Modular Odometries Fusion with Online Targetless Extrinsic Calibration in a Loosely-Coupled SLAM Framework

Jiaming Wu, Yang Lyu*, Jiakai Gao

Abstract— **To address the challenges of extrinsic calibration and motion fusion in modular odometry systems, this paper proposes a method to integrate both extrinsic calibration and fused localization into a unified system. First, we introduce a non-decoupled optimization approach for Online Targetless Extrinsic Calibration. Instead of decoupling the motion equation for extrinsic calibration, we formulate it as a graph optimization problem, which is solved iteratively using g2o. This method prevents rotational errors from propagating into translation errors by incorporating multiple motion constraints through additional edges in the graph. Next, we propose a Multi-State Loosely-Coupled SLAM Framework. The motion data from multiple sensors are transformed into a unified world coordinate system using the extrinsic parameters obtained from the calibration process. These transformed poses are then incorporated into the back-end optimization, where GTSAM is used to manage the factor graph and the constraints on the pose transformations. This framework effectively fuses the motion information from the modular odometry systems into a cohesive solution. Through experiments, we validate the effectiveness of the Online Targetless Extrinsic Calibration, achieving the required precision.**

Index Terms— **Extrinsic calibration, loosely-coupled SLAM, graph optimization.**

I. INTRODUCTION

With the rapid development of robotics and automation systems, sensor fusion has become crucial in various applications. Among the many sensor fusion approaches, Simultaneous Localization and Mapping (SLAM) technology[1] plays a key role. The performance of SLAM is highly dependent on the accuracy of the extrinsic parameters, which describe the relative poses between sensors. Accurate extrinsic calibration is essential for effectively integrating sensor data[2].

Traditional methods for sensor extrinsic calibration typically rely on manually set calibration targets, such as calibration boards[3] or specific markers. In dynamic and unknown environments, deploying or maintaining calibration targets can be challenging, limiting the applicability of these methods. To address these challenges, Online Targetless Extrinsic Calibration techniques[4] have emerged. These techniques leverage the sensor's own motion information (e.g., odometry) to estimate and update the extrinsic parameters between sensors in real-time, without relying on external calibration targets.

However, in multi-sensor systems, differences in kinematic properties and data synchronization among sensors increase the complexity of online targetless calibration. Coordinating multiple sensors' odometry information to achieve precise calibration is a crucial research direction.

On the other hand, with the rapid development of sensor technology, there are now more "black box" localization systems on the market, integrating multiple sensors and outputting odometry information as a packaged solution.Although these systems greatly simplify the development process, effectively fusing the odometry output from these systems with data from other sensors to improve the robustness and accuracy of the overall localization system remains a pressing technical challenge.

The motivation for this paper is to address these issues by proposing an online, targetless extrinsic calibration method based on the odometry from several sensors, aiming to improve both the real-time performance and accuracy of extrinsic calibration. This is further applied in a loosely-coupled SLAM[5] framework to achieve optimized pose estimation.

Our proposed method requires several sensors being calibrated to be rigidly attached to the same platform and sufficient rotational and translational motion to achieve accurate extrinsic estimation. The contributions of this paper are as follows:

- Motion estimation only: Our extrinsic optimization method only requires motion estimates from each sensor and does not rely on additional information such as external calibration devices or specific environmental features.
- loosely-coupled SLAM Framework: In our loosely-coupled SLAM framework, multi-sensor odometry information is fused and optimized using GTSAM[6], improving the overall system's robustness and localization accuracy.
- Fusion of Modular Localization Systems: Our approach encapsulates targetless online extrinsic calibration and odometry fusion, enabling black-box modular localization systems to quickly initialize and perform localization fusion.

II. RELATED WORKS

Our work primarily relates to extrinsic calibration and loosely-coupled SLAM. Here, we primarily introduce two mainstream techniques related to extrinsic calibration and loosely-coupled SLAM technologies.

*Y. Lyu is with the Key Laboratory of Information Fusion Technology, Ministry of Education, School of Automation, Northwestern Polytechnical University, Xi'an 710072, China, and also with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: lyu.yang@nwpu.edu.cn).

A. Appearance-Based Calibration

Extrinsic calibration based on appearance features is commonly employed for sensors that can identify and extract environmental features, such as radar and cameras. A method involving a lidar and camera that scan a shared chessboard calibration board was demonstrated by Ou [7]. This method includes the construction of a virtual feature point cloud at the board corners and projecting these onto camera pixels to minimize projection errors, thus determining the extrinsic parameters. Meanwhile, Yuan et al. [8] focused on aligning natural edge features between radar and cameras, conducting an analysis on how the precision of these features affects calibration accuracy. They employed voxel segmentation and plane fitting techniques on the LIDAR point cloud to meticulously extract edge features, enhancing calibration precision. In a different approach, Liu et al. [9] developed the gradient-weighted multi-view calibration method (GWM-View), tailored for machining robot positioning based on convergent binocular vision. This innovative method facilitates the robot's precise positioning in multi-view scenarios, demonstrating its effectiveness in complex environments.[10] presented a new self-calibration method to calibrate and compensate for the robot-based inspecting system's kinematic errors by deriving a MDH model without redundancy.

B. Motion-Based Calibration

Motion-Based Calibration primarily relies on sensor motion to solve for extrinsic parameters without the need for specifically prepared features. The foundational equation for most calibrations is the hand-eye calibration equation, represented as AX=XB[11].

[12], [13], and [14] all employ a method that decouples rotation and translation when solving the AX=XB equation, but this approach can result in the propagation of rotation errors to the translation component. Chen [15] was the first to jointly solve the aforementioned equation using screw theory. This method provides a more integrated solution by considering both rotation and translation as interconnected elements, thus potentially reducing error propagation between them. Zhuang [16] employs a nonlinear optimization method that minimizes the Frobenius norms of the rotation matrix. This approach has provided a valuable reference for our work. Taylor et al. [17] enhanced the standard motion-based calibration techniques by estimating the motion observed by each sensor and incorporating estimates of each sensor's reading accuracy. This approach also enables the calibration of GPS with other sensors, offering more flexibility compared to other methods.

C. Loosely-Coupled Slam

Currently, tightly coupled SLAM[18] is the mainstream in SLAM technology, but loosely-coupled systems offer several advantages. Loosely-coupled SLAM can incorporate existing odometry into pose optimization, enhancing the system's robustness. Additionally, it offers benefits such as fast computational speed, high compatibility, and ease of integrating odometry information from different sensors.

Semi-loose-coupled combines the advantages of both coupled methods to a certain extent.SVO[19] utilizes direct sparse image alignment to initially estimate the camera pose and establish feature correspondences. Subsequently, it conducts geometric bundle adjustment(BA)[20] to refine the pose and structural details.[21] proposes a novel semi-direct monocular SLAM method that effectively integrates direct odometry with feature-based SLAM.

Loosely-coupled SLAM is also commonly used as part of backend optimization nowadays. For example, FAST-LIO-SAM builds on FAST-LIO [22] by drawing inspiration from LIO-SAM [23]. In the backend, it integrates keyframes detected by loop closure and their corresponding pose transformations into the factor graph, achieving loop closure optimization within a loosely-coupled framework.

III. Methodology

The above Fig. 1 shows the pipeline of our method, which is mainly divided into three modules. In the initialization phase, we can choose multiple sensors for localization information fusion.

We input the initial motion estimates into our online, targetless extrinsic calibration estimator, collect several sets of motion data, and obtain accurate extrinsic parameters through iterative optimization. The initial motion estimates from the two sensors are transformed into the body coordinate system via the extrinsic parameters, which are then used as the robot's motion state. The robot's initial motion state is determined by the valid odometry received, with the odometry factors derived from valid LiDAR-inertial odometry and visual-inertial odometry.

A. Acquisition of Odometry Information

We first need to obtain the localization information from both sensors to perform robust and accurate extrinsic calibration and fused localization. Selecting an appropriate and precise SLAM algorithm is crucial for validating our subsequent extrinsic calibration and fused localization. Additionally, to meet our requirement for integrating modular odometry, we need a SLAM algorithm that can ultimately output odometry information. This facilitates the integration of modular odometry localization information.

In this paper, we choose to use the FAST-LIO2[24] algorithm to provide localization information for the LiDAR. FAST-LIO2 can directly register raw point clouds to the map and update the map, avoiding the need to extract environmental features. This approach benefits from utilizing subtle features in the environment, thereby improving localization accuracy. Additionally, FAST-LIO2 maintains the map using an incremental k-d tree data structure, and experiments have shown that k-d trees offer the best overall performance among state-of-the-art kNN search[25] data structures in LiDAR odometry.

Since we are using the T265 camera as one of the experimental sensors, the localization information for the T265 is obtained using the proprietary VIO algorithm provided by RealSense T265. Intel has not disclosed the details

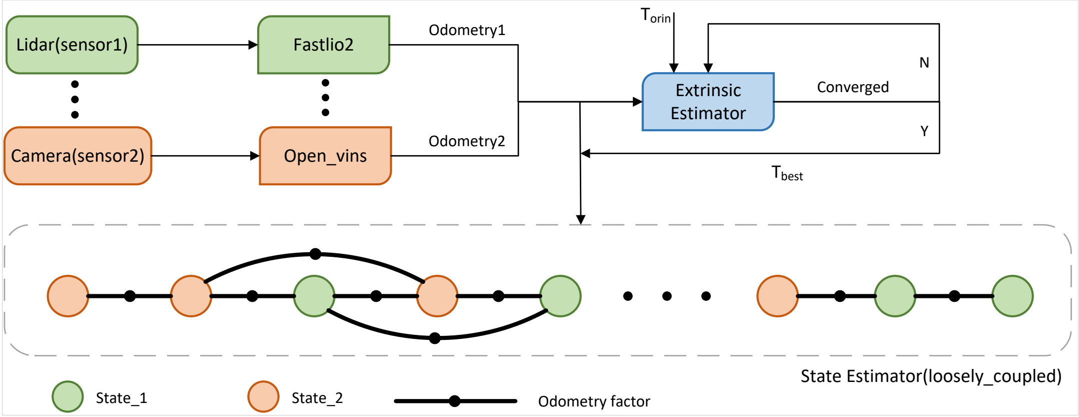

Fig. 1. Framework of the proposed system.

of the algorithm used in the T265. This aligns with our requirements for validating the modular odometry sensors.

B. Online Targetless Extrinsic Calibration Optimizer

Our method is based on the hand-eye calibration problem, typically represented as AX = XB[11], where A and B represent the motion transformation matrices of two sensors over the same time interval, and X is the extrinsic transformation matrix we want to estimate. We have the following notations: the extrinsic parameters from the camera to the LiDAR are denoted as $\mathbf{T}_L^C(\mathbf{R}_L^C, \mathbf{t}_L^C)$, $\mathbf{R}_{Cj}^{Ci}$, and $\mathbf{t}_{Cj}^{Ci}$, which represent the rotation and translation from time i to time j made by the camera, respectively. Similarly, $\mathbf{R}_{Lj}^{Li}$ and $\mathbf{t}_{Lj}^{Li}$ represent the rotation and translation made by the LiDAR.

We can expand the above AX=XB as follows:

$$\mathbf{R}_L^C \cdot \mathbf{t}_{Lj}^{Li} = \mathbf{R}_{Cj}^{Ci} \cdot \mathbf{R}_L^C \tag{1}$$

$$\mathbf{R}_L^C \cdot \mathbf{t}_{Lj}^{Li} + \mathbf{t}_L^C = \mathbf{R}_{Cj}^{Ci} \cdot \mathbf{t}_L^C + \mathbf{t}_{Cj}^{Ci} \tag{2}$$

We derive the odometry obtained from SLAM as the pose transformation from time 0 to time n, i.e., T_n^0, and since $\mathbf{T}_i^0 \cdot \mathbf{T}_j^i = \mathbf{T}_j^0$, we have:

$$\begin{bmatrix} \mathbf{R}_i^0 & \mathbf{t}_i^0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{R}_j^i & \mathbf{t}_j^i \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_j^0 & \mathbf{t}_j^0 \\ 0 & 1 \end{bmatrix} \tag{3}$$

It can be obtained that:

$$\mathbf{R}_j^i = (\mathbf{R}_i^0)^T \cdot \mathbf{R}_j^0 \tag{4}$$

$$\mathbf{t}_j^i = (\mathbf{R}_i^0)^T \cdot (\mathbf{t}_j^0 - \mathbf{t}_i^0) \tag{5}$$

Both $\mathbf{R}_{Cj}^{Ci}$, $\mathbf{R}_{Lj}^{Li}$, $\mathbf{t}_{Cj}^{Ci}$, $\mathbf{t}_{Lj}^{Li}$ are calculated according to the above rules.

We use a graph-based optimization method for extrinsic calibration. We define the residuals for solving extrinsic parameters as follows. $\mathbf{r}_1$ is the rotation residual, $\mathbf{r}_2$ is the translational residual. we use ”Exp” and ”Log” to respectively denote the exponential map from Lie algebra to Lie group[26] and the logarithmic map from Lie group to Lie algebra vector. $\mathrm{Exp}: R^3 \rightarrow \mathrm{SO}(3)(\phi \mapsto \exp(\phi^\wedge))$; $\mathrm{Log}: \mathrm{SO}(3) \rightarrow R^3(\mathbf{R} \mapsto \log(\mathbf{R})^\vee)$:

$$\mathbf{r}_1 = \mathrm{Log}\left[\left(\mathbf{R}_L^C \cdot \mathbf{t}_{Lj}^{Li}\right)^T \cdot \left(\mathbf{R}_{Cj}^{Ci} \cdot \mathbf{R}_L^C\right)\right] \tag{6}$$

$$\mathbf{r}_2 = \left(\mathbf{R}_L^C \cdot \mathbf{t}_{Lj}^{Li} + \mathbf{t}_L^C\right) - \left(\mathbf{R}_{Cj}^{Ci} \cdot \mathbf{t}_L^C + \mathbf{t}_{Cj}^{Ci}\right) \tag{7}$$

We solve the following least squares problem to obtain the extrinsic parameters

$$\mathbf{R}_L^C, \mathbf{t}_L^C = \arg \min_{\mathbf{R}_L^C, \mathbf{t}_L^C} \sum_{i=0}^{n} \left(\left\|\mathbf{r}_1\right\|^2 + \left\|\mathbf{r}_2\right\|^2 \right) \tag{8}$$

Where n is the number of data sets collected. We need to collect multiple sets of SLAM data as synchronized data for optimization constraints to perform Bundle Adjustment to improve the robustness and accuracy of the extrinsic parameter estimation.

The Jacobian matrices of the residuals with respect to the state variable and are given as follows:

$$\begin{aligned} \frac{\partial \mathbf{r}_1}{\partial \delta \phi_L^C} =& \mathbf{J}_r^{-1}\left(\log\left[\left(\mathbf{R}_L^C \cdot \mathbf{R}_{Lj}^{Li}\right)^T \cdot \left(\mathbf{R}_{Cj}^{Ci} \cdot \mathbf{R}_L^C\right)\right]\right) \\ & \cdot \left(I - \left(\mathbf{R}_L^C\right)^T \cdot \left(\mathbf{R}_{Cj}^{Ci}\right)^T \cdot \mathbf{R}_L^C\right) \end{aligned} \tag{9}$$

$$\frac{\partial \mathbf{r}_1}{\partial \delta t_L^C} = 0_{3\times 3} \tag{10}$$

$$\frac{\partial \mathbf{r}_2}{\partial \delta \phi_L^C} = -\mathbf{R}_L^C \cdot \left(\mathbf{t}_{Lj}^{Li}\right)^\wedge \tag{11}$$

$$\begin{aligned} \frac{\partial \mathbf{r}_2}{\partial \delta t_L^C} &= \lim_{\delta t_L^C \rightarrow 0} \frac{\mathbf{r}_1\left(\mathbf{t}_L^C + \mathbf{R}_L^C \cdot \delta t_L^C\right) - \mathbf{r}_1\left(\mathbf{t}_L^C\right)}{\delta t_L^C} \\ &= \left(I - \mathbf{R}_{Cj}^{Ci}\right) \cdot \mathbf{R}_L^C \end{aligned} \tag{12}$$

We use the g2o solver[27] to solve the above optimization problem. We combine the Jacobians of the two residuals with respect to the rotation and translation state increments into the combined Jacobian matrix as shown in Fig. 2, which is a 6 × 6 combined Jacobian matrix. We choose the

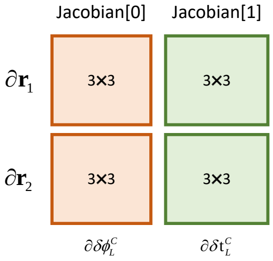

Fig. 2. Schematic of the Jacobian matrix of the residuals with respect to the state increments

Levenberg-Marquardt(LM) optimization method. To reduce the influence of outliers, we set the Huber robust kernel function as:

$$H(\mathbf{r}_1, \mathbf{r}_2) = \begin{cases} \frac{1}{2}(\|\mathbf{r}_1\|^2 + \|\mathbf{r}_2\|^2) & (\sqrt{\|\mathbf{r}_1\|^2 + \|\mathbf{r}_2\|^2} < 0.5) \\ 0.5 \cdot \sqrt{\|\mathbf{r}_1\|^2 + \|\mathbf{r}_2\|^2 - 0.25} & \text{else} \end{cases} \tag{13}$$

In the above optimization, to ensure the convergence of the optimization results, we set the convergence condition as the translation adjustment in each iteration being less than 0.01. For the rotation part of the extrinsics, we use the rotation angle to measure the difference between the current and previous iterations. n denotes the most recent iteration number:

$$\Delta\mathbf{R} = \mathbf{R}_L^C(n-1)^T \cdot \mathbf{R}_L^C(n) \tag{14}$$

$$\theta = \arccos\left(\frac{\text{trace}(\Delta\mathbf{R}) - 1}{2}\right) \tag{15}$$

When the rotation angle difference θ is less than $0.05°$ and the translation vector is less than 0.01, we consider the result to have converged and obtain the optimized result. Otherwise, the optimized result is used as the initial value for the next iteration, and the parameters are updated by optimizing again.

By iteratively solving the above least squares problem, we can obtain the optimized parameters , which are used for further optimization of the odometry information from the subsequent sensors.

C. Multi-State Loosely-Coupled SLAM Framework

In the previous section, we obtained the transformation relationship between the camera coordinate system and the LiDAR coordinate system. Next, we consider how to combine the modular odometry information of other sensors with the extrinsic parameters obtained above. Thus, we propose a multi-state loosely-coupled SLAM framework, where this framework can effectively fuse odometry information to prevent the occurrence of multiple conflicting fusion scenarios.

Our loosely-coupled framework is depicted in Fig. 1. In our work, since the initial coordinate system of the LiDAR is used as the world coordinate system, the LiDAR odometry can be directly incorporated into the backend optimization states. The camera odometry, after being transformed by the optimized external parameters obtained as described, can be converted into the world coordinate system. Temporally, in the backend, odometry from any sensor can be added to the backend states as long as it is deemed valid. If the previous state in the backend is $\mathbf{x}_n$ and the newly added state is $\mathbf{x}_{n+1}$, then odometry from other sensors serves as the odometry factor between $\mathbf{x}_n$ and $\mathbf{x}_{n+1}$.When the LiDAR odometry is the most recent state, the constraint between states is the motion transformation between this state and the previous state. Additionally, if $\mathbf{x}_n$ and $\mathbf{x}_{n+1}$ originate from different sensors, then the pose transformation of $\mathbf{x}_{n+1}$ from the last position of the same sensor serves as an additional constraint, and the covariance of this constraint is set lower than that of inter-sensor constraints, because the pose transformations from the same sensor are not interpolated and theoretically have higher reliability. We denote the pose of the optimized state at $\mathbf{x}_n$ as $\overline{\mathbf{T}}_n$, and if ${}^{L0}\mathbf{T}_{L,(n+1)}^{L,0}$ represents the LiDAR pose at the moment $n+1$ in the initial LiDAR coordinate system. The odometry factor constraint between $\mathbf{x}_n$ and $\mathbf{x}_{n+1}$ would be

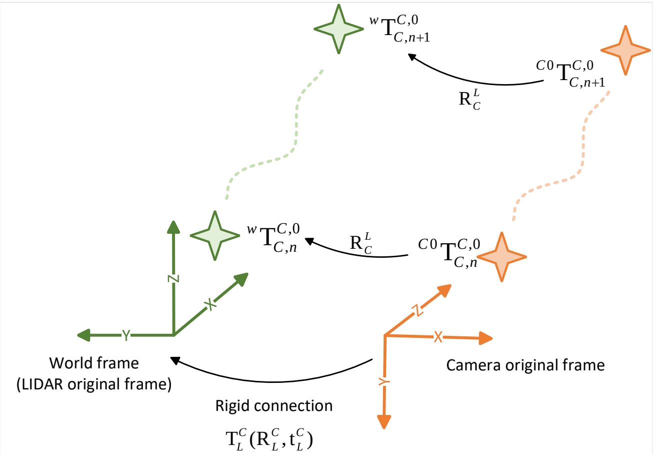
${}^{w}\mathrm{T}_{C,n+1}^{C,0}$
${}^{C0}\mathrm{T}_{C,n+1}^{C,0}$
R_C^L
${}^{w}\mathrm{T}_{C,n}^{C,0}$
R_C^L
${}^{C0}\mathrm{T}_{C,n}^{C,0}$
Z
X
Y
Z
X
Y
World frame
(LIDAR original frame)
Camera original frame
Rigid connection
$\mathrm{T}_L^C(\mathrm{R}_L^C, \mathrm{t}_L^C)$

Fig. 3. The diagram illustrating the transformation of the camera odometry to the world coordinate system

$$\Delta\mathbf{T}_{n+1}^{n} = \overline{\mathbf{T}}_n{}^{T} \cdot {}^{L0}\mathbf{T}_{L,(n+1)}^{L,0} \tag{16}$$

If ${}^{C0}\mathbf{T}_{C,(n+1)}^{C,0}$ is the camera pose at the moment $n+1$, before being added to the factor graph, the camera's motion needs to be transformed from the original camera coordinate system to the world coordinate system,as shown in Fig. 3. Then, the odometry factor constraint between $\mathbf{x}_n$ and $\mathbf{x}_{n+1}$ would be

$$\Delta\mathbf{T}_{n+1}^{n} = \overline{\mathbf{T}}_n{}^{T} \cdot (\mathbf{R}_C^L) \cdot {}^{C0}\mathbf{T}_{C,(n+1)}^{C,0} \tag{17}$$

IV. EXPERIMENTAL RESULTS

We designed an experiment to evaluate the accuracy and error of targetless online extrinsic calibration as well as the effects of integrating the calibrated extrinsics into our proposed loosely coupled framework. We used the VIO and LIO provided by the R3LIVE[28] framework to test

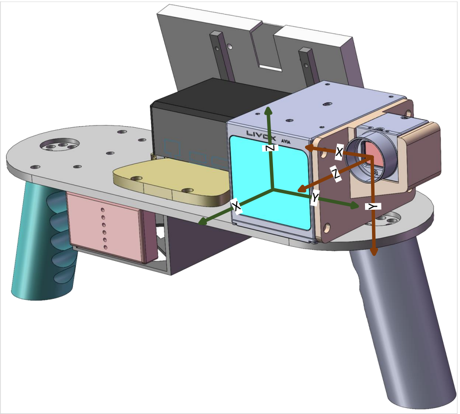

Fig. 4. Diagram of LIDAR and camera coordinate positions in the R3LIVE experiment

our algorithm, as its relatively accurate VIO and LIO can eliminate most of the errors beyond the intrinsic system errors in extrinsic calibration.

A. Metrics

We use the Umeyama algorithm[29] to align the LIO trajectory with the VIO trajectory transformed to the world coordinate system by obtained extrinsics to evaluate the extrinsic errors. For the evaluation of the loosely-coupled SLAM, we use the Relative Pose Error (RPE) and Absolute Pose Error (APE) to assess the effectiveness of our method. These metrics are calculated using the EVO[30] toolbox.

B. Evaluation With R3LIVE Data

Data was collected using a LiVOX AVIA LIDAR and a FLIR Blackfly BFS-u3-13y3c global shutter camera. Fig. 4 illustrates the positional relationship between the camera and the lidar. We utilized the previously collected hku-main-building rosbag, which contains indoor and outdoor scenes with significant features. The 3D reconstructed point cloud provides a visual representation of the general scanning environment. The computation results for $\mathbf{T}_L^C$ are as follows:

$$\begin{gathered}[0.00192365, -0.99999, -0.00409489, 0.0257051, \\ -0.0135795, 0.00406839, -0.9999, 0.0243618, \\ 0.999906, 0.00197906, -0.013715, 0.00864774, \\ 0.0, 0.0, 0.0, 1.0]\end{gathered}$$

The error in converting the rotation matrix to the absolute rotation angle is $0.3485°$. Both the rotational and translational errors are well within reasonable bounds, with the rotational error being less than 0.5 degrees and the translational error at the centimeter level.

TABLE I

APE,RPE evaluation results [m] (VIO,RIO to World Frame)

	max.	mean	median	min.	RMSE	SSE	std.
APE	0.537	0.272	0.279	0.006	0.304	217.906	0.136
RPE	9.552	0.482	0.201	0.002	1.176	247.788	1.073

To assess the accuracy of our loosely coupled framework, we plotted the trajectories and the positions of VIO, LIO, and the loosely coupled odometry results in Fig. 5 and Fig. 6. Figure X shows the trajectories for VIO, LIO, and the loosely coupled odometry, respectively. Tab. I and Tab. II show the Absolute Pose Error (APE) and Relative Pose Error (RPE) results for the VIO and LIO, transformed using external parameters, compared to the loosely coupled odometry and LIO results.

TABLE II

APE,RPE evaluation results [m] (VIO,Loosely-Coupled Odometry)

	max.	mean	median	min.	RMSE	SSE	std.
APE	0.804	0.145	0.124	0.023	0.174	325.977	0.095
RPE	0.492	0.004	0.0007	0.0	0.039	16.549	0.039

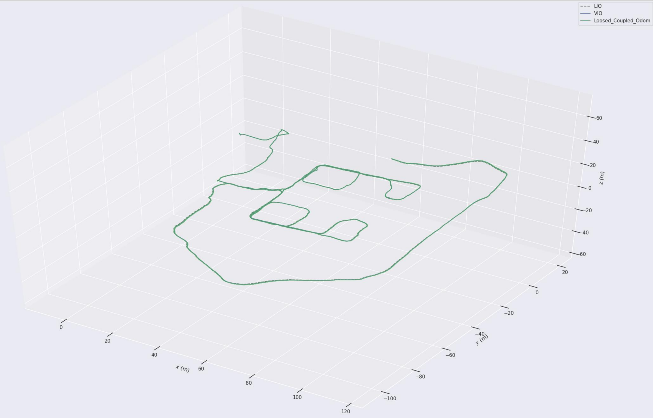

Fig. 5. Experimental trajectory diagram

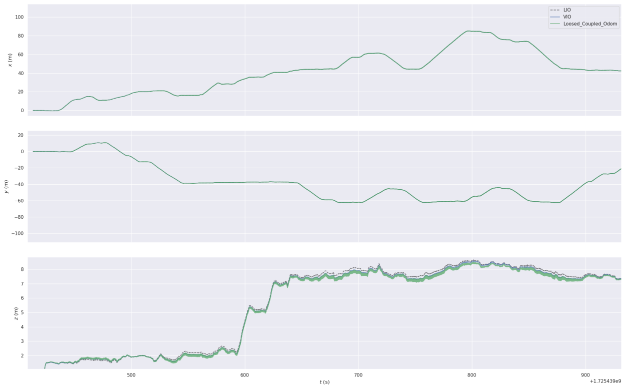

Fig. 6. Experimental positioning results

Since both VIO (Visual-Inertial Odometry) and LIO (LiDAR-Inertial Odometry) are highly accurate, the broad range of motion in the dataset being tested results in overlapping effects in the trajectory plot. Overall, the experiments have validated the effectiveness of our entire modular odometry extrinsic calibration and the SLAM framework with loose coupling.

V. CONCLUSIONS

In the context of increasingly modular odometry systems, the fusion of different odometries can complement the advantages of each system in varying environments.

First, we jointly solve the motion equation AX=XB and solve it using nonlinear optimization. By modeling the problem as a graph optimization task in the g2o framework, we ensure that the rotation and translation equations constrain each other. The rotational residuals are mapped onto the Lie algebra $so(3)$ vectors, avoiding redundancy in the optimization variables. This Online Targetless Extrinsic Calibration Optimizer achieves high precision, meeting the requirements for subsequent loosely-coupled motion fusion. Next, we propose a Multi-State Loosely-Coupled SLAM Framework, which fuses motion information from various types of sensors. The back-end uses GTSAM to manage optimization factors, ensuring smooth factor graphs globally.

Looking forward, there is still room for improvement in our loosely-coupled framework. We plan to integrate environmental feature detection to diagnose and remove factors with significant errors or failures. By discarding unreliable factors from the optimization process, we aim to enhance the robustness and accuracy of odometry fusion across diverse environments, ultimately making the loosely-coupled SLAM system more reliable and precise.

References

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[2] F. Basso, E. Menegatti, and A. Pretto, "Robust intrinsic and extrinsic calibration of rgb-d cameras," *IEEE Transactions on Robotics*, vol. 34, no. 5, pp. 1315–1332, 2018.

[3] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[4] J. Lv, J. Xu, K. Hu, Y. Liu, and X. Zuo, "Targetless calibration of lidar-imu system based on continuous-time batch estimation," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9968–9975.

[5] T. Li, L. Pei, Y. Xiang, Q. Wu, S. Xia, L. Tao, X. Guan, and W. Yu, "P 3-loam: Ppp/lidar loosely coupled slam with accurate covariance estimation and robust raim in urban canyon environment," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 6660–6671, 2020.

[6] F. Dellaert, "Factor graphs and gtsam: A hands-on introduction," *Georgia Institute of Technology, Tech. Rep*, vol. 2, p. 4, 2012.

[7] J. Ou, P. Huang, J. Zhou, Y. Zhao, and L. Lin, "Automatic extrinsic calibration of 3d lidar and multi-cameras based on graph optimization," *Sensors*, vol. 22, no. 6, p. 2221, 2022.

[8] C. Yuan, X. Liu, X. Hong, and F. Zhang, "Pixel-level extrinsic self calibration of high resolution lidar and camera in targetless environments," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7517–7524, 2021.

[9] H. Liu, J. Fu, M. He, L. Hua, and D. Zhu, "Gwm-view: gradient-weighted multi-view calibration method for machining robot positioning," *Robotics and Computer-Integrated Manufacturing*, vol. 83, p. 102560, 2023.

[10] C. Yu and J. Xi, "Simultaneous and on-line calibration of a robot-based inspecting system," *Robotics and Computer-Integrated Manufacturing*, vol. 49, pp. 349–360, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736584516301806

[11] K. H. Strobl and G. Hirzinger, "Optimal hand-eye calibration," in *2006 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2006, pp. 4647–4653.

[12] Y. Shiu and S. Ahmad, "Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form ax=xb," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 1, pp. 16–29, 1989.

[13] R. Y. Tsai, R. K. Lenz *et al.*, "A new technique for fully autonomous and efficient 3 d robotics hand/eye calibration," *IEEE Transactions on robotics and automation*, vol. 5, no. 3, pp. 345–358, 1989.

[14] J. C. K. Chou and M. S. Kamel, "Finding the position and orientation of a sensor on a robot manipulator using quaternions," *The International Journal of Robotics Research*, vol. 10, pp. 240 – 254, 1991. [Online]. Available: https://api.semanticscholar.org/CorpusID:33764587

[15] H. Chen, "A screw motion approach to uniqueness analysis of head-eye geometry," in *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1991, pp. 145–151.

[16] H. Zuang and Y. C. Shiu, "A noise-tolerant algorithm for robotic hand-eye calibration with or without sensor orientation measurement," *IEEE transactions on systems, man, and cybernetics*, vol. 23, no. 4, pp. 1168–1175, 1993.

[17] Z. Taylor and J. Nieto, "Motion-based calibration of multimodal sensor extrinsics and timing offset estimation," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1215–1229, 2016.

[18] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, p. 1004–1020, Aug. 2018. [Online]. Available: http://dx.doi.org/10.1109/TRO.2018.2853729

[19] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 15–22.

[20] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment—a modern synthesis," in *Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece, September 21–22, 1999 Proceedings*. Springer, 2000, pp. 298–372.

[21] S. H. Lee and J. Civera, "Loosely-coupled semi-direct monocular slam," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 399–406, 2019.

[22] W. Xu and F. Zhang, "Fast-lio: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter," 2021. [Online]. Available: https://arxiv.org/abs/2010.08196

[23] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," 2020. [Online]. Available: https://arxiv.org/abs/2007.00258

[24] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," 2021. [Online]. Available: https://arxiv.org/abs/2107.06829

[25] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, "Efficient knn classification with different numbers of nearest neighbors," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 5, pp. 1774–1785, 2017.

[26] C. Chevalley, *Theory of Lie groups*. Courier Dover Publications, 2018.

[27] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g 2 o: A general framework for graph optimization," in *2011 IEEE international conference on robotics and automation*. IEEE, 2011, pp. 3607–3613.

[28] J. Lin and F. Zhang, "R3live: A robust, real-time, rgb-colored, lidar-inertial-visual tightly-coupled state estimation and mapping package," 2021. [Online]. Available: https://arxiv.org/abs/2109.07982

[29] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 13, no. 04, pp. 376–380, 1991.

[30] M. Grupp, "evo: Python package for the evaluation of odometry and slam." https://github.com/MichaelGrupp/evo, 2017.