IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

HR^2-KILO: A High-Rate, Robust, Kinematic-Inertial-LiDAR Odometry for Humanoid Robots

Jixin Gao, Fusheng Zha, Lianzhao Zhang, Wei Guo, Pengfei Wang, Lining Sun, Mantian Li

***Abstract*—In this letter, we present a high-rate and robust multi-sensor fusion framework for state estimation of humanoid robots, named HR^2-KILO. To handle the inherently high-dynamic characteristics of humanoid robots, the proposed framework tightly couples the measurements from the joint encoder, inertial sensor, and LiDAR. We estimate states within the error-state Kalman filter, incorporating the pointwise update strategy, IMU measurement model, and multiple leg kinematic information. Moreover, acceleration fluctuations, foot positions, and the history map are utilized for online contact detection without any contact sensors. The overall system fully utilizes the available multi-source information, making it compact and easy to deploy. Extensive experiments are conducted both on the public dataset and in the real world, including different humanoid robots and diverse scenarios. The results demonstrate that HR^2-KILO achieves extremely high rate output and lower drift compared to state-of-the-art LiDAR-inertial(-kinematic) methods. To contribute to the community, the source code and the multi-sensor humanoid dataset are released.**

***Index Terms*—SLAM, Sensor Fusion, Humanoid and Bipedal Locomotion.**

I. Introduction

HUMANOID robots are capable of seamlessly integrating into human-designed environments to perform diverse tasks [1], demonstrating high versatility and adaptability. Towards the ubiquitous deployment and application of humanoid robots, a critical challenge is the need for high-rate and robust state estimation. Specifically, humanoid robot locomotion is inherently unstable and non-smooth, particularly during high-dynamic activities. The instability necessitates high-frequency controllers for dynamic balance, requiring state estimation with matching real-time performance, which typically exceeds the frame rate of the sensors. Meanwhile, non-smooth motion, resulting from constant ground impacts, introduces additional sensor noise and periodic body oscillations, also posing a significant challenge for robust localization.

Manuscript received July 15, 2025; accepted October 6, 2025.

This paper was recommended for publication by Abderrahmane Kheddar upon evaluation of the Associate Editor and Reviewers' comments. This work was supported in part by the National Natural Science Foundation of China under Grant U2013602, in part by the Self-Planned Task of State Key Laboratory of Robotics and System (HIT) under Grant 2023FRFK01001, and in part by the Key Technology Research and Development of Humanoid Robots under Grant SYG2024142. *(Corresponding author: Fusheng Zha.)*

Jixin Gao, Fusheng Zha, Lianzhao Zhang, Wei Guo, Pengfei Wang and Lining Sun are with the State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150006, China (e-mail: 24b936018@stu.hit.edu.cn; zhafusheng@hit.edu.cn; lianzhao@stu.hit.edu.cn; wguo01@hit.edu.cn; wangpengfei@hit.edu.cn; lnsun@hit.edu.cn).

Mantian Li is with the Institute of Intelligent Manufacturing Technology, Shenzhen Polytechnic University, Shenzhen 518055, China (e-mail: limtsz@szpu.edu.cn).

Digital Object Identifier (DOI): see top of this page.

Responding to these challenges, some state estimation methods for legged robots are mainly based on proprioceptive sensors, including Inertial Measurement Units (IMU), joint encoders, and contact sensors. The methods incorporating this set of sensors are known as proprioceptive odometry, typically utilizing the Extended Kalman Filter (EKF) and its variants [2]–[4]. They are easily deployable and meet high-rate demands. However, the proprioceptive odometry struggles to maintain low drift during long-term operation [5], [6].

To overcome these limitations, a prominent solution is the fusion of exteroceptive sensors. The fusion methods based on optimizers are computationally intensive, making it difficult to achieve high-rate output. VILENS [7] is a fusion odometry based on factor graph optimization (FGO) for quadruped robots that effectively handles sensor degeneration due to non-ideal contacts. Leg-KILO [5] is also FGO-based, employing ikd-tree and keyframe selection to reduce computation, but the output frequency still depends on the LiDAR frame rate. In contrast, filter-based solutions naturally offer advantages in efficiency. LIKO [8] is proposed for bipedal robots, which incorporates leg kinematics with LiDAR-inertial odometry (LIO) based on iterated error-state Kalman filter (ESKF), improving the output rate to 1 kHz while neglecting LiDAR degradation. Additionally, for contact detection, the above methods either use contact sensors or compute the ground reaction forces (GRF) through dynamic modeling. The former introduces system redundancy, as contact sensors are non-essential for humanoid robots, while the latter requires more detailed model parameters, increasing the deployment complexity.

Therefore, this letter aims to: a) improve the odometry output rate to cover controllers' or planners' demands of humanoid robots, b) fully utilize the multi-source data to achieve long-term and robust localization under the non-smooth locomotion, c) achieve contact detection without contact sensors or dynamic model, making the system easy to deploy. To deliver on these promises, HR^2-KILO is proposed with the following contributions:

- A high-rate and robust kinematic-inertial-LiDAR odometry is presented for humanoid robots. The framework is based on the manifold ESKF, which fully utilizes multi-source data with the pointwise strategy, using IMU as a model measurement and tightly coupling the leg kinematic information.
- Online contact detection and foot position estimation are involved in our system, effectively integrating the foot position, LiDAR points and IMU fluctuations without relying on any contact sensors.
- Experiment results indicate that the proposed system can be deployed on different humanoid robots and achieves

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

high-rate outputs (6-15 kHz) with improved accuracy and robustness compared to the state-of-the-art methods.
- To benefit the community, we release our work and the dataset on Github[1].

II. Related Works

Some earlier works estimate the state for legged robots based on unscented KF or ESKF by proprioceptive sensors [2], [3]. For improved convergence properties and consistency, Hartley *et al.* [4] introduce invariant EKF to estimate the bipedal robot pose, velocity, and contact point, facilitating subsequent proprioceptive state estimators. Multiple IMUs are considered in the ESKF-based framework [9] to estimate both the robot state and structural deformation. The above methods provide a low-cost and lightweight solution for state estimation. Nevertheless, as described in [5], [6], the proprioceptive-only systems often suffer from the inevitable yaw drift, leading to poor long-term performance.

With the advancements in SLAM, many methods integrate proprioceptive sensors with camera or LiDAR. The reliability of the invulnerability to lighting conditions makes LiDAR prevalent in state estimation. LIO-SAM [10] is a representative LiDAR-inertial SLAM based on FGO, with the factors of IMU, LiDAR, and GPS. FAST-LIO2 [11] is based on ESKF and registers points directly in the map organized by ikd-tree. Faster-LIO [12] further introduces incremental voxels as the map structure to achieve higher efficiency. Since LiDAR is frame-based, motion compensation is required for point de-distortion. This process relies on the constant-velocity assumption, often violated in aggressive motions, causing LiDAR degradation. However, humanoid robots frequently experience such motions due to the ground impacts. Point-LIO [13] radically eliminates motion distortion by the pointwise state update, enabling a high-rate output (4-8 kHz). Nevertheless, its design does not target legged robots.

Fallon *et al.* [14] probabilistically fuse inertial, kinematic, and LiDAR data for humanoid robot localization, but priori points need to be collected for 10-30 s before operation. Recently, Legged-KILO [5], to cope with the dynamic motion of quadrupedal robots, combines leg odometry with LIO based on FGO, additionally including the contact height constraint. Based on FAST-LIO2, LIKO [8] introduces kinematic measurements to enable output rate up to 1 kHz and improve the estimation accuracy for humanoid robots. The common problems with the above methods are that the frame-based LiDAR update still affects the state estimation accuracy of humanoid robots, and contact sensors are all considered as the basis to leverage kinematic measurements.

For contact detection without contact sensors, model-based methods estimate GRF and get the contact state by thresholding [7], [15], [16]. Camurri *et al.* [17] employ logistic regression to learn the dynamic thresholds. In contrast, Lin *et al.* [18] introduce a learning-based contact estimator to classify contacts without wrench/force sensors. However, ground truth labels of contact states are often difficult to obtain, limiting the method generalization. The works in [19], [20] implement generalized contact estimation methods based on deep learning or the probabilistic model, but require at least an IMU on each leg. Different from the existing approaches, we reuse map points, inertial and kinematic information to detect contact events, making the system tightly coupled and easy to deploy.

[1]https://github.com/JixinGao/HR2-KILO

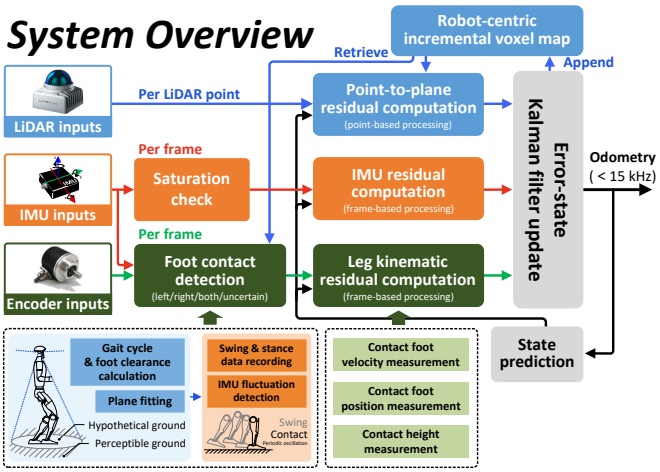

Fig. 1. The overview of the proposed system.

III. The Overview of Our System

In our system, LiDAR degeneration during humanoid robot locomotion is handled by pointwise state update, whereby high rate output is naturally achieved. IMU is used as a measurement rather than the model input. In addition, leg kinematic measurements without contact sensors are tightly coupled to enhance system robustness and accuracy.

The system overview is illustrated in Fig. 1. The data from LiDAR, IMU, and joint encoders are treated as measurements, and fed into the system at respective frequencies. When all the sensor streams are valid, the latter two are cached until the LiDAR data arrives for asynchronous fusion without frame alignment. Each LiDAR point in a frame is sequentially searched by the incremental voxel map for its local patch, then computing the point-to-plane residual. IMU residual is computed based on acceleration and angular velocity after saturation check. For leg kinematics, we first perform foot contact detection by integrating internal and external sensing information. The leg kinematic residuals are then constructed, considering foot position, velocity, and height constraints. The state update is performed at point and frame combination level frequency by ESKF, allowing the high-rate odometry output.

IV. State Estimation

A. Definitions

Some notations involved in this letter are shown in Table I. The system state $\mathbf{x}$ on the manifold $\mathcal{M}$ and process noise $\mathbf{w}$ are defined as:

$$\begin{aligned}
&\mathcal{M} \triangleq \mathrm{SO}(3) \times \mathbb{R}^{24}, dim(\mathcal{M}) = 27 \\
&\mathbf{x} \triangleq \left[{}^{W}\mathbf{R}_I^T \ {}^{W}\mathbf{p}_I^T \ {}^{W}\mathbf{v}_I^T \ \mathbf{b}_{\boldsymbol{\omega}}^T \ \mathbf{b}_{\mathbf{a}}^T \ {}^{I}\mathbf{a}^T \ {}^{I}\boldsymbol{\omega}^T \ {}^{W}\mathbf{p}_c^T \ {}^{W}\mathbf{g}^T\right]^T \\
&\mathbf{w} \triangleq \left[\mathbf{n}_{\mathbf{b}\boldsymbol{\omega}}{}^T \ \mathbf{n}_{\mathbf{b}\mathbf{a}}{}^T \ \mathbf{w}_{\boldsymbol{\omega}}{}^T \ \mathbf{w}_{\mathbf{a}}{}^T \ \mathbf{w}_{\mathbf{p}_c}^T\right]^T \approx \mathcal{N}(0, \mathcal{Q})
\end{aligned} \tag{1}$$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE I
SOME NOTATIONS

Notations	Explanations
${}^{I}(\cdot)$, ${}^{L}(\cdot)$	The IMU frame and the LiDAR frame.
${}^{B}(\cdot)$, ${}^{W}(\cdot)$	The body frame and the world frame.
$\boxplus/\boxminus$	The encapsulation operators on manifold.
$\mathrm{Exp}(\cdot)/\mathrm{Log}(\cdot)$	The exponential map and its inverse map between manifold and tangent vector.
$\lfloor\cdot\rfloor_\wedge$	The skew-symmetric cross product matrix.
$\mathbf{x}$	The ground-true state.
$\mathbf{x}_k$	The state at the k-th measurement sampling time.
$\hat{\mathbf{x}}$, $\bar{\mathbf{x}}$	The predicted and updated estimation of state.
$\tilde{\mathbf{x}}$	The error state between $\mathbf{x}$ and its estimation.

where ${}^{W}\mathbf{R}_I$, ${}^{W}\mathbf{p}_I$, and ${}^{W}\mathbf{v}_I$ represent the body rotation, position and velocity, $\mathbf{b}_{\boldsymbol{\omega}}$ and $\mathbf{b}_{\mathbf{a}}$ are random-walk biases of gyroscope and accelerometer, ${}^{I}\boldsymbol{\omega}$ and ${}^{I}\mathbf{a}$ are the acceleration and angular velocity measurements, ${}^{W}\mathbf{p}_c$ is the contact foot position, which is also involved in the system state, ${}^{W}\mathbf{g}$ is the gravity vector, $\mathbf{w}$ is the process noise with additional contact noise $\mathbf{w}_{\mathbf{p}_c}$ to improve robustness against slippages, and $\mathcal{Q} = diag(\mathcal{Q}_{b_\omega}, \mathcal{Q}_{b_a}, \mathcal{Q}_\omega, \mathcal{Q}_a, \mathcal{Q}_{p_c})$.

In addition, $\boxplus$ and $\boxminus$ in [21] are introduced to compactly describe the system on the manifold and parameterize the state error. For a compound manifold $\mathcal{M} = SO(3) \times \mathbb{R}^n$, it could be with the following expression:

$$\begin{bmatrix}\mathbf{R}\\ \mathbf{a}\end{bmatrix} \boxplus \begin{bmatrix}\mathbf{r}\\ \mathbf{b}\end{bmatrix} = \begin{bmatrix}\mathbf{R}\,\mathrm{Exp}(\mathbf{r})\\ \mathbf{a}+\mathbf{b}\end{bmatrix}, \begin{bmatrix}\mathbf{R}_1\\ \mathbf{a}\end{bmatrix} \boxminus \begin{bmatrix}\mathbf{R}_2\\ \mathbf{b}\end{bmatrix} = \begin{bmatrix}\mathrm{Log}(\mathbf{R}_2^T\mathbf{R}_1)\\ \mathbf{a}-\mathbf{b}\end{bmatrix} \tag{2}$$

B. State Prediction

The IMU frame is considered to coincide with the robot body, its first frame is taken as the world frame, and the extrinsic between the sensors is precalibrated. Assuming the input holds constant for period Δt_k, the system discrete model is expressed as:

$$\mathbf{x}_{k+1} = \mathbf{x}_k \boxplus (\Delta t_k \mathbf{f}(\mathbf{x}_k, \mathbf{w}_k)),$$
$$\mathbf{f}(\mathbf{x}_k, \mathbf{w}_k) = \begin{bmatrix} {}^{I}\boldsymbol{\omega}_k \\ {}^{W}\mathbf{v}_{I_k} \\ {}^{W}\mathbf{R}_{I_k} \cdot {}^{I}\mathbf{a}_k + {}^{W}\mathbf{g}_k \\ \mathbf{n}_{\mathbf{b}\boldsymbol{\omega}_k} \\ \mathbf{n}_{\mathbf{b}\mathbf{a}_k} \\ \mathbf{w}_{\boldsymbol{\omega}} \\ \mathbf{w}_{\mathbf{a}} \\ {}^{W}\mathbf{R}_{I_k} \cdot \mathbf{h}_{\mathbf{R}}(\boldsymbol{\alpha}_k) \cdot \mathbf{w}_{\mathbf{p}_c} \\ \mathbf{0}_{3\times 1} \end{bmatrix} \in \mathbb{R}^{27} \tag{3}$$

where $\mathbf{h}_{\mathbf{R}}(\boldsymbol{\alpha}_k)$ is the measured orientation from the contact frame to the IMU frame through leg forward kinematics.

The predicted state $\hat{\mathbf{x}}_{k+1}$ is propagated from the last updated state $\bar{\mathbf{x}}_k$ by (3) with noise $\mathbf{w}_k = 0$:

$$\hat{\mathbf{x}}_{k+1} = \bar{\mathbf{x}}_k \boxplus (\Delta t_k \mathbf{f}(\bar{\mathbf{x}}_k, \mathbf{0})) \tag{4}$$

And the error state $\tilde{\mathbf{x}}$ and the prior covariance $\hat{\mathbf{P}}_k$ after linearization can be described as follows:

$$\begin{aligned}\tilde{\mathbf{x}}_{k+1} &= \mathbf{x}_{k+1} \boxminus \hat{\mathbf{x}}_{k+1} \\ &\simeq \mathbf{F}_{\tilde{\mathbf{x}}_k}\tilde{\mathbf{x}}_k + \mathbf{F}_{\mathbf{w}_k}\mathbf{w}_k \\ \hat{\mathbf{P}}_{k+1} &= \mathbf{F}_{\tilde{\mathbf{x}}_k}\bar{\mathbf{P}}_k\mathbf{F}_{\tilde{\mathbf{x}}_k}^T + \mathbf{F}_{\mathbf{w}_k}\mathcal{Q}_k\mathbf{F}_{\mathbf{w}_k}^T\end{aligned} \tag{5}$$

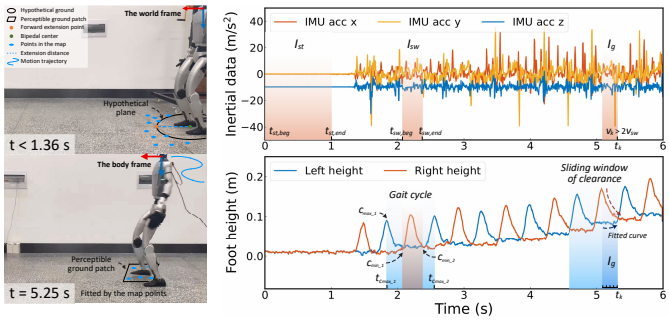

Fig. 2. Conatact detection process. The left shows the humanoid robot in the initial and walking phases. The right shows the relevant variables and the process of contact detection for the right foot.

where $\mathbf{F}_{\tilde{\mathbf{x}}_k}$ and $\mathbf{F}_{\mathbf{w}_k}$ are the Jacobian matrices of $\mathbf{x}_{k+1} \boxminus \hat{\mathbf{x}}_{k+1}$ with respect to $\tilde{\mathbf{x}}_k$ and $\mathbf{w}_k$.

C. Foot Contact Detection

1) Foot Clearance Sliding Window: We assume that the ground patch near the foothold is flat, and further fit the local plane by map points to calculate the foot-ground clearance. However, due to the limited field of view (FoV) of LiDAR, no points are available around the feet when the robot is stationary or moving within a small range. A hypothetical plane is thus introduced to replace the real ground when it cannot be perceived. As shown in Fig. 2, the perceptible ground patch is fitted by the map points, and the hypothetical plane is fitted by a set $\mathcal{S}_p$ of four points:

$$\mathcal{S}_p = \{{}^{W}\mathbf{p}_{fk_l}, {}^{W}\mathbf{p}_{fk_r}, {}^{W}\mathbf{p}_{fk_f}, {}^{W}\mathbf{p}_{fk_b}\} \tag{6}$$

where ${}^{W}\mathbf{p}_{fk_l}$ is the left foot position in the world frame transformed from ${}^{B}\mathbf{p}_{fk_l}$, which is measured from the leg forward kinematics function by joint encoders, ${}^{W}\mathbf{p}_{fk_r}$ is similar. ${}^{W}\mathbf{p}_{fk_f}$ and ${}^{W}\mathbf{p}_{fk_b}$ are the forward and backward extension points, respectively, defined as:

$$\{{}^{W}\mathbf{p}_{fk_f}, {}^{W}\mathbf{p}_{fk_b}\} = {}^{W}\mathbf{p}_{fk_m} \pm d\,{}^{W}\mathbf{R}_F\mathbf{e}_x \tag{7}$$

where ${}^{W}\mathbf{p}_{fk_m}$ is the bipedal center in the world frame, $d = \frac{1}{m}\sum_{i=1}^{m}\|{}^{W}\mathbf{p}_i - {}^{W}\mathbf{p}_{fk_m}\|$ is the extension distance, ${}^{W}\mathbf{p}_i$ is the i-th nearest point for ${}^{W}\mathbf{p}_{fk_m}$ searched in the map, and m is their number, $\mathbf{e}_x = [1,0,0]^T$ and ${}^{W}\mathbf{R}_F$ is the known rotation from the robot fixed frame (with x, y and z axes point forward, left and up of the robot) to the world frame.

Next, we denote ${}^{W}\mathbf{u}_{hypo}$ and ${}^{W}\mathbf{u}_{real}$ are the normal vectors of the hypothetical and the perceptible ground, ${}^{W}\mathbf{q}_{hypo}$ and ${}^{W}\mathbf{q}_{real}$ are any points on their corresponding planes, $d_r = \|{}^{W}\mathbf{p}_{fk_l} - {}^{W}\mathbf{p}_{fk_r}\|$ is the bipedal clearance. The foot-ground clearance is computed by the distance from the foot position to the corresponding local ground patch:

$$c_i = \begin{cases} {}^{W}\mathbf{u}_{hypo}^T({}^{W}\mathbf{p}_{fk_i} - {}^{W}\mathbf{q}_{hypo}), & d \geq \frac{1}{2}d_r \\ {}^{W}\mathbf{u}_{real}^T({}^{W}\mathbf{p}_{fk_i} - {}^{W}\mathbf{q}_{real}), & d < \frac{1}{2}d_r. \end{cases} \tag{8}$$

Notice that once the real ground is detected, the hypothetical ground is discarded, while the perceptible ground is updated in real time.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

The calculated foot clearance c_i is saved to a sliding window $\mathcal{C}$, which is maintained at the size of a gait cycle duration g. It is determined during the first two swing phases of one leg, for example, the left leg is used in our system:

$$g = t_{c_{max\_2}} - t_{c_{max\_1}} \tag{9}$$

where $t_{c_{max\_1}}$ and $t_{c_{max\_2}}$ are the time of the maximum foot clearance during the first and second leg swings, respectively.

2) IMU Fluctuation & Contact Classification: We utilize acceleration fluctuations from ground impacts to trigger contact detection. The reference variances during the stance and swing phase are calculated as the thresholds. As shown in Fig. 2, the time of the first IMU measurement is denoted as $t_{st,beg}$, with $t_{st,end}$ one second later. The single-leg swing phase spans from $t_{sw,beg}$ to $t_{sw,end}$, calculated as follows:

$$\begin{aligned} c_{min\_1} &= \min_{1\le i\le g/2} \Delta\left|c_{l,i} - c_{r,i}\right|, \\ c_{min\_2} &= \min_{g/2<i\le g} \Delta\left|c_{l,i} - c_{r,i}\right|, \\ t_{sw,beg} &= t_{c_{min\_1}}, t_{sw,end} = t_{c_{min\_2}}. \end{aligned} \tag{10}$$

Then, we denote the reference variances of stance and swing as v_{st} and v_{sw}, and compute them based on acceleration measurements $\mathbf{a}$:

$$v_{st} = \frac{1}{|I_{st}|}\sum_{i\in I_{st}} \|\mathbf{a}_i - \bar{\mathbf{a}}\|^2, v_{sw} = \frac{1}{|I_{sw}|}\sum_{i\in I_{sw}} \|\mathbf{a}_i - \bar{\mathbf{a}}\|^2 \tag{11}$$

where $I_{st} = \{i \mid t_{st,beg} \le t_i \le t_{st,end}\}$ and $I_{sw} = \{i \mid t_{sw,beg} \le t_i \le t_{sw,end}\}$, $|I_{st}|$ and $|I_{sw}|$ is the number of elements in I_{st} and I_{sw}, respectively.

The current variance of acceleration v_k is computed and compared with the reference variances:

$$\begin{aligned} v_k &= \frac{1}{|I_g|}\sum_{i\in I_g} \|\mathbf{a}_i - \bar{\mathbf{a}}\|^2 \\ I_g &= \{i \mid t_k - \frac{g}{s} \le t_i \le t_k\} \end{aligned} \tag{12}$$

where s is set to 5 for slicing the gait cycle.

When $v_k < 2v_{st}$, the robot is considered in a stance state. When $v_k > 2v_{sw}$, a contact event is assumed to have occurred, and the clearance curves are subsequently utilized to classify the contact. We sample five foot clearance values at equal time intervals within a slice of $\mathcal{C}$. These data are used for curve fitting of quadratic polynomials, where the first derivative provides the corresponding slopes, taking the left foot $\mathcal{S}_l$ as an example:

$$\mathcal{S}_l = \{f_l'(i) = 2a_l \cdot i + b_l \mid i \in [1,5] \cap \mathbb{Z}\} \tag{13}$$

The calculation of the slopes for the right foot $\mathcal{S}_r$ is similar. By comparing the slopes of the clearance curves $\mathcal{S}_l$ and $\mathcal{S}_r$ at each corresponding point i, the contact event at t_k can be determined by the number of the smaller slopes.

D. Residual Computation

1) Leg Kinematic Measurement: The kinematic measurements of the foot in contact with the ground are used for residual computation. The contact foot velocity measurement is modeled as:

$$\begin{aligned} \mathbf{0} =& \mathbf{h}_{CV}(\mathbf{x}, \mathbf{n}_{cv}) \\ \simeq& -{}^{W}\mathbf{R}(\lfloor\boldsymbol{\omega}\rfloor_\wedge {}^{B}\mathbf{p}_{fk} + {}^{B}\mathbf{v}_{fk}) - {}^{W}\mathbf{v} + \mathbf{n}_{cv} \end{aligned} \tag{14}$$

where ${}^{B}\mathbf{p}_{fk}$ and ${}^{B}\mathbf{v}_{fk}$ are the foot position and velocity in the body frame from forward kinematics function, $\mathbf{n}_{cv} \approx \mathcal{N}(\mathbf{0}, \mathcal{R}_{cv})$ is the noise caused by imprecise modeling.

A residual $\mathbf{r}_{CV}$ is further computed as:

$$\begin{aligned} \mathbf{r}_{CV} &= \mathbf{0} - \mathbf{h}_{CV}(\hat{\mathbf{x}}, \mathbf{0}) \\ &= \mathbf{h}_{CV}(\mathbf{x}, \mathbf{n}_{cv}) - \mathbf{h}_{CV}(\hat{\mathbf{x}}, \mathbf{0}) \\ &\simeq \mathbf{H}_{CV}\tilde{\mathbf{x}} + \mathbf{D}_{CV}\mathbf{n}_{cv} \end{aligned} \tag{15}$$

For the contact foot position measurement, we get the foot position in the body frame ${}^{B}\mathbf{p}_{fk}$ based on the state vector with the measurement noise $\mathbf{n}_{cp} \approx \mathcal{N}(\mathbf{0}, \mathcal{R}_{cp})$:

$${}^{B}\mathbf{p}_{fk} = \mathbf{h}_{CP}(\mathbf{x}, \mathbf{n}_{cp}) = {}^{W}\mathbf{R}^T({}^{W}\mathbf{p}_c - {}^{W}\mathbf{p}) + \mathbf{n}_{cp} \tag{16}$$

Then, a residual $\mathbf{r}_{CP}$ is computed as:

$$\begin{aligned} \mathbf{r}_{CP} =& {}^{B}\mathbf{p}_{fk} - \mathbf{h}_{CP}(\hat{\mathbf{x}}, \mathbf{0}) \\ =& \mathbf{h}_{CP}(\mathbf{x}, \mathbf{n}_{cp}) - \mathbf{h}_{CP}(\hat{\mathbf{x}}, \mathbf{0}) \\ \simeq& \mathbf{H}_{CP}\tilde{\mathbf{x}} + \mathbf{D}_{CP}\mathbf{n}_{cp} \end{aligned} \tag{17}$$

For the contact foot height measurement, we introduce it to reduce the z-axis drift [5]. During a foot is in contact with the ground, it is assumed to lie on its local ground patch. However, this assumption is not applicable to the hypothetical ground due to the absence of realistic local features. For perceptible ground, we search the 15 nearest map points of ${}^{W}\mathbf{p}_c$ to fit a local ground patch and compute the projection point ${}^{W}\mathbf{p}_{c_{proj}}$:

$${}^{W}\mathbf{p}_{c_{proj}} = ({}^{W}\mathbf{p}_c - \mathbf{u}^T({}^{W}\mathbf{p}_c - {}^{W}\mathbf{q})\mathbf{u}) - {}^{W}\mathbf{p}_{fk} \tag{18}$$

where $\mathbf{u}$ and ${}^{W}\mathbf{q}$ are the normal vector and a point lying on the ground patch, respectively.

Then, define the contact foot height measurement as:

$$\begin{aligned} \mathbf{0} =& \mathbf{h}_{CH}(\mathbf{x}, \mathbf{n}_{ch}) \\ \simeq& \mathbf{e}_z^T {}^{W}\mathbf{p}_{c_{proj}} - \mathbf{e}_z^T {}^{W}\mathbf{p}_{fk} + \mathbf{n}_{ch} \end{aligned} \tag{19}$$

where $\mathbf{e}_z^T = [0,0,1]$, and $\mathbf{n}_{ch} \approx (\mathbf{0}, \mathcal{R}_{ch})$.

A residual $\mathbf{r}_{CH}$ can be further computed as:

$$\begin{aligned} \mathbf{r}_{CH} =& \mathbf{0} - \mathbf{h}_{CH}(\hat{\mathbf{x}}, \mathbf{0}) \\ =& \mathbf{h}_{CH}(\mathbf{x}, \mathbf{n}_{ch}) - \mathbf{h}_{CH}(\hat{\mathbf{x}}, \mathbf{0}) \\ \simeq& \mathbf{H}_{CH}\tilde{\mathbf{x}} + \mathbf{D}_{CH}\mathbf{n}_{ch} \end{aligned} \tag{20}$$

The above Jacobian matrices are defined as:

$$\mathbf{H} = \left.\frac{\partial \mathbf{h}(\hat{\mathbf{x}} \boxplus \tilde{\mathbf{x}}, \mathbf{0})}{\partial \tilde{\mathbf{x}}}\right|_{\tilde{\mathbf{x}}=\mathbf{0}}, \mathbf{D} = \left.\frac{\partial \mathbf{h}(\hat{\mathbf{x}}, \mathbf{n})}{\partial \mathbf{n}}\right|_{\mathbf{n}=\mathbf{0}} \tag{21}$$

According to the leg kinematic measurements functions of (14), (16), and (19), the Jacobian matrices are computed as follows:

$$\begin{aligned} \mathbf{H}_{CV} =& [{}^{W}\hat{\mathbf{R}}\lfloor\lfloor\boldsymbol{\omega}\rfloor_\wedge {}^{B}\mathbf{p}_{fk} + {}^{B}\mathbf{v}_{fk}\rfloor_\wedge \quad \mathbf{0}_{3\times3} \\ & -\mathbf{I}_{3\times3} \quad \mathbf{0}_{3\times9} \quad {}^{W}\hat{\mathbf{R}}\lfloor{}^{B}\mathbf{p}_{fk}\rfloor_\wedge \quad \mathbf{0}_{3\times6}] \\ \mathbf{H}_{CP} =& [-{}^{W}\hat{\mathbf{R}}\lfloor{}^{B}\mathbf{p}_{fk}\rfloor_\wedge \quad \mathbf{I}_{3\times3} \quad \mathbf{0}_{3\times15} \quad -\mathbf{I}_{3\times3} \quad \mathbf{0}_{3\times3}] \\ \mathbf{H}_{CH} =& [\mathbf{0}_{1\times21} \quad [0,0,1] \quad \mathbf{0}_{1\times3}] \\ \mathbf{D}_{CV} =& \mathbf{D}_{CP} = \mathbf{I}_{3\times3}, \mathbf{D}_{CH} = 1 \end{aligned} \tag{22}$$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

2) IMU Measurement: An IMU measurement is first assessed through a saturation check. The IMU data, which remains within the measurement range, is used to compute residuals $\mathbf{r}_I$ according to the measurement function $\mathbf{h}_I(\mathbf{x}, \mathbf{w}_I)$. This process can be expressed as:

$$\begin{aligned} \begin{bmatrix} {}^I\boldsymbol{\omega}_m \\ {}^I\mathbf{a}_m \end{bmatrix} &= \mathbf{h}_I(\mathbf{x}, \mathbf{n}_I) = \begin{bmatrix} {}^I\boldsymbol{\omega} + \mathbf{b}_{\boldsymbol{\omega}} + \mathbf{n}_{I_\omega} \\ {}^I\mathbf{a} + \mathbf{b}_{\mathbf{a}} + \mathbf{n}_{I_a} \end{bmatrix} \\ \mathbf{r}_I &= \begin{bmatrix} {}^I\boldsymbol{\omega}_m \\ {}^I\mathbf{a}_m \end{bmatrix} - \mathbf{h}_I(\hat{\mathbf{x}}, \mathbf{0}) \\ &= \mathbf{h}_I(\mathbf{x}, \mathbf{n}_I) - \mathbf{h}_I(\hat{\mathbf{x}}, \mathbf{0}) \\ &= \mathbf{H}_I \tilde{\mathbf{x}} + \mathbf{D}_I \mathbf{n}_I \end{aligned} \tag{23}$$

where $\mathbf{n}_{I_\omega} \approx \mathcal{N}(\mathbf{0}, \mathcal{R}_\omega)$ and $\mathbf{n}_{I_a} \approx \mathcal{N}(\mathbf{0}, \mathcal{R}_a)$ are noises that follow Gaussian distributions, $\mathbf{n}_I = \begin{bmatrix} \mathbf{n}_\omega^T & \mathbf{n}_a^T \end{bmatrix}^T \approx \mathcal{N}(\mathbf{0}, diag(\mathcal{R}_\omega, \mathcal{R}_a))$ is the IMU noise, and

$$\begin{aligned} \mathbf{H}_I &= \begin{bmatrix} \mathbf{0}_{6\times 9} & \mathbf{I}_{6\times 12} & \mathbf{0}_{6\times 6} \end{bmatrix} \\ \mathbf{D}_I &= \mathbf{I}_{6\times 6} \end{aligned} \tag{24}$$

3) LiDAR Measurement: The LiDAR measurements are modeled in the same way as [13]. For per true point ${}^L\mathbf{p}^{GT} = {}^L\mathbf{p}_m - \mathbf{w}_L$ from LiDAR, it should lie on a local patch in the world frame, that is:

$$\begin{aligned} \mathbf{0} &= \mathbf{h}_L(\mathbf{x}, {}^L\mathbf{p}_m, \mathbf{n}_L) \\ &\simeq {}^W\mathbf{u}^T({}^W\mathbf{T}_I \, {}^I\mathbf{T}_L({}^L\mathbf{p}_m - \mathbf{n}_L) - {}^W\mathbf{q}) \end{aligned} \tag{25}$$

where ${}^W\mathbf{u}$ and ${}^W\mathbf{q}$ are the normal vector and any point of the small patch, ${}^W\mathbf{T}_I$ implicitly contains the state vector $\mathbf{x}$, and ${}^I\mathbf{T}_L$ is the known extrinsic from LiDAR to IMU.

During the LiDAR residual computation, the measured point ${}^L\mathbf{p}_m$ is projected to ${}^W\hat{\mathbf{p}}$ by the extrinsic ${}^I\mathbf{T}_L$ and the predicted pose ${}^W\hat{\mathbf{T}}_I$. We search 5 nearest points to ${}^W\hat{\mathbf{p}}$ in the historical map, which is organized by incremental voxels. After successfully fitting the local patch, a residual $\mathbf{r}_L$ is:

$$\begin{aligned} \mathbf{r}_L &= 0 - \mathbf{h}_L(\hat{\mathbf{x}}, {}^L\mathbf{p}_m, \mathbf{0}) \\ &= \mathbf{h}_L(\mathbf{x}, {}^L\mathbf{p}_m, \mathbf{n}_L) - \mathbf{h}_L(\hat{\mathbf{x}}, {}^L\mathbf{p}_m, \mathbf{0}) \\ &\simeq \mathbf{H}_L \tilde{\mathbf{x}} + \mathbf{D}_L \mathbf{n}_L \end{aligned} \tag{26}$$

where $\mathbf{n}_L \approx \mathcal{N}(\mathbf{0}, \mathcal{R}_L)$ and

$$\begin{aligned} \mathbf{H}_L &= {}^W\mathbf{u}^T \begin{bmatrix} -{}^W\hat{\mathbf{R}}_I \lfloor {}^I\mathbf{R}_L \, {}^L\mathbf{p}_m \rfloor_\wedge & \mathbf{I}_{1\times 3} & \mathbf{0}_{1\times 21} \end{bmatrix} \\ \mathbf{D}_L &= -{}^W\mathbf{u}^T \, {}^W\hat{\mathbf{R}}_I \end{aligned} \tag{27}$$

E. State Update

Based on the prior distribution in (5) and the measurement models in (14), (16), (19), (23), and (25), the Kalman gain $\mathbf{K}$ and the updated state $\bar{\mathbf{x}}_{k+1}$ with its covariance $\bar{\mathbf{P}}_{k+1}$ are computed as follows:

$$\begin{aligned} \mathbf{K} &= \hat{\mathbf{P}}\mathbf{H}^T(\mathbf{H}\hat{\mathbf{P}}\mathbf{H}^T + \mathbf{D}\mathcal{R}\mathbf{D}^T)^{-1} \\ \bar{\mathbf{x}}_{k+1} &= \hat{\mathbf{x}}_{k+1} \boxplus \mathbf{K}_{k+1}\mathbf{r}_{k+1} \\ \bar{\mathbf{P}}_{k+1} &= (\mathbf{I} - \mathbf{K}\mathbf{H}_{k+1})\hat{\mathbf{P}}_{k+1} \end{aligned} \tag{28}$$

The updated state at different measurements is output directly, thus realizing high-rate output. If the update process at t_{k+1} is performed using a point from LiDAR, the point will be transformed to the world frame and then appended to the incremental voxel map.

TABLE II
DATASET SUMMARY

Robots	Places	Sensors
BHR-B3	Laboratory (walk)	Velodyne VLP16 LiDAR (100 m range, 360°×30° FoV, 10 Hz) Xsens Mti-100 IMU(200 Hz) HDH EBI-1135 Encoder (1000 Hz) SRI F/T sensor (1000 Hz)
Unitree G1	Laboratory/ corridors (walk/run/spin)	Livox Mid360 LiDAR (70 m range, 360°×59° FoV, 10 Hz) ICM40609 IMU (200 Hz) Self-developed Encoder (1000 Hz)

V. EXPERIMENTS

A. Dataset & Implementation

To verify the effectiveness of the proposed method, extensive experiments are conducted on the public humanoid robot dataset and tested in the real world. The public dataset we selected is from the work LIKO [8] with the ground truth collecting by VICON MoCap. For a comprehensive evaluation, we also perform real-world tests on Unitree G1 in *long corridor*, *switchback running*, etc., while collecting these data. The dataset summary is shown in Table II, and more details are in Supplementary Material [22].

We develop HR^2-KILO inspired by Point-LIO [13] based on C++ and robot operating system (ROS). The leg kinematic computation is based on Pinocchio [23]. The experiments on the public dataset are implemented on a desktop (Intel i5 13490F CPU and 16-GB RAM), and real-world tests are performed on a laptop (Intel i9-13900H and 32-GB RAM). For the sake of fairness, the experimental parameters are configured as the respective recommendations, where downsampled parameters (0.5 m) and sensor noise are set the same as possible for all the methods.

B. Contact Estimation

This experiment is divided into two parts: contact event detection and contact foot position estimation. We evaluate complete contact events in the public dataset and real-world tests. The compared methods include deep contact estimator (DCE) [18], and thresholding GRF, which is calculated similarly to [15], [16]. We use a decimal $\mathcal{S}_c \in \{-1, 0, 1, 2, 3\}$ to represent uncertain, none, right, left, and both contact. The comparison on *square walk* and *soft terrain* are shown in Fig. 3. It can be seen that DCE experiences degradation when the robot contacts the soft terrain (after 15 seconds) due to insufficient training on various terrains. The high-frequency noise in GRF necessitates smoothing procedures, which introduces significant detection delays, and elaborate thresholding and the body state are also required. In contrast, the proposed method detects contact events robustly without additional data and training. Notice that the first two contacts are used to determine the gait cycle for adaptation to multiple humanoid robot platforms. And the leg kinematic information will not be used for state update when contact is uncertain to avoid incorrect constraints. Table III provides the quantitative results for more sequences, including the evaluation of detected events and the performance within the state estimation

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

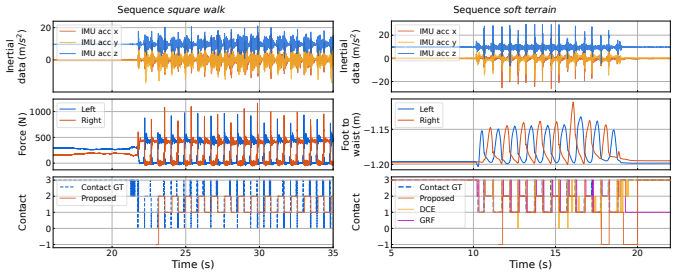

Fig. 3. Contact detection on *square walk* and *soft terrain* sequence, including inertial data, force data, foot-to-waist height, and contact.

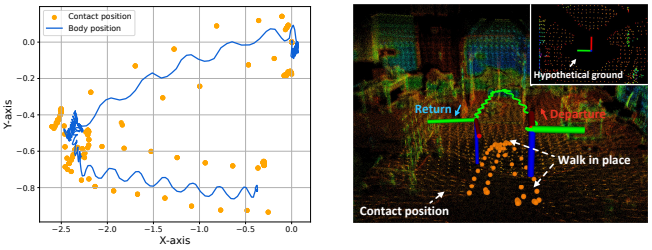

Fig. 4. Contact position. The left shows estimated body and contact foot positions. The right shows the body trajectories with contact positions in the 3D map, the upper right indicates the hypothetical ground replaces the actual ground in the initial phase of motion.

TABLE III
CONTACT DETECTION RESULTS

Seq	Proposed		DCE	GRF	GT
	L/R	U	L/R	L/R	L/R
fwd. bwd.	74/73	0	—[1]	—	74/73
square walk	86/82	4	—	—	86/85
square walk l.	134/134	9	—	—	163/163
walk in place	202/202	1	—	—	202/202
up slope	73/71	2	—	—	73/72
walk & back	14/13	2	14/11	—	15/14
fwd. running	7/6	0	6/3	7/6	7/6
soft terrain	7/7	1	7/5	8/7	8/7
FPR[2] (%)	3.34		2.22	4.27	—
Accuracy (%)	92.57		95.60	68.68	—

[1]The sequences does not produce the joint position, velocity or torque data, which is necessary for DCE and GRF.
[2]The FPR and accuracy of contact detection for state estimation.

context. These demonstrate our method's effectiveness and generality, more detailed experiments and analysis can be found in Supplementary Material [22].

Then, we qualitatively analyze estimated foot positions in the real-world test. As shown in Fig. 4, the robot steps at the start point, walks about 2.5 m and setps in place again, then returns. It can be seen that the estimated contact positions are consistent with the actual situation. The backward offset of the body and contact position during stepping are caused by the controller, also can be estimated successfully.

C. Accuracy

1) Benchmark Results: In this section, a total of 5 sequences with ground truth from the public dataset are used to evaluate the accuracy. The proposed method is compared with the state-of-the-art LiDAR-inertial(-kinematic) methods, including LIO-SAM (without closed-loop correction), FAST-LIO2, Point-LIO, and LIKO. "Point-LIO (H)" and "Proposed (H)" specify the high-rate pattern, "Point-LIO" and "Proposed" refer to the output at LiDAR frame rate (10 Hz). We use the root-mean-square error (RMSE) of absolute pose error (APE) and relative pose error (RPE) as the metrics.

The benchmark results of accuracy evaluation are shown in Table IV. On 4 of 5 sequences, the proposed method achieves the lowest error in terms of APE. LIO-SAM performs best on *forward backward* sequence, but our method is second-best. For the RMSE of RPE, our proposed method also performs best on 4 of 5 sequences. For *up slope* sequence, LIKO achieves the lowest drift (0.0013%), while our method is still competitive with 0.0015% drift. By integrating multiple leg kinematic measurements and the pointwise update strategy into the system, our odometry is more suitable for humanoid robots, which can robustly cope with aggressive motions and improve accuracy.

TABLE IV
BENCHMARK RESULTS

Methods	*fwd. bwd.*	*square walk*	*square walk l.*	*walk in place*	*up slope*
APE RMSE (m) Results					
LIO-SAM	**0.0047**	0.0226	0.0191	0.0096	1.5001
FAST-LIO2	0.0125	0.0296	0.0174	0.0302	0.0311
Point-LIO	0.0123	0.0151	0.0138	0.0069	0.0296
Point-LIO (H)	0.0147	0.0174	0.0142	0.0070	0.0274
LIKO	0.0126	0.0250	0.0242	0.0301	0.0372
Proposed	0.0073	0.0106	**0.0125**	**0.0063**	**0.0262**
Proposed (H)	0.0076	**0.0105**	0.0134	0.0134	0.0281
RPE RMSE (%) Results					
LIO-SAM	0.0312	0.0379	0.0286	0.0208	0.4463
FAST-LIO2	0.0064	0.0081	0.0079	0.0073	0.0100
Point-LIO	0.0089	0.0111	0.0099	0.0068	0.0116
Point-LIO (H)	0.0050	0.0067	0.0018	0.0014	0.0018
LIKO	**0.0012**	0.0015	0.0020	0.0016	**0.0013**
Proposed	0.0064	0.0083	0.0094	0.0066	0.0103
Proposed (H)	**0.0012**	**0.0013**	**0.0016**	**0.0012**	0.0015

2) Ablation Study: Here, we study the individual contribution of each leg kinematic measurement. We denote the proposed method without leg kinematic measurements as "w/o kinematics", while "full kinematics" means all kinematic measurements are used. The case utilizing only one single leg kinematic constraint is named "contact velocity", "contact position", or "contact height". The results are reported in Table V, "full kinematics" achieves the lowest absolute pose error, both "contact velocity" and "contact position" show a positive effect for accuracy. In contrast, "w/o kinematics" exhibits significant degradation in accuracy, and its pose error is close to Point-LIO. In the sequences of the public dataset, the robot's foot motion is confined to the LiDAR FoV blindness. Therefore, the hypothetical ground persists throughout the process, which renders the contact height measurement unutilized.

3) Real World Test: To comprehensively evaluate the odometry performance of humanoid robots, we extend the scenarios to the real world. *long corridor* and *short corridor* verify the long-term performance, *switchback running* and *spinning motion* reflect the robustness under the high-dynamic motion.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

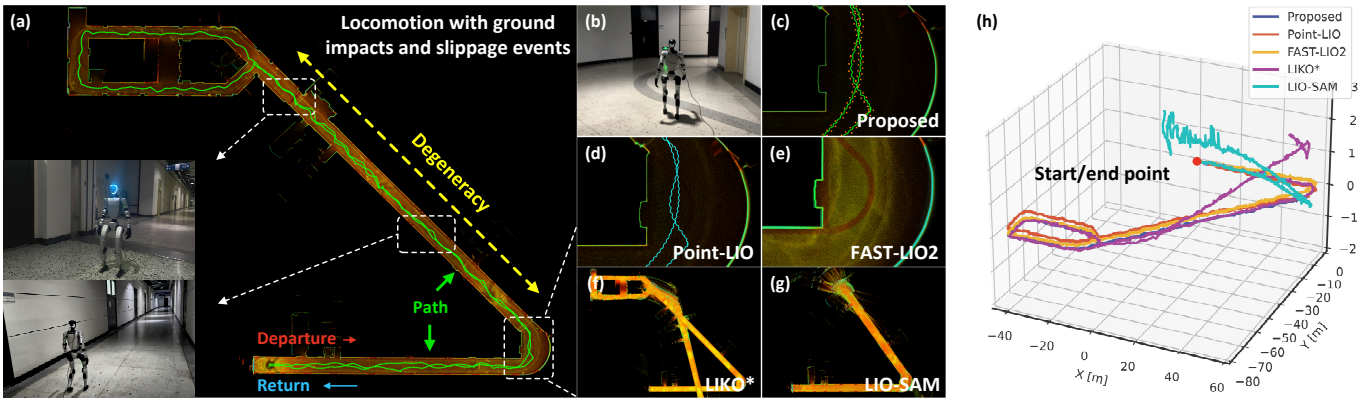

Fig. 5. Experiments on *long corridor*. (a) presents a global view of the mapping results and trajectories for our proposed method. (b) shows G1 is walking around the turn in the return phase. (c) shows the mapping details of the proposed method, the orange dots are contact positions and the green curve is the estimated trajectory. (d), (e), (f), and (g) are the results of compared methods. (h) shows the estimated trajectories of different methods.

TABLE V
APE RMSE (M) FOR ABLATION STUDY

Configuration	*square walk*	*walk in place*	*up slope*
w/o kinematics	0.0110	0.0068	0.0299
contact velocity	0.0106	0.0066	0.0287
contact position	0.0109	0.0067	0.0281
contact height	—[1]	—	—
full kinematics	0.0106	0.0063	0.0262

[1]Contact foot height is passivey unused.

TABLE VI
END-TO-END ERROR (M) ON THE Z-AXIS

Methods	*long corridor*	*short corridor*	*switchback running*	*spinning motion*
LIO-SAM	×[1]	0.0140	0.0187	0.0090
LIKO*	×	0.0090	0.0227	0.0086
FAST-LIO2	0.0178	0.0093	0.0251	0.0063
Point-LIO	0.0069	0.0013	0.0033	0.0038
Proposed	**0.0062**	**0.0012**	**0.0019**	**0.0027**

[1]× denotes that the system failed.

All the sequences start from and end at the same location. Additionally, G1 is not equipped with contact sensors, causing LIKO to operate without leg kinematic information in this section, which we denote as LIKO*.

As the start and the end points are not exactly coincident, the end-to-end error on the z-axis is used for evaluation. The results are shown in Table VI, the proposed method achieves the lowest errors on all sequences, with a mean end-to-end error of 0.003 m on the z-axis. Point-LIO follows closely with a mean error of 0.0038 m. LIKO* and LIO-SAM failed on *long corridor* sequence, with mean errors of 0.0134 m and 0.0139 m for other sequences.

More detailed information on *long corridor* is provided in Fig 5. The robot departs from the end of a corridor and returns to the same location, traveling the degraded scenario while suffering ground impacts. Our method shows accurate localization and mapping performance in this experiment (Fig. 5(a)). At the turn of the return phase, the proposed method produces a thinner wall than Point-LIO, see Fig. 5(c) and Fig. 5(d). The wall produced by FAST-LIO2 is slightly distorted, which reduces the map consistency. Besides degraded scenarios, we found that the slippages during walking disrupt the robot's balance, resulting in aggravated and irregular ground impacts. This further makes state estimation more challenging and might make LIKO* and LIO-SAM fail. On *spinning motion*, the experiment lasts a total of 58.24 s, with the spin starting at 8.94 s and ending at 54.78 s. Fig. 6 provides the estimated position on the z-axis for different methods. It can be demonstrated that the proposed method achieves more stable estimation and lower errors.

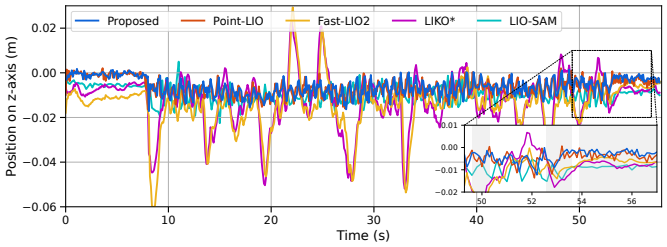

Fig. 6. Estimated positions on the z-axis for different methods on *spinning motion*.

To further demonstrate the robustness of the proposed framework, we conduct the experiment simulating the sensor corruption. As shown in Fig. 7, temporary dropouts and intermittent anomalies in LiDAR or encoder data occur in the sequence. In these cases, HR^2-KILO is resilient to maintain state estimation and constrains the drift by leveraging the other data. The system works robustly with achieving 0.0029 m end-to-end error on Z-axis, while other LiDAR-inertial(-kinematic) methods are unable to complete this task.

D. Output Rate & Efficiency

In this section, we analyze the output rate and efficiency of our method. The output rates with different downsampled parameters are shown in Table VII. By processing LiDAR point by point and integrating high-frequency kinematic information, the proposed odometry achieves the output of at least 6 kHz, even with the 0.5 m downsampling parameters. When the parameter is reduced to 0.2 m, the real-time performance of the system is degraded due to the increased number of points involved in state update. We also test the average time usage of each step for processing one LiDAR scan (see Fig. 8). With

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

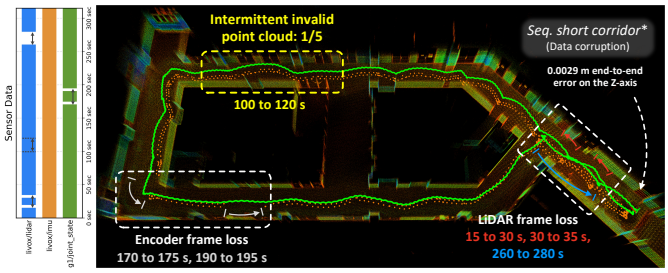

Fig. 7. Experiments on *short corridor**. Some of sensor data is filtered out at given periods for robustness testing.

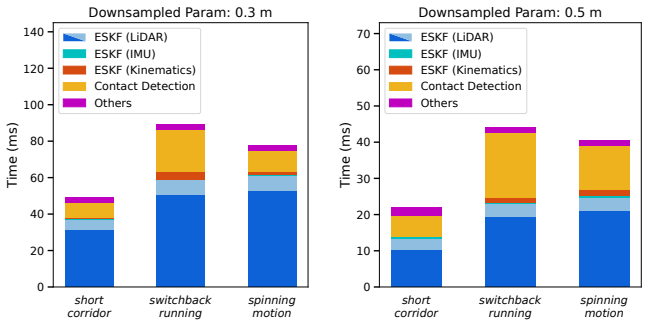

Fig. 8. Average time usage per scan. Blue, cyan, and orange are the time taken by ESKF to process LiDAR, IMU, and leg kinematics data, respectively. In ESKF (LiDAR), the dark blue shows the time of point search, while light blue shows the remaining time.

TABLE VII
AVERAGE OUTPUT RATE (KHZ)

Methods	Downsampled Parameter	*short corridor*	*switchback running*	*spinning motion*
Proposed (H)	0.2 m	19	16	18
	0.3 m	14	15	15
	0.5 m	6	7	7

the default downsampled parameters (0.5 m), the total time consumption is less than 50 ms. The majority of the time is taken by contact detection and LiDAR processing due to the intensive search for map points.

VI. CONCLUSION AND FUTURE WORK

This letter focuses on the performance and requirements of state estimation on humanoid robots. We propose a high-rate and robust odometry by tightly coupling joint encoder, IMU, and LiDAR. The framework is based on the ESKF with pointwise update, an IMU measurement model, and leg kinematic constraints. In addition, we utilize multi-source data to detect contact without contact sensors. Extensive experiments demonstrate that the proposed method achieves better performance compared to the state-of-the-art LiDAR-inertial(-kinematic) methods. Visual sensors are expected to be considered to further improve the odometry performance.

REFERENCES

[1] X. Gu *et al.*, "Advancing humanoid locomotion: Mastering challenging terrains with denoising world model learning," in *Proc. Robot.: Sci. Syst.*, July 2024.

[2] M. Bloesch, C. Gehring, P. Fankhause, M. Hutter, M. A. Hoepflinger, and R. Siegwart, "State estimation for legged robots on unstable and slippery terrain," in *Proc. 2013 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 6058–6064.

[3] N. Rotella, M. Bloesch, L. Righetti, and S. Schaal, "State estimation for a humanoid robot," in *Proc. 2014 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 952–958.

[4] 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.

[5] 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.

[6] J. Kang, Y. Wang, and X. Xiong, "Fast decentralized state estimation for legged robot locomotion via ekf and mhe," *IEEE Robot. Autom. Lett.*, vol. 9, no. 12, pp. 10914–10921, 2024.

[7] D. Wisth, M. Camurri, and M. Fallon, "Vilens: visual, inertial, lidar and leg odometry for all-terrain legged robots," *IEEE Trans. Robot.*, vol. 39, no. 1, pp. 309–326, Feb. 2023.

[8] Q. Zhao *et al.*, "Liko: Lidar, inertial, and kinematic odometry for bipedal robots," in *Proc. 2024 IEEE Int. Conf. Robot. Autom.*, 2024, pp. 1180–1185.

[9] F. E. Xavier, G. Burger, M. Ptriaux, J.-E. Deschaud, and F. Goulette, "Multi-imu proprioceptive state estimator for humanoid robots," in *Proc. 2023 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2023, pp. 10880–10887.

[10] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *Proc. 2020 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5135–5142.

[11] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct, lidar-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, 2022.

[12] C. Bai, T. Xiao, Y. Chen, H. Wang, F. Zhang, and X. Gao, "Faster-lio: Lightweight tightly coupled lidar-inertial odometry using parallel sparse incremental voxels," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4861–4868, 2022.

[13] D. He, W. Xu, N. Chen, F. Kong, C. Yuan, and F. Zhang, "Point-lio: Robust high-bandwidth light detection and ranging inertial odometry," *Adv. Intell. Syst.*, vol. 5, no. 7, p. 2200459, 2023.

[14] M. F. Falln, M. Antone, N. Roy, and S. Teller, "Drift-free humanoid state estimation fusing kinematic, inertial and lidar sensing," in *Proc. 2014 IEEE Int. Conf. Hum. Robot.*, 2014, pp. 112–119.

[15] M. Neunert, F. Farshidian, A. W. Winkler, and J. Buchli, "Trajectory optimization through contacts and automatic gait discovery for quadrupeds," *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1502–1509, 2017.

[16] G. Fink and C. Semini, "Proprioceptive sensor fusion for quadruped robot state estimation," in *Proc. 2020 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 10914–10920.

[17] M. Camurri *et al.*, "Probabilistic contact estimation and impact detection for state estimation of quadruped robots," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 1023–1030, 2017.

[18] T.-Y. Lin, R. Zhang, J. Yu, and M. Ghaffari, "Legged robot state estimation using invariant kalman filtering and learned contact events," in *Proc. 5th Annu. Conf. Robot Learn.*, Nov 2021, pp. 1057–1066.

[19] S. Piperakis, M. Maravgakis, D. Kanoulas, and P. Trahanias, "Robust contact state estimation in humanoid walking gaits," in *Proc. 2022 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 6732–6738.

[20] M. Maravgakis, D.-E. Argiropoulos, S. Piperakis, and P. Trahanias, "Probabilistic contact state estimation for legged robots using inertial information," in *Proc. 2023 IEEE Int. Conf. Robot. Autom.*, 2023, pp. 12163–12169.

[21] D. He, W. Xu, and F. Zhang, "Symbolic representation and toolkit development of iterated error-state extended kalman filters on manifolds," *IEEE Trans. Ind. Electron.*, vol. 70, no. 12, pp. 12533–12544, 2023.

[22] "Supplementary material for hr^2-kilo", Oct. 2025. [Online]. Available: https://github.com/JixinGao/HR2-KILO/blob/main/Supplementary/HR2_KILO_supplementary.pdf

[23] J. Carpentier *et al.*, "The pinocchio c++ library: A fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," in *Proc. 2019 IEEE/SICE Int. Symp. Syst. Integr.*, 2019, pp. 614–619.