position estimation algorithm, the position error is commonly expressed as a vector in Cartesian space with x-, y-, and z-components or as a scalar Euclidean norm. These two error metrics benefit from their simplicity and intuitive comprehensibility. However, they sometimes hide systematic effects that can be identified using other error metrics. Deriving the position error on the wall from Eq. (3) we get the error sources for one single position increment ${}^{\mathcal{S}}\Delta\boldsymbol{p}_{\mathcal{SO}}$ as

$$\begin{aligned}\Delta\,{}^{\mathcal{S}}\Delta\boldsymbol{p}_{\mathcal{SO}}(t_i) = {} & k\cdot\boldsymbol{m}\cdot\frac{\partial}{\partial\phi_{\text{IMU}}}\left[\mathbf{R}_y(\phi_{\text{IMU}})\right]\cdot\Delta\phi_{\text{IMU}} \\ & + \mathbf{R}_y(\phi_{\text{IMU}})\cdot\boldsymbol{m}\cdot\Delta k + \mathbf{R}_y(\phi_{\text{IMU}})\cdot k\cdot\Delta\boldsymbol{m}.\end{aligned} \tag{16}$$

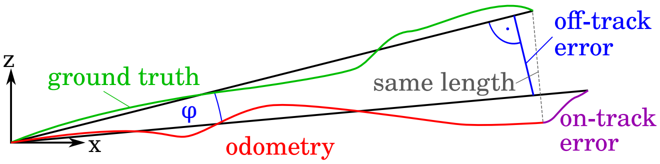

Fig. 5. Schematic definition of the on-track-error (purple) and off-track-error (blue) for a single position increment.

It can be seen clearly that besides the measurement error $\Delta\boldsymbol{m}$ the position error on the wall is directly dependent on the systematic errors from the trackball constant Δk and on the orientation of the IMU $\Delta\phi_{\text{IMU}}$. Thus, it is also beneficial to express the error in a way that enables the characterization of these error sources. To characterize the influence of the trackball constant k, the traveled distance L, or *on-track-error* is considered one error metric. As the motion capture system provides the ground truth w.r.t. the world frame, it has to be transformed to the wall frame before comparing it with the odometry position estimate

$${}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SO},\text{true}} = {}^{\mathcal{S}}_{\mathcal{W}}\mathbf{R}^\top_{\text{true}}\cdot\left({}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WO},\text{true}} - {}^{\mathcal{W}}\boldsymbol{p}_{\mathcal{WO},\text{true}}(t_c)\right). \tag{17}$$

With this, the length L for every increment ${}^{\mathcal{S}}\Delta\boldsymbol{p}_{\mathcal{SO}}$ is

$$L_{\text{true/odom}} = \sum_{t_i} ||\,{}^{\mathcal{S}}\Delta\boldsymbol{p}_{\mathcal{SO},\text{true/odom}}(t_i))||_2 \tag{18}$$

consequently, the *on-track-error* is defined as

$$\Delta p_{||} = L_{\text{true}} - L_{\text{odom}}. \tag{19}$$

To characterize the influence from an incorrect IMU heading ϕ_{IMU} that causes the position increments to be summed up in a wrong direction making the position estimation move orthogonally to the ground truth, we consider the *off-track-error* as

$$\Delta p_{\perp} = \tan(\varphi)\cdot||\,{}^{\mathcal{S}}\Delta\boldsymbol{p}_{\mathcal{SO},\text{true}}||_2 \tag{20}$$

with the angle φ between the simplified true and estimated paths. To allow for a more intuitive understanding, the off-track-error is defined as the distance error perpendicular to the direction of motion and not as an angular error. Besides a good possibility for optimizing the calibration parameters, the on- and off-track-error also realize a clear and practical relevant error metric to characterize the general position accuracy. Fig. 5 shows a schematic visualization of the definition of the on-track-error and off-track-error.

B. End-effector Position Estimation

Estimating the 2D position of the TCP on the wall is the foundation for estimating the 3D position of the UAV and thus is evaluated first. As the trackball odometry system is an incremental measurement system, the errors accumulate over time, as well as the total distance traveled. The error is evaluated over distinct path segments of varying lengths to visualize both the error of a single position increment and the resulting accumulated error. To evaluate the estimation quality statistically, the error metrics from Eq. (19) and

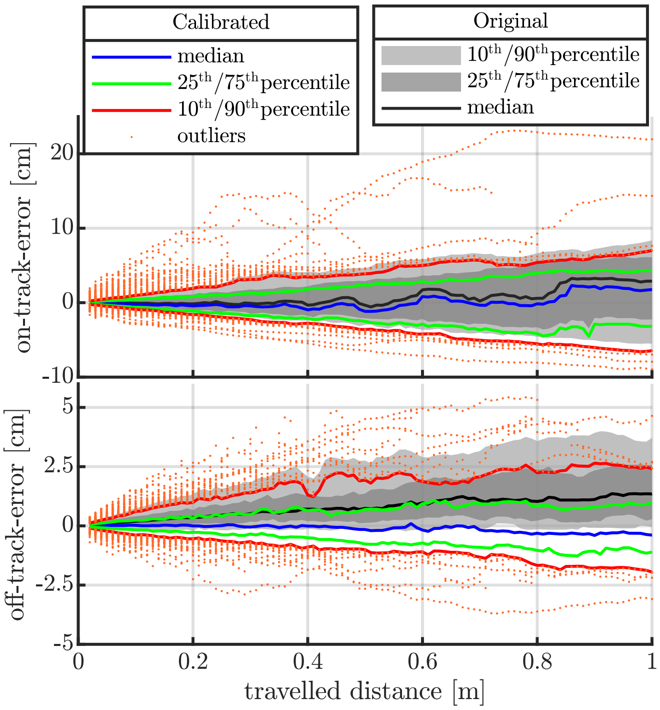

Fig. 6. Absolute on-track and off-track error for all experiments. Original data (grey) and calibrated data filtered with 3-sample-median-filter and calibration parameters corrected with $\Delta k = 1.04$, $\phi_{\text{IMU,offset}} = 1^\circ$.

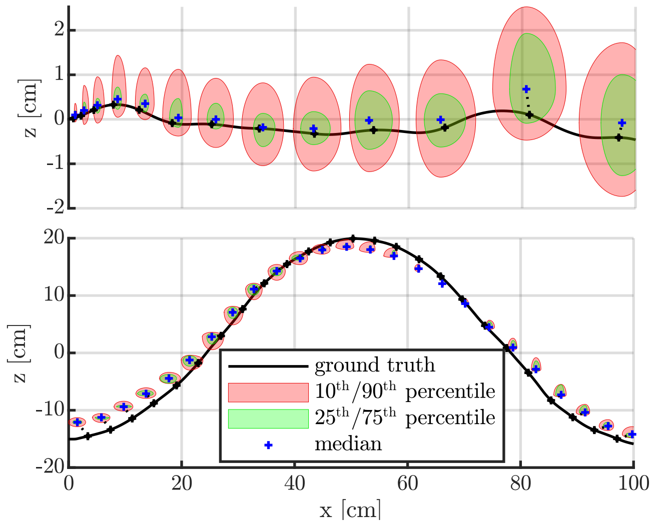

Fig. 7. Ground truth position in the wall frame, averaged over all line- and sinusoidal-flights respectively. The red and green ellipsoids indicate the region in which all odometry position estimates between the 25th and 75th resp. 10th and 90th percentile lie.

Eq. (20) are applied to segments of constant length of the ground truth measurement. Subsequently, the segment length is varied to show the error for different traveled distances.

One influence for a systematic error is the angle of the IMU. A small error in ϕ_{IMU} will cause the odometry estimation to drift in the wrong direction, causing an increasing off-track error. Likewise there can be a systematic error causing the trackball constant k to deviate from its nominal value. To compensate for the systematic errors, the calibration parameters k and $\phi_{\text{IMU,offset}}$ can be adapted. In particular, adjusting the trackball constant k, minimizes the median on-track-error and introducing a systematic offset of the IMU angle $\phi_{\text{IMU,offset}}$, minimizes the off-track-error. Fig. 6 shows original and reduced on- and off-track-errors for a calibration parameter set of $k = 1.04 \cdot k_0$ and $\phi_{\text{IMU,offset}} = 1^\circ$. Additionally, a median filter with a window length of three reduces sensor noise. While the on-track-error of the $10^{\text{th}}/90^{\text{th}}$ percentile is only a little above the 5% limit, some outliers to this range exist. The outliers with positive errors exceed those with negative errors by a multiple. From the definition of the error we can derive that a positive error results from the odometry length being smaller than the ground truth. The underlying cause is the trackball's slippage on the wall's surface, which is the most significant error source for the tactile areal odometry. As the off-track error is not sensitive to phenomena like slippage, a more symmetric shape of the outliers can be observed. The $10^{\text{th}}/90^{\text{th}}$ percentile stays below the 2.5% limit. However, there are also effects that influence the estimation performance. To give a qualitative impression of the estimation performance Fig. 7 shows the ground truth and odometry estimation in a 2D view on the wall for all 26 line flights from sets no. 1-4 and six sine flights from set 5. The averaged ground truth path is displayed in black together with the red and green ellipse-like areas that visualize the region that contains all odometry estimates from the 25th to the 75th and 10th to the 90th percentile respectively for distinct points on the path. Each ellipse-like area with each semi-major/-minor visualizes the respective percentile along the x- and z-axis. A blue cross indicates the median of the position estimation. The cartesian error has been considered for this 2D plot in the x-z-plane of the wall. For the line flights, it can be seen that the overall error stays within a small region around the ground truth and is dominated by the spread, while the median error stays small. Instead, the median error dominates the spread for the sine flights, indicating a systematic component in the error.

C. UAV Position Estimation

The position of the UAV can be obtained by the transformation of the position of the TCP on the wall by the inverse kinematics shown in Eq. (5). Thus the estimation error of the UAV comprises the estimation error of the TCP and the error propagation of the inverse kinematics. To analyze the error propagation in more detail, the error of the 3D position of the UAV is derived from Eq. (5) with the approximation ${}^{S}_{B}\mathbf{R} = \mathbf{R}_z(\vartheta)$ and ${}^{S}_{B}\mathbf{R} \cdot {}^{B}_{E}\mathbf{R} = \mathbf{I}$ as

$$\begin{aligned} \Delta\, {}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SB}} = \Delta\, {}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SO}} + \frac{\partial}{\partial\vartheta}\left[\mathbf{R}_z(-\vartheta)\right] \cdot {}^{\mathcal{B}}\boldsymbol{p}_{\mathcal{BE}}(d) \\ + \mathbf{R}_z(-\vartheta) \cdot \Delta\, {}^{\mathcal{B}}\boldsymbol{p}_{\mathcal{BE}}(d) + \Delta\, {}^{\mathcal{E}}\boldsymbol{p}_{\mathcal{EO}} \end{aligned} \tag{21}$$

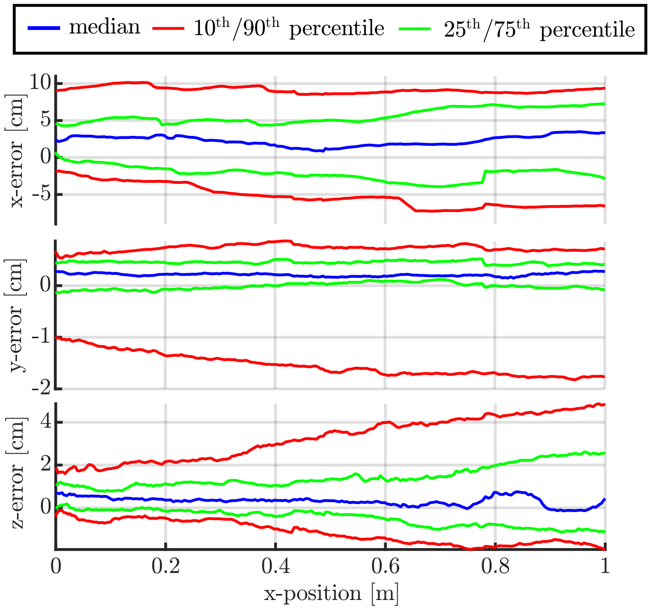

Fig. 8. Cartesian error of UAV-Body position w.r.t. wall for all line flights. Statistical evaluation over all 32 flights.

for small ϑ and known UAV geometry, $\Delta {}^{\mathcal{E}}\boldsymbol{p}_{\mathcal{EO}}$ and $\Delta {}^{\mathcal{B}}\boldsymbol{p}_{\mathcal{BE}}$ become neglectable, yielding to:

$$\Delta {}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SB}} \approx \Delta {}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SO}} + \begin{bmatrix} 0 & -\cos(\vartheta) & 0 \\ \cos(\vartheta) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot {}^{\mathcal{B}}\boldsymbol{p}_{\mathcal{BE}}(d) \cdot \Delta\vartheta \tag{22}$$

This shows that the error mostly consists of the 2D position estimation error $\Delta {}^{\mathcal{S}}\boldsymbol{p}_{\mathcal{SO}}$ and an orientation estimation error from $\Delta\vartheta$. As the orientation estimation for ${}^{S}_{B}\mathbf{R}$ with the error $\Delta\vartheta$ is computed only once at the instant of contact, the resulting error offset is present during the whole trajectory. The resulting 3D cartesian error of the UAV position is shown in Fig. 8. The orientation error $\Delta\vartheta$ is in the range of several degrees, see Table II, especially the x-error shows a significant constant offset and negligible growth. As the x-component of ${}^{\mathcal{B}}\boldsymbol{p}_{\mathcal{BE}}$ is zero according to Eq. (15), there is almost no error offset in the y-direction. The error in z-direction reflects the TCP position error in z.

D. Orientation Estimation

The estimation of the orientation between the wall and the UAV as described in Eq. (1) relies on the joint encoder of the EE and the roll and pitch angle measurements of the IMU. As the latter ones are considered to be state of the art, the error of the yaw angle $\gamma_{\mathcal{SB}}$ derived from the joint encoder is evaluated in Table II. In total, the mean yaw angle error $\Delta\gamma_{\mathcal{SB}} \approx 3°$ with a standard deviation of approx. $4.5°$. For further evaluation of the yaw angle error see Section VI-F.

E. Tactile Wall Following

To validate the wall following capabilities of the system, Figs. 9 and 10 show the alignment and wall following during contact. It can be seen clearly that the angle of the revolute axle ϑ is driven to the tolerance band by the 2-point-controller and the yaw angle $\gamma_{\mathcal{WB}}$ of the UAV shows a good alignment with the wall angle $\gamma_{\mathcal{WS}}$. In Fig. 10, the UAV following the wall is displayed as a 2D view of the x-y-plane of the world frame. The UAV starts on the left by initially aligning itself to the wall and regulating the distance and contact force magnitude. Subsequently, it follows the wall well from left to right, with the desired path being almost parallel to the true wall. The gray dotted lines represent points of equal time with $\Delta T \approx 3\,\mathrm{s}$.

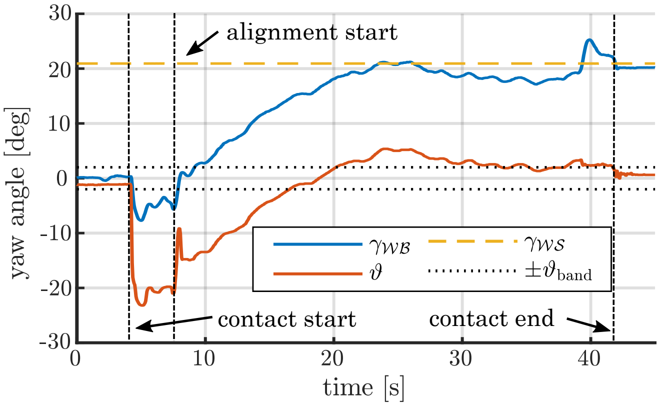

Fig. 9. Yaw angle $\gamma_{\mathcal{WB}}$ of the UAV, the wall $\gamma_{\mathcal{WS}}$, and the revolute EE joint ϑ during the alignment procedure.

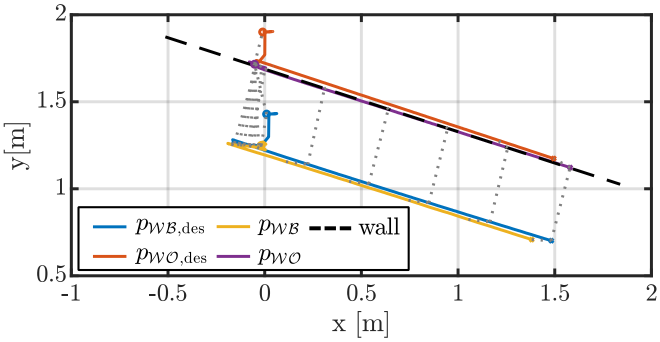

Fig. 10. 2D top-view of the estimated and true position of the UAV during wall following in the world frame. The dotted gray lines indicate the points of the same time. The circles indicate the starting points, and the crosses are the endpoints.

F. Flight Performance

The proposed approach was successfully validated in 32 flight experiments. Minor shortcomings were found in the trackball slippage which reduced the position-tracking capability in some cases. This phenomenon mostly occurs at the beginning or end of the flight, as shown in Table II. For larger initial angles between the wall and the UAV (set 3), many phases of weak contact resulted in slippage of the trackball. Additionally, the yaw angle estimation suffers higher errors (Table II) appointed to the fast rotation of the revolute axle upon establishing contact, causing the signal processing unit to miss several ticks.

TABLE II

YAW ANGLE ERROR $\Delta\gamma_{\mathcal{SB}}$ FOR ALL FLIGHT SETS (N TRIALS) AND PERCENTAGE OF TRAVELED DISTANCE WITH FUNCTIONAL ODOMETRY.

Set No.	Path shape (nominal wall angle)	N	$\Delta\gamma_{\mathcal{SB}}$ mean	$\Delta\gamma_{\mathcal{SB}}$ min	$\Delta\gamma_{\mathcal{SB}}$ max	$\Delta\gamma_{\mathcal{SB}}$ std	Mean Coverage [%]
1[1]	Line (0°)	7	5.5°	1.8°	11.9°	4.3°	99.9
2	Line (0°)	8	2.8°	0.9°	5.2°	1.7°	100
3	Line (15°)	4	−3.3°	−3.7°	−3.0°	0.3°	60.1
4	Line (−10°)	7	2.0°	−3.1°	4.3°	2.4°	83.1
5[1]	Sine (0°)	6	5.3°	−1.2°	14.5°	6.9°	93.3
total		32	2.9°	−3.7°	14.8°	4.6°	90.0

[1] alignment and distance control not enabled.

VII. CONCLUSION

This paper presents a novel methodology for estimating a UAV's pose by exploiting information acquired during continuous contact. Akin to traditional wheeled robots that use encoders to estimate their position, we propose to use a trackball embedded in a compliant EE to estimate the aerial robot's position w.r.t. a static environment. By measuring the displacement of the linear and rotational compliance, as well as the distance traveled on the environment's surface while in contact, the proposed methodology can regulate and ensure sufficient physical contact while estimating the UAV's position and yaw. In a total of 32 flight experiments, covering various relative environment orientations and trajectory profiles, the proposed methodology showcases centimeter accuracy with a relative error below 5% of the traveled distance, on average. This shows to be comparable to the accuracy of vision-based methods [2], [3], but without the need for sufficient lighting or otherwise favorable conditions for remote-sensing-based estimation methods. Furthermore, the system is capable of following unknown flat surfaces, allowing suitable contact for the position estimation in 90% of the tracked distance. The proposed approach is particularly suited to aerial interaction tasks where contact is desired, as such contact-based odometry provides feedback directly in the task space, enabling more precise EE control.

Overall, this work provides a valuable alternative solution to the odometry problem for aerial robots where conventional state estimation approaches, e.g., vision-based or GNSS-based, cannot be exploited. Such an approach paves the way to robust and lightweight aerial tactile navigation in challenging environments for emerging applications and robots-led underground surveys [18].

Future work in this direction will look into increasing the nominal accuracy by reducing the friction and the backlash in the mechanical system. Moreover, we will look into closing the loop: from contact-based odometry estimation to contact-based flight control, thereby providing a dual feedback response.

ACKNOWLEDGMENT

The work of Salua Hamaza was supported by project *"Aerial Robots in a Tangible World: Drones with the Sense of Touch Act upon Their Surroundings"* funded by the Dutch Research Council (NWO), grant number NWO-VENI-20308.

REFERENCES

[1] P. Henkel and A. Sperl, "Real-time kinematic positioning for unmanned air vehicles," in *2016 IEEE Aerospace Conf.*.

[2] G. Cioffi, L. Bauersfeld, E. Kaufmann, and D. Scaramuzza, "Learned inertial odometry for autonomous drone racing," *IEEE Robotics and Automation Letters*, 2023.

[3] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, "Iterated extended kalman filter based visual-inertial odometry using direct photometric feedback," *International Journal of Robotics Research*,

[4] T. H. Chan, H. Hesse, and S. G. Ho, "Lidar-based 3d slam for indoor mapping," in *2021 7th International Conference on Control, Automation and Robotics (ICCAR)*, 2021.

[5] H. W. Wopereis, J. J. Hoekstra, T. H. Post, G. A. Folkertsma, S. Stramigioli, and M. Fumagalli, "Application of substantial and sustained force to vertical surfaces using a quadrotor," in *2017 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2017.

[6] T. Ikeda, S. Yasui, M. Fujihara, *et al.*, "Wall contact by octo-rotor uav with one dof manipulator for bridge inspection," in *2017 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*.

[7] K. Bodie, M. Brunner, M. Pantic, *et al.*, "An omnidirectional aerial manipulation platform for contact-based inspection," in *Proceedings of Robotics: Science and Systems*, 2019.

[8] S. Hamaza, I. Georgilas, and T. Richardson, "An adaptive-compliance manipulator for contact-based aerial applications," in *2018 IEEE/ASME AIM*, 2018.

[9] M. Á. Trujillo, J. R. Martínez-de Dios, C. Martín, A. Viguria, and A. Ollero, "Novel aerial manipulator for accurate and robust industrial ndt contact inspection: A new tool for the oil and gas inspection industry," *Sensors*, 2019.

[10] S. Hamaza, I. Georgilas, and T. Richardson, "2d contour following with an unmanned aerial manipulator: Towards tactile-based aerial navigation," in *2019 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2019.

[11] M. Tognon, H. A. T. Chávez, E. Gasparin, *et al.*, "A truly-redundant aerial manipulator system with application to push-and-slide inspection in industrial plants," *IEEE Robotics and Automation Letters*, 2019.

[12] G. Malczyk, M. Brunner, E. Cuniato, M. Tognon, and R. Siegwart, "Multi-directional interaction force control with an aerial manipulator under external disturbances," *Autonomous Robots*, 2023.

[13] M. Schuster, D. Bernstein, P. Reck, S. Hamaza, and M. Beitelschmidt, "Automated aerial screwing with a fully actuated aerial manipulator," in *2022 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2022.

[14] C. Ding, L. Lu, C. Wang, and C. Ding, "Design, sensing, and control of a novel uav platform for aerial drilling and screwing," *IEEE Robotics and Automation Letters*, 2021.

[15] A. Bredenbeck, C. Della Santina, and S. Hamaza, "An aerial manipulator with a compliant finger embodies touch for navigation," *IEEE Transactions on Robotics*, 2024.

[16] K. S. Ali, C. A. Vanelli, J. J. Biesiadecki, *et al.*, "Attitude and position estimation on the mars exploration rovers," in *2005 IEEE Int. Conf. on Systems, Man and Cybernetics*.

[17] L. Meier, D. Honegger, and M. Pollefeys, "Px4: A node-based multithreaded open source robotics framework for deeply embedded platforms," *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[18] M. Tranzatto, T. Miki, M. Dharmadhikari, *et al.*, "Cerberus in the darpa subterranean challenge," *Science Robotics*, 2022.