Autonomous–Targetless Extrinsic Calibration of Thermal, RGB, and LiDAR Sensors

Wenyu Yang, Haojun Luo, Kwai-Wa Tse, Haochen Hu, Kang Liu, Boyang Li, and Chih-Yung Wen

***Abstract*— Mobile robots extensively employ multiple sensors, including RGBD cameras, LiDAR, and thermal sensors. Sensor fusion plays a vital role in localization and environment perception tasks. However, traditional manual target-based methods for achieving consistent alignment among multiple sensors, especially thermal cameras, are laborious and lack adaptability. This work introduces an autonomous–targetless framework for calibrating LiDAR–RGB and LiDAR–thermal sensors in mobile robots. In our proposed framework, we examine the characteristics of thermal images, identify suitable calibration scenarios, and employ the thermal bridge and line-based PnL algorithm to enable autonomous and targetless calibration. Experimental results demonstrate the efficiency of our method, with overall translation errors of 2.77 and 3.86 cm, and overall rotation errors of 0.21° and 0.46°, respectively, in LiDAR–RGB and LiDAR–thermal calibrations. These results are comparable to state-of-the-art techniques and traditional target-based manual methods. The analysis of different thermal scenes highlights the importance of well-aligned and distinguishable edges across thermal–RGB–LiDAR modalities for optimal calibration results. Simulation tests using synthetic data and validation tests using real-world data showcase the robustness of our model in executing targetless extrinsic calibrations.**

***Index Terms*— Autonomous–targetless calibration, LiDAR–RGB sensor calibration, LiDAR–thermal sensor calibration, multimodal extrinsic calibration, PnL.**

I. Introduction

IN COMPUTER vision and robotics applications, sensor fusion enhances various functionalities such as robot perception, localization, SLAM, and decision-making. The combination of RGBD cameras, LiDAR, and thermal cameras has demonstrated synergistic benefits that surpass the output of individual sensors. By merging information from these diverse sensors, a robot's perception system can acquire data encompassing visible light, ambient temperature, and depth. Fig. 1 illustrates an example use case where an aerial robot equipped with LiDAR, RGBD camera, and thermal camera efficiently performs multimodal inspection tasks. However, sensor fusion also presents challenges, with precise alignment of sensor data becoming crucial. External calibration is a key step to establish associations between sensors and ensure accurate alignment for effective fusion.

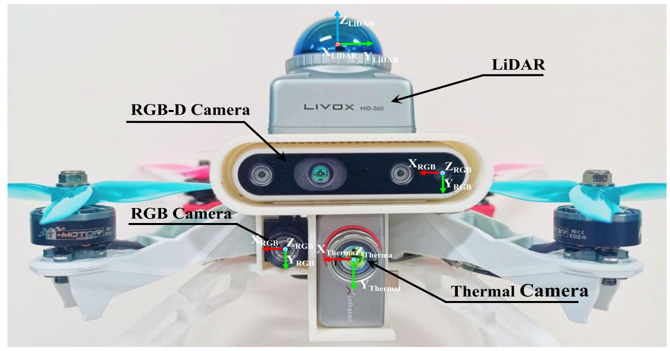


Fig. 1. Sensor rig assembled on a UAV.

Extrinsic calibration entails establishing correspondences between data obtained from two different sensors, encompassing combinations such as LiDAR–RGB, RGB–thermal, LiDAR–thermal, and others. Our primary focus centers on multimodal extrinsic calibration, specifically LiDAR–RGB and LiDAR–thermal calibrations. Among these, LiDAR–RGB calibration has been extensively studied [1], [2], followed by RGB–thermal calibration [3], while LiDAR–thermal calibration remains relatively underexplored [4].

Various calibration methods can be broadly categorized as target-based or targetless and manual or autonomous. Early approaches predominantly employed target-based and manual techniques. Autonomous and targetless calibration is most desired, although this poses the greatest challenge. Currently, autonomous and targetless methods are prevalent for RGB-LiDAR calibration. While LiDAR–thermal calibration methods are virtually nonexistent. Many existing extrinsic calibration methods that involve a thermal camera still rely on target-based approaches. This lack of research and development in LiDAR–thermal calibration hampers the full utilization of thermal cameras' advantages, particularly in the context of service robots. The absence of robust calibration techniques impedes the advancement of multisensor fusion and downstream algorithms, preventing the realization of their full potential.

Extrinsic calibration encounters its first major challenge in achieving cross-modal information correspondence, which

Received 7 April 2024; revised 29 August 2024; accepted 18 September 2024. Date of publication 14 October 2024; date of current version 31 October 2024. This work was supported by the Research Center for Unmanned Autonomous Systems, Hong Kong Polytechnic University, Hong Kong, SAR, China, under Project CE55 and Project CE0F. The Associate Editor coordinating the review process was Dr. Md. Moinul Hossain. *(Corresponding authors: Boyang Li; Chih-Yung Wen.)*

Wenyu Yang, Kwai-Wa Tse, Haochen Hu, Kang Liu, and Chih-Yung Wen are with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong, SAR, China (e-mail: wenyu.yang@connect.polyu.edu.hk; kwai-wa.tse@connect.polyu.hk; haru-haochen.hu@connect.polyu.hk; kang1.liu@polyu.edu.hk; cywen@polyu.edu.hk).

Haojun Luo is with the Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong, SAR, China (e-mail: hluoaw@connect.ust.hk).

Boyang Li is with the School of Engineering, The University of Newcastle, Callaghan, NSW 2308, Australia (e-mail: boyang.li@newcastle.edu.au).

Digital Object Identifier 10.1109/TIM.2024.3480237

proves difficult even when performed manually. Each sensor captures data from different domains: thermal cameras capture infrared (IR) waves, RGB cameras capture visible light, and LiDAR relies on IR but with a different wavelength, each representing unique and complementary information. RGB cameras are adept at capturing detailed texture information, whereas LiDAR provides precise distance measurements despite having sparser data points. On the other hand, thermal cameras excel in perceiving temperature, making them particularly suitable for tasks like pedestrian detection and avoidance in service robots and autonomous driving. The complementary nature of these sensors enhances their effectiveness when used together.

However, their complementarity presents challenges in achieving extrinsic calibration, as establishing relevant correspondences between sensors to solve for extrinsic parameters becomes increasingly difficult. Traditional calibration methods involve creating calibration patterns capable of generating temperature information and possessing associated features like texture and specific geometric shapes. These common features are then extracted manually or automatically. However, the complexity and bulkiness of such calibration patterns make their manufacturing arduous. Moreover, if the extrinsic parameters of the sensors on the robot change due to vibrations or collisions, the original calibration becomes invalid, and further calibration and intervention will be required or false alignment will be caused.

The second challenge arises from the intricate nature of the thermal imaging process and the dearth of robotic algorithms designed to address the autonomous–targetless calibration problem. The complexity inherent in thermal imaging presents significant challenges for researchers, resulting in a relative scarcity of research focused on thermal camera calibration compared to other sensor types. As a result, most existing calibration methods for thermal cameras tend to rely on target-based or manual approaches. The development of automatic calibration algorithms specifically tailored for thermal cameras is limited, and even fewer options exist for autonomous–targetless calibration methods. This lack of automated calibration techniques for thermal cameras impedes the full utilization of their capabilities within multisensor fusion systems, hindering a comprehensive understanding of their potential benefits across various applications.

We tackle the above two problems by investigating the thermal imaging principle and a cross-modal common feature extraction method is developed. A LiDAR–RGB and LiDAR–thermal calibration framework is proposed. The main contributions of this work are as follows.

1) We propose an autonomous and targetless calibration framework for LiDAR–RGB and LiDAR–thermal sensors calibration, which is the first autonomous–targetless calibration method that can directly obtain the extrinsic parameters between the thermal camera and LiDAR sensors based on our knowledge.
2) Based on the analysis of thermal edge properties, we propose thermal edge detection and edge-matching algorithms and evaluated them in various scenes in terms of convergence precision and accuracy.
3) The source codes is released[1] to benefit the community.

The structure of this article is organized as follows. We review the related works of multisensor fusion and autonomous–targetless calibration in Section II. Section III provides some preliminaries. Section IV details the extrinsic calibration methodology. Simulation and real-world experiment results are presented in Section V. Finally, we conclude in Section VI.

II. Related Work

Extrinsic calibration can be categorized based on various factors or characteristics. Two factors, in particular, significantly influence the applicability and cost considerations [5]: 1) autonomous versus manual calibration and 2) calibration with or without a target [6]. Initially, the manual target-based approach was the most basic and commonly employed method. However, the autonomous calibration method without the need for target pairs is highly sought after due to its cost-effectiveness and broader scene coverage, making it the preferred choice. In this section, we conduct a literature review on sensor extrinsic calibration from these two dimensions. In Section II-A, we first review the calibration of RGB–RGB and the calibration of LiDAR–RGB sensor calibration. In Section II-B, we review RGB–thermal and LiDAR–thermal sensor calibration.

A. RGB–RGB and LiDAR–RGB Sensor Calibration

As the most widely used and most similar sensor to human vision, the RGB camera is intensively studied and best understood. For camera calibration, the most widely applied method is the one developed by Zhang [1]. This method uses a checkerboard with a known size as a target and manually extracts features to obtain responses between real-world and image points. Subsequent research has largely focused on improving and extending Zhang's method. Some researchers have provided more user-friendly GUIs [7] to label the key points manually, while others have tried to develop better algorithms for automatic checkerboard corner detection [8].

With the rapid development of autonomous driving in the past decade, the price of LiDAR sensors has also dropped significantly [9]. As a 3-D sensor, LiDAR is naturally complementary to cameras, so there has been a lot of research on LiDAR–RGB calibration [10]. The development of LiDAR–RGB is also similar to camera calibration. In the early days, target-based and manual methods were predominant [11]. Then, some researchers developed autonomous feature detection [2] feature matching [12]. Most recently, autonomous–targetless extrinsic calibration methods have drawn the attention of the research community [13], [14].

The main breakthroughs and innovations in calibration methods are in two areas:

1) toward autonomous feature detection and
2) toward generic scenes.

In the first domain, with the advancement of computer vision technology, automatic feature detection algorithms have continuously evolved, thereby simplifying the calibration process.

[1] https://github.com/allenthreee/thermal_range_calib

Simultaneously, these automatic detection algorithms have also enhanced the precision when utilizing fixed calibration patterns [15]. In the second domain, feature detection methods have expanded beyond the confines of checkerboards. Automatic detection approaches for objects such as boxes [12], spheres [16], and other entities have been developed. Furthermore, methods for detecting people, skylines, or edge discontinuities have also emerged [17]. Overall, the progress in this field is geared toward more versatile features and increasingly automated and accurate feature detection. However, these two factors somewhat impose constraints on each other since more general features may result in reduced accuracy. To enhance accuracy and automation, the utilization of more specific calibration targets becomes necessary.

In summary, the core process of these methods is:

1) feature detection;
2) feature matching; and
3) extrinsic solving.

Although there are also other schools of thought, such as information theory-based methods [18], the most widely used, mature, and reliable technique, offering the highest accuracy, remains the aforementioned three-step process. From Zhang's [1] pioneering method to the present day, the variations among different techniques lie primarily in whether features are manually or automatically extracted and whether these features are obtained from artificially created targets or extracted from the surrounding environment. As the pursuit of greater autonomy and universality in extrinsic calibration methods intensifies, so does the challenge. Consequently, we speculate that the main direction of development lies in autonomous–targetless calibration methods.

B. RGB(D)–Thermal and LiDAR–Thermal Sensor Calibration

There exists a scarcity of studies focused on the extrinsic calibration between thermal cameras and other sensors; however, the calibration process bears similarities to that of RGB cameras [19]. The use of calibration checkerboards is common in earlier researchers [20]. Nonetheless, the creation of a checkerboard for thermal cameras poses challenges due to the requirement of introducing a temperature difference onto the checkerboard, which subsequently induces heat transfer, while the colored pattern remains fixed [21]. Numerous endeavors have been undertaken to devise reliable calibration patterns [3]. Nevertheless, the manufacturing of such patterns proves exceedingly laborious, thereby impeding the widespread adoption of thermal cameras and their fusion with other types of sensors.

Following the standard pipeline of 1) feature extraction, 2) feature matching, and 3) extrinsic solving, one of the main difficulties is to align thermal and visible or geometric features by finding features in the environment. Lussier and Thrun [22] used the human body as a target because the temperature of the human body is usually significantly different from the ambient temperature. Recently, Fu et al. [23] used buildings to extract edges and then used these edges for calibration.

TABLE I
COMPARISON OF DIFFERENT MULTIMODAL EXTRINSIC CALIBRATION METHODS

Method	LiDAR-RGB	LiDAR-thermal	auto	targetless
Beltran's [2]	✓	×	✓	×
Yuan's [13]	✓	×	✓	✓
Fu's [23]	✓	×	✓	✓
Zhang's [25]	✓	✓	✓	×
Ours	✓	✓	✓	✓

While several attempts have been made to calibrate thermal cameras and LiDAR sensors directly, no successful outcomes have been reported thus far. Fu et al. [23] have made notable progress toward this objective, although the presence of relatively large errors in their method can be attributed to the inherent blurriness of thermal images when compared to colored images [24] and LiDAR point clouds are sparse. These factors present challenges for optimization algorithms, hindering their convergence and the attainment of accurate results. In our research, we thoroughly analyze these factors and propose solutions to mitigate their impact, potentially establishing the first autonomous and targetless calibration method capable of directly calibrating thermal cameras, RGB cameras, and LiDAR sensors. Table I provides an overview of the sensor modalities and algorithmic limitations associated with various approaches for multimodal extrinsic parameter calibration.

III. PRELIMINARIES

In this article, we minimize the reprojection error of edge features to solve for the extrinsic parameters (i.e., relative pose including rotation and translation) between different sensors. The translation vector can be represented by a 3-D vector, while there are multiple choices for representing rotation, such as Euler angles, quaternions, rotation matrices, and rotation vectors. To facilitate optimization without introducing new constraints (quadratic constraint of unit quaternions), we use rotation matrices (Lie Group on SO(3)) and rotation vectors (Lie Algebra of the Lie Group SO(3)) to represent rotation and optimize using rotation vectors that are living on smooth manifolds. We minimize the cost function using nonlinear optimization on a smooth manifold. To clearly state the problem, we provide a brief review of some useful concepts.

A. Basic Notations

We use $\boldsymbol{R} \in \mathbb{R}^{3\times 3}$ to denote a 3-D rotation matrix, $\boldsymbol{t} \in \mathbb{R}^3$ to denote the 3-D translation vector and

$$\boldsymbol{T} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0} & 1 \end{bmatrix} \in \mathbb{R}^{4\times 4}$$

to denote 3-D rigid body transformation. Considering two transformation matrices

$$\boldsymbol{T}_1 = \begin{bmatrix} \boldsymbol{R}_1 & \boldsymbol{t}_1 \\ \boldsymbol{0} & 1 \end{bmatrix}$$

and

$$\boldsymbol{T}_2 = \begin{bmatrix} \boldsymbol{R}_2 & \boldsymbol{t}_2 \\ \boldsymbol{0} & 1 \end{bmatrix} \in \mathrm{SE}(3)$$

where $SE(3)$ represents the Special Euclidean Group in three dimensions, we define the transformation as

$$T_1 \cdot T_2 = \begin{bmatrix} \boldsymbol{R_1 R_2} & \boldsymbol{t_1} + \boldsymbol{R_1 t_2} \\ \boldsymbol{0} & 1 \end{bmatrix}.$$

We use T_{21} to denote the transformation matrix from coordinate 1 to coordinate 2. For a specific scenario, we aim to add two vectors r_v^{pq} and r_w^{qi}. Here, the vector r_w^{qi} points from point q to point i located in the world coordinate system, while r_v^{pq} points from point p to point q in the vehicle coordinate system. To achieve this, we first transform the vector r_v^{pq} to the world coordinate system using T_{vw}. Subsequently, we add this transformed vector to r_w^{qi}. The complete transformation operation can be described as follows:

$$r_w^{pi} = T_{wv} r_v^{pq} + r_w^{qi}. \tag{1}$$

B. Lie Group and Lie Algebra

The Special Orthogonal Group (the Rotation Group) describes the group of rotation matrices and it is formally defined as $\mathrm{SO}(3) = \{\boldsymbol{R} \in \mathbb{R}^{3\times3} | \boldsymbol{R}\boldsymbol{R}^\top = \boldsymbol{I},\ det(\boldsymbol{R}) = 1\}$, and the Special Euclidean Group describes the group of rigid motion in 3-D, which is the semi-direct product of SO(3) and $\mathbb{R}^3$. It is defined as

$$\mathrm{SE}(3) = \left\{ \boldsymbol{T} = \begin{bmatrix} \boldsymbol{R} & t \\ \boldsymbol{0} & 1 \end{bmatrix} \in \mathbb{R}^{4\times4} | \boldsymbol{R} \in \mathrm{SO}(3), t \in \mathbb{R}^3 \right\}.$$

The Group SO(3) forms a smooth manifold, and in its tangent space of the identity is the Lie Algebra $\mathfrak{so}(3)$ (which can be identified with the rotation vectors) of SO(3). The Lie Algebra $\mathfrak{so}(3)$ consists of all skew-symmetric matrices $\boldsymbol{R} \in \mathbb{R}^{3\times3}$. This means that every skew-symmetric matrix in $\mathbb{R}^{3\times3}$ can be mapped by a vector $w \in \mathbb{R}^3$ by the hat operator $(\cdot)^\wedge$

$$\boldsymbol{w}^\wedge = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}^\wedge = \begin{bmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & w_1 \\ -w_2 & w_1 & 0 \end{bmatrix} \in \mathfrak{so}(3). \tag{2}$$

For the 3-D rotation group SO(3), we have its Lie algebra

$$\mathfrak{so}(3) = \left\{ \phi \in \mathbb{R}^3, \boldsymbol{\Phi} = \phi^\wedge \in \mathbb{R}^{3\times3} \right\}. \tag{3}$$

And for the 3-D transformation group SE(3), we have its Lie algebra

$$\mathfrak{se}(3) = \left\{ \boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{\rho} \\ \boldsymbol{\phi} \end{bmatrix} \in \mathbb{R}^6, \boldsymbol{\xi}^\wedge = \begin{bmatrix} \boldsymbol{\phi}^\wedge & \boldsymbol{\rho} \\ \boldsymbol{0}^\top & 0 \end{bmatrix} \in \mathbb{R}^{4\times4} \right\} \tag{4}$$

where $\boldsymbol{\rho} \in \mathbb{R}^{3\times3}$, $\boldsymbol{\phi} \in \mathfrak{so}(3)$. We can map a skew-symmetric matrix to a vector $\boldsymbol{w} \in \mathbb{R}^3$ using the vee operator $(\cdot)^\vee$.

The exponential map (at the identity) exp: $\mathfrak{so}(3) \to \mathrm{SO}(3)$ maps an element of the Lie Algebra to its Lie Group (the rotation matrix) and coincides with standard matrix exponential (Rodrigues' formula)

$$\exp(\phi^\wedge) = \boldsymbol{I} + \frac{\sin(\|\phi\|)}{\|\phi\|}\phi^\wedge + \frac{1 - \cos(\|\phi\|)}{\|\phi\|^2}(\phi^\wedge)^2. \tag{5}$$

The logarithm map (at the identity) associates a matrix $\boldsymbol{R} \neq \boldsymbol{I}$ in SO(3) to a skew-symmetric matrix

$$\log(\boldsymbol{R}) = \frac{\varphi \cdot (\boldsymbol{R} - \boldsymbol{R}^\top)}{2\sin\varphi} \text{ with } \varphi = \cos^{-1}\frac{tr(\boldsymbol{R}) - 1}{2}. \tag{6}$$

The capitalized are convenient shortcuts to map vector elements. For notational convenience, we adopt capitalized Exp and Log maps to map vector elements to the Rotation matrix and vice versa

$$\mathrm{Exp} : \mathbb{R}^3 \to \mathrm{SO}(3); \quad \phi \mapsto \exp(\phi)^\wedge \tag{7}$$

$$\mathrm{Log} : \mathrm{SO}(3) \to \mathbb{R}^3; \quad \boldsymbol{R} \mapsto \log(\boldsymbol{R})^\vee \tag{8}$$

which operate directly on vectors, rather than on skew-symmetric matrices in $\mathfrak{se}(3)$.

C. Derivatives on Lie Group SO(3)

Following [26], we define plus and minus operators to introduce increments between elements on the manifold of SO(3), which are expressed on the tangent space of the manifold. In our notation, we denote the plus and minus operators as $\oplus$ and $\ominus$, respectively. The $\oplus$ adds an incremental element in the Lie algebra $\mathfrak{so}(3)$ to an element within the Lie Group SO(3). The $\ominus$ subtracts two elements within the Lie Group SO(3) and yields an element within Lie Algebra $\mathfrak{so}(3)$.

Giving a 3-D rotation $f : \mathrm{SO}(3) \to \mathbb{R}^3$ of a 3-D point $\boldsymbol{p}$; $f(\boldsymbol{R}) = \boldsymbol{R}\boldsymbol{p}$, we have

$$\begin{aligned} \frac{\mathrm{d}\boldsymbol{R}\boldsymbol{p}}{\boldsymbol{p}} &= \lim_{\boldsymbol{\theta}\to 0} \frac{(\boldsymbol{R} \oplus \boldsymbol{\theta})\boldsymbol{p} \ominus \boldsymbol{R}\boldsymbol{p}}{\boldsymbol{\theta}} = \lim_{\boldsymbol{\theta}\to 0} \frac{(\boldsymbol{R}\mathrm{Exp}(\boldsymbol{\theta})\boldsymbol{p} - \boldsymbol{R}\boldsymbol{p}}{\boldsymbol{\theta}} \\ &= \lim_{\boldsymbol{\theta}\to 0} \frac{\boldsymbol{R}(\boldsymbol{I} + [\boldsymbol{\theta}]_\times)\boldsymbol{p} - \boldsymbol{R}\boldsymbol{p}}{\boldsymbol{\theta}} = \lim_{\boldsymbol{\theta}\to 0} \frac{\boldsymbol{R}[\boldsymbol{\theta}]_\times \boldsymbol{p}}{\boldsymbol{\theta}} \\ &= -\boldsymbol{R}[\boldsymbol{p}]_\times \in \mathbb{R}^{3\times3}. \end{aligned} \tag{9}$$

IV. Extrinsic Calibration

The most widely used method for extrinsic calibration can be divided into three steps: 1) feature extraction; 2) feature matching; and 3) extrinsic parameter estimation, as shown in Fig. 2. Common features include point features, line features, and surface features. Feature matching can be divided into two types: 1) descriptor-based matching, which calculates the similarity of features using descriptors, and 2) spatial-based matching, which calculates the spatial distance of features and selects matched features with similar spatial distances. Point feature matching is usually achieved by descriptors and is used more in camera images, while line feature matching often uses spatial distance, and line features are commonly used in cross-modal calibration. In this article, we extract line features and match them using spatial distance.

A. Edge Detection

1) LiDAR Edge Detection: LiDAR generates 3-D point clouds, capturing 3-D spatial information without being affected by changes in lighting. Extensive research has been conducted on extracting features from LiDAR point clouds, typically treating discontinuities in the 3-D point cloud as 3-D edges. However, the point clouds generated by LiDAR have bleeding edges, and extracting all edges would introduce significant errors and reduce the accuracy of the calibration algorithm. To avoid such error sources, connected edges are preferred. In this article, we primarily use the method proposed by Yuan et al. [13] to extract 3-D edges. Specifically, as shown

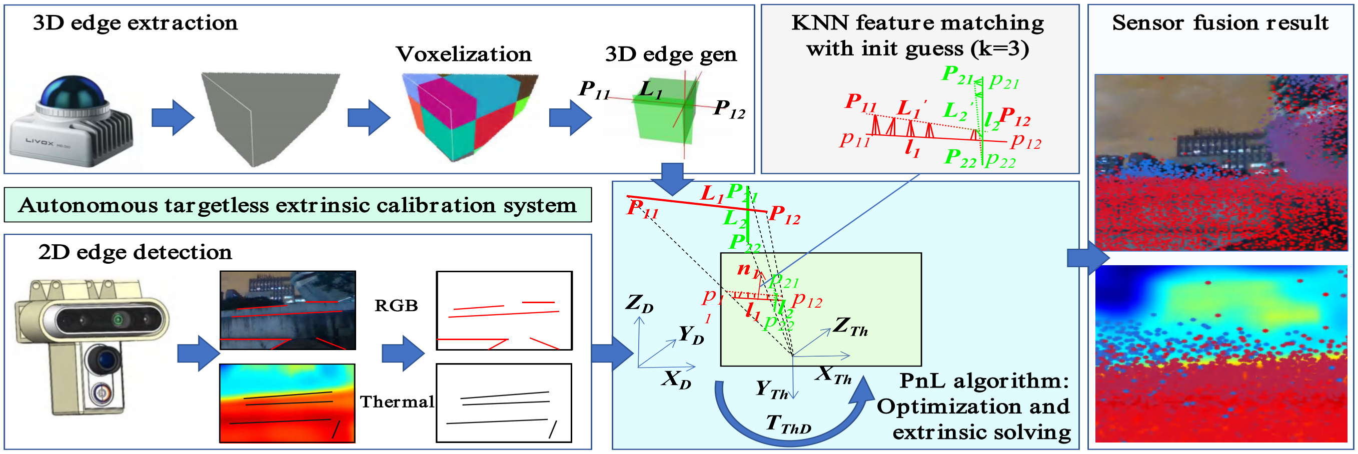


Fig. 2. Schematic of the autonomous–targetless extrinsic calibration process.

in the top-left of Fig. 2, the RANSAC algorithm is used to fit planes on dense point clouds. Then, the intersection lines of the planes are extracted as connected edges and prepared for subsequent calibration. For solid-state LiDAR such as the Livox series, dense point clouds can be obtained by accumulating point clouds at the same location. For mechanical (rotating) LiDARs, the LiDAR needs to be moved around and algorithms such as Fast-LIO [27] are run to generate dense point clouds.

2) Thermal Edge Detection: The key distinctions between thermal images and RGB images can be summarized in two aspects. First, most RGB images represent steady-state conditions, whereas thermal images capture transient states. Second, thermal edges tend to be blurrier compared to RGB edges. In most scenarios, the color and shape of an object remain unchanged, while its temperature changes, flowing from high to low. RGB images rely on visible light to capture information related to color and geometry, while thermal images capture temperature differences. Consequently, the dynamic nature of temperature variations introduces the difference between transient and steady states in thermal images (whereas RGB images can be considered as representing steady states), and it also contributes to the blurred appearance of thermal edges. The widely used Canny Edge algorithm, which has been optimized over the years for RGB images, is not specifically tailored for thermal images. Therefore, we have developed a novel edge detection algorithm to address the aforementioned issues.

Transient and steady-state temperature fields are influenced by the principles of Fourier's law, which governs the flow of heat transfer from high-temperature objects to low-temperature objects. In the absence of an external heat source, the temperature field of an isolated system with a temperature difference will evolve over time until the temperature becomes homogeneous. In a thermal image, black tends to "flow" toward white. In contrast, in an RGB image, there is no inherent "flow" from black to white based on temperature. Hence, we distinguish between transient temperature differences and steady-state temperature differences in thermal images based on whether the temperature changes over time (i.e., the presence or absence of an external heat source maintaining the temperature difference). We classify a stable environment that can be maintained during the period of data collection for calibration (approximately 30 min) as a steady state. On the other hand, scenes that need to be heated multiple times such as passive calibration patterns are categorized as transient.

In the absence of a stable heat source, artificially created temperature differences for calibration purposes are typically transient. A checkerboard pattern heated by a heat gun is an example of a transient state. In these artificially created transient temperature difference scenarios, due to their transient nature and the gradual reduction of temperature differences, the clarity of boundaries is limited, leading to increasingly blurred thermal edges during the calibration process.

Stable temperature differences in an environment are generated by heat sources such as human body temperature [22] and active calibration patterns that have heating resistors. When the heat dissipation from the air and the heat generated by these sources reach equilibrium, steady-state temperature differences occur, leading to local stability. Observing that the temperature variation of building contours is relatively slow, within the data collection period of several tens of minutes, it can be considered as a steady-state condition. Therefore, structured environments capable of generating temperature differences can be utilized as calibration scenes. This approach eliminates the need for external heat sources or active calibration patterns, reducing complexity and cost while still providing a suitable environment for calibration.

However, before utilizing building contours for calibration, it is necessary to address the issue of detecting blurry edges in thermal images. Heat transfer occurs at the boundaries between objects with different temperatures, resulting in a diffuse phenomenon at the edges, making thermal images typically blurrier compared to RGB images. As shown in Fig. 3, there are four common image edge models: step, ramp, pulse, and ridge. In thermal images, the temperature cannot exhibit a sudden change (or maintain a sudden change continuously) because heat conducts and flows. Unlike color and geometry, which can remain constant at a pixel, temperature values cannot. Therefore, step and pulse models, which are common in RGB images, are almost nonexistent in thermal images. The

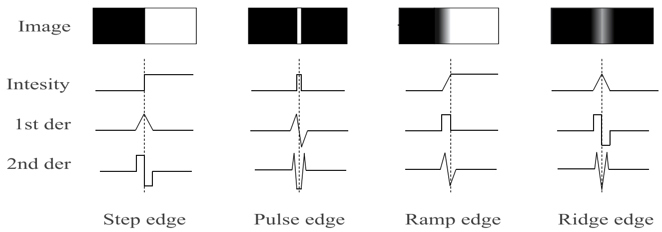


Fig. 3. Different edge models.

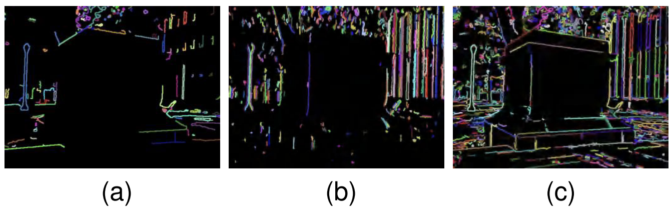


Fig. 4. Thermal edge extraction example. (a) Thermal edge detection with a Sobel operator. (b) Thermal edge detection with a second-order edge detector. (c) Thermal edge detection with a combined edge detector.

remaining ramp and ridge edge models are more prevalent in thermal images.

Most edge detection algorithms are developed for the common step and pulse models observed in RGB images, where the first-order gradient is used to detect edges. These algorithms are unable to effectively detect edges corresponding to the ramp and ridge models commonly found in thermal images. As shown in Fig. 3, the first-order derivative for ramp and ridge edge models is not local maximum, whereas the second-order derivative for the ridge model is the local maximum.

Ramp edges typically occur at the boundary of different mediums, such as the boundary of solid and air, which are also bleeding edges for LiDAR edges. On the other hand, ridge edges typically occur at the junctions of different materials or where there are geometric changes of the same material, such as corners of walls. Hence, ridge edges in thermal images correspond to connecting edges in the 3-D space, which are preferred for LiDAR extrinsic calibration. Therefore, this study proposes a method that combines the first- and second-order derivatives for thermal image edge detection. Traditional Canny edge detection [28] is used for RGB image edge detection. An example of the proposed edge detection method and Canny edge detection is shown in Fig. 4.

B. Feature Matching

According to the summary of Section II, feature matching is mainly divided into two types: the former is based on feature descriptor, which is often used for point features; and the latter is based on spatial relationship, which is mainly used for edge or line features. In this article, we propose a novel edge-matching method based on spatial relationships. The edge matching process is summarized in Algorithm 1. This method utilizes geometric constraints and edge orientation to achieve robust and accurate results. By incorporating the spatial information of edges, it can handle complex scenes with occlusions and cluttered backgrounds. The experimental results demonstrate that our method outperforms the state-of-the-art methods in accuracy and robustness.

Assume M 3-D edges are extracted from the 3-D point cloud and N 2-D edges are extracted from the RGB image or thermal image. Denote the set of edge points in 3-D point cloud as $\boldsymbol{P} = \{P_i^j; i = 1, \ldots, M; j = 1, \ldots, \text{edge}[i].\text{size}()\}$ and the set of edge points in the 2-D image plane as $\boldsymbol{p} = \{p_i^j; i = 1, \ldots, N; j = 1, \ldots, \text{edge}[i].\text{size}()\}$. To build the matching relationship between 2-D and 3-D edges, we first randomly sample multiple points in every 3-D edge, then project this 3-D edge point cloud to a 2-D image plane according to the initial extrinsic and search in the neighbor area of the projected points. Lastly, the image coordinate of the projected point is adjusted to the mean value of its k nearest neighbors. After all the sampled points have been matched, we can match the 3-D edge with the 2-D edges.

Algorithm 1 2-D and 3-D Edge Matching Algorithm

Input: 2D edges l, 3D edges L, thermal camera intrinsic, initial extrinsic

Output: Matching pairs between 2D and 3D edges

for $i = 1$ to $3D\_edges.size()$ **do**
  random sample k points in each edge
  for $j = 1$ to K **do**
    project 3D point P_i^j onto 2D image plane as p_i^j
    search q_i^j in the k nearest neighbors of p_i^j
    compute $q_i = \frac{1}{k}\sum_{pt=1}^{k} q_i^j$
    compute $\boldsymbol{n}_i = \sum_{pt=1}^{k}(q_i^j - q_i)(q_i^j - q_i)^\top$
  end for
end for
return edge pairs $\{L, l\}$

C. Extrinsic Calibration

Having edges extracted and matched, we will address the problem of extrinsic calibration of a camera $\mathbf{C}$ or a thermal camera $\mathbf{Th}$ and a LiDAR $\mathbf{LiDAR}$. Let $\mathbf{T}_{\text{CLiDAR}}$ be the relative pose between the reference frame from the LiDAR to the camera and $\mathbf{T}_{\text{ThLiDAR}}$ be the relative pose from the LiDAR to the thermal camera. To simplify notation, we denote $T = [R|t] \in \mathbb{SE}(3)$ as the relative pose from the LiDAR to the other two 2-D sensors.

1) PnL: As shown in Fig. 5, given a pair of 2-D-to-3-D line correspondence, we have two constraints on the camera pose [29]. Formulating the extrinsic solving as a PnL problem, the relative pose could be solved by at least three line correspondences. Most scenes hold more than three edge correspondences, and the extrinsic parameter could be solved by at least one shot.

Assume that the intrinsic matrix of the camera is K_C and the homogeneous coordinate of a 2-D line l_i^j is $\mathbf{l}_i^j$. The backprojection plane of l_i is $[K_C\mathbf{l}_i; 0]$ [29], and the normal of the 2-D line on the backprojection plane l_i^j is $\mathbf{n}_i^j$. A straight line could be represented by a unit direction vector $\mathbf{v}$ and a point $\mathbf{p}$ which is closest to the origin of the line. Together with the camera center, the 3-D line forms a norm vector in the camera frame $\mathbf{n_C} = (n_C^x, n_c^y, n_C^z)^\top$.

Considering one line correspondence in 3-D and 2-D space $L_i^j \leftrightarrow l_i^j$, the 2-D line in the thermal image is the intersection

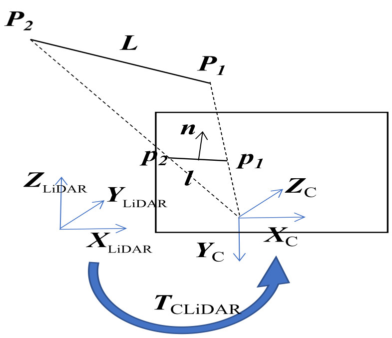


Fig. 5. Perspective-n-line algorithm schematic.

of the 3-D line with the thermal image plane. The 2-D line, the 3-D line, and the camera center lie on the same plane which can be parameterized by the camera center and a normal vector $\boldsymbol{n}$. As in (1), we can transform the normal vector from the LiDAR coordinate to the camera coordinate as follows:

$$\boldsymbol{n}_C = \boldsymbol{T}_{\text{CLiDAR}} \cdot \boldsymbol{n}^{\text{LiDAR}}. \tag{10}$$

For any 3-D point $\boldsymbol{P} = (x, y, z)^\top$ on the 3-D line and a 2-D point $p = (u, v, 1)^\top$ on its corresponding 2-D line, we have the constraint with the norm vector formed by the 3-D line and the camera center. After transforming the normal vector into the thermal camera coordinate, $\boldsymbol{n}_C$ can form a constraint with any point on the corresponding thermal edge

$$0 = \boldsymbol{n}_C(\boldsymbol{K}_C \boldsymbol{T}_{\text{CLiDAR}} \boldsymbol{P} - \boldsymbol{p})^\top. \tag{11}$$

The above equation can be viewed as the cost function of the extrinsic parameter. The extrinsic parameter can be obtained by solving the least-mean-square error problem

$$\boldsymbol{T}_{\text{CLiDAR}} = \arg \min_{\boldsymbol{T}_{\text{CLiDAR}}} \frac{1}{2} \left\| \sum_{i=1}^{n} \boldsymbol{n}_C (\boldsymbol{K}_C \boldsymbol{T}_{\text{CLiDAR}} \boldsymbol{P}_i - \boldsymbol{p}_i)^\top \right\|_2^2. \tag{12}$$

The main reprojection error is caused by noise orthogonal to the direction of the line and is unrelated to the line length. A scalar len, which is the length of the line, is applied to the distance function to make longer and shorter edges contribute equally to the cost function.

To obtain an unconstrained optimization form, we solve (12) on the tangent space of the manifold. As in (4), we use $\boldsymbol{\xi} \in \mathbb{R}^6$ to represent the relative transformation between the thermal camera and the range sensor. Then, the error term can be expressed as

$$\boldsymbol{e}(\boldsymbol{\xi}) = \boldsymbol{n}_C(\boldsymbol{K}_C \exp(\boldsymbol{\xi}^\wedge) \boldsymbol{P}_i - \boldsymbol{p}_i)^\top. \tag{13}$$

The derivative can be derived using the $\oplus$ and $\ominus$ operators defined in Section III-C

$$\frac{\partial \boldsymbol{e}}{\partial \delta \boldsymbol{\xi}} = \lim_{\boldsymbol{\xi} \to 0} \frac{\boldsymbol{e}(\delta \boldsymbol{\xi} \oplus \boldsymbol{\xi}) - \boldsymbol{e}(\boldsymbol{\xi})}{\delta \boldsymbol{\xi}} = \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{T}_{\text{CLiDAR}} \boldsymbol{P}} \frac{\partial \boldsymbol{T}_{\text{CLiDAR}} \boldsymbol{P}}{\partial \delta \boldsymbol{\xi}}. \tag{14}$$

Substituting (9) into (14), we have the gradient of the line point reprojection error (15), as shown at the bottom of the next page.

This nonlinear optimization problem is solved using the Levenberg–Marquardt method [30].

2) Uncertainty: Since sensor data such as LiDAR point cloud and thermal image can be noisy, there will be inevitably uncertainty in our calibration results. It is better to estimate the calibration uncertainty at the same time. The uncertainty is characterized by a covariance Σ of the calibration results.

V. Experiment Results

In this section, we evaluate our method by both simulation and real-world datasets. An Intel Realsense D435 RGBD camera, an iRay T3S thermal camera, and a Livox Mid-360 LiDAR mounted on a Totem-250 UAV frame are used to generate the datasets, as shown in Fig. 1. The LiDAR is installed on top of the UAV platform, while the thermal camera and the RGBD camera are assembled inside a 3-D printed rig which is mounted in the front of the UAV. The LiDAR, the stereo camera, and the thermal camera run at 10, 25, and 30 Hz, respectively. Some examples of the data captured by different sensors are shown in Fig. 6.

Section V-A introduces the simulation environment and results, followed by the introduction of real-world experiments in Section V-B. Table II gives the sensor specifications.

A. Simulation Results

A common problem for quantitatively assessing extrinsic calibration results is the unavailability of the exact ground truth [2]. Due to the lack of ground truth, many earlier efforts used manually annotated results as a benchmark and evaluated the calibration results qualitatively by contrasting the discontinuity in data fusion. Qualitative analysis only provides a general comparison and obscures the specific factors that influence the results. To fully describe the effectiveness of the suggested strategy and the impact of various variables, some researchers focused on simulation verification.

Benefiting from recent developments, we built a simulation suite that contains a thermal camera in the Gazebo platform. The simulation suite consists of a thermal camera, an RGBD camera, a LiDAR, and the calibration scene. The sensor properties, including resolution, field of view, and accuracy, are replicated in the simulation suite. As shown in Fig. 7, a thermal camera, an Intel Realsense D435 RGBD camera, and a LiDAR are modeled in the simulation suite. RGB images are captured by the left camera of the RGBD camera, and thermal images are captured by the thermal camera. To obtain a dense LiDAR point cloud, the LiDAR is slightly moved during the point cloud record process. Afterward, the LiDAR-inertial odometry Fast-lio [31] is used to track LiDAR motion and register all points to the LiDAR's initial pose. We build a calibration scene using an apartment model as the background and several walls with different temperatures as the foreground to generate edge features in the simulation environment. The point cloud generated by the LiDAR in the simulation is shown in Fig. 8.

In the simulation environment, all parameters are under control, so we assume perfect feature detection and matching. Under this assumption, the factors that may influence the

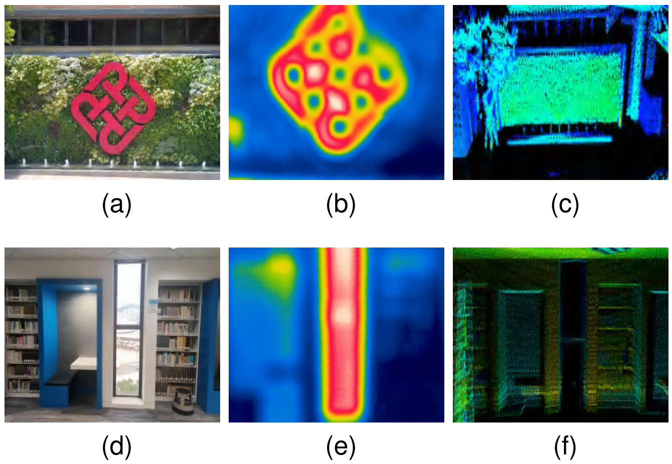


Fig. 6. Example scenes in different modality. (a) and (d) RGB image, (b) and (e) thermal image, and (e) and (f) LiDAR point cloud.

TABLE II
SENSOR SPECIFICATIONS

Device	Modality	Resolution	FOV
iRay T3S	Thermal	384×288	28.2° × 21.3°
Intel Realsense D435	RGB	1920×1080	69.4° × 42.5°
Livox Mid-360	LiDAR	1.3° × 1.3°	360°× 59°

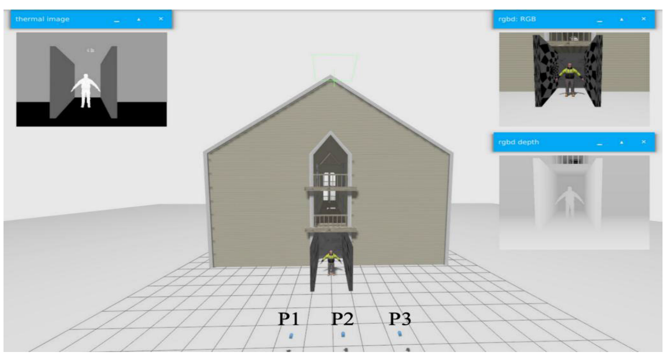


Fig. 7. Simulation in Gazebo, RGB, depth, thermal image, and LiDAR point clouds are captured. Different temperature is assigned to the ground, different walls, and human models. The house has the same temperature as the gray wall.

calibration results could be categorized into two folds, namely: 1) the number of frames captured at different relative poses between the sensor rig and the calibration scene and 2) the noise level added to the data captured by the different sensors. Since the FOV of the thermal camera is relatively small, the thermal camera is placed right under the RGBD camera and the LiDAR. All sensors heading in the same direction, so that the image and point cloud captured by the three different sensors can be fused effectively.

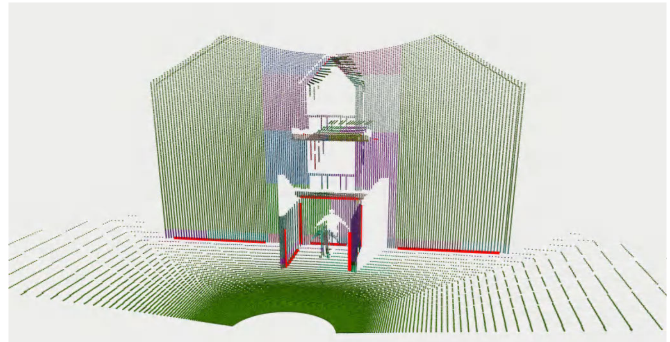


Fig. 8. LiDAR point cloud and 3-D edge extraction in RVIZ.

TABLE III
MEAN (AND STD DEV) OF LINEAR (e_t) AND ANGULAR (e_r) CALIBRATION ERRORS IN DIFFERENT POSES

	Error	P1	P2	P3
LiDAR-thermal	e_t(cm)	8.31(1.329)	7.89(1.326)	8.44(1.458)
	e_r(degree)	5.72(0.718)	5.39(1.078)	5.98(0.583)
LiDAR-RGB	e_t(cm)	3.37(1.102)	3.52(1.287)	3.58(1.541)
	e_r(degree)	3.69(1.219)	3.85(1.320)	3.32(1.491)

1) Single-Pose Experiments: In this section, we feed only one frame into the pipeline to demonstrate the feasibility of single-frame calibration. To evaluate the impact of acquiring data from different relative positions and angles on the calibration results, we collect data at three distinct points: $P1$, $P2$, and $P3$ shown in Fig. 7. Given the ground-truth transformation $\boldsymbol{T}_{gt}$ and the estimated transformation $\hat{\boldsymbol{T}}$, we compute the transformation error in terms of rotation error $e_{\boldsymbol{r}}$ and translation error $e_{\boldsymbol{t}}$ as follows:

$$e_r = \|\mathrm{Log}(\boldsymbol{R}_{gt}^{\top}\boldsymbol{R})\| = \|\mathrm{Log}(\delta\boldsymbol{R})\| \tag{16}$$

$$e_t = \|\boldsymbol{t}_{gt} - \hat{\boldsymbol{t}}\| = \|\delta\boldsymbol{t}\| \tag{17}$$

where $\boldsymbol{R}$ is the rotation matrix, Log() operation refers to (7), and $\boldsymbol{t}$ is the translation vector. $\boldsymbol{R}$ and $\boldsymbol{t}$ together form the transformation matrix as in (4). The single pose calibration results are shown in Table III.

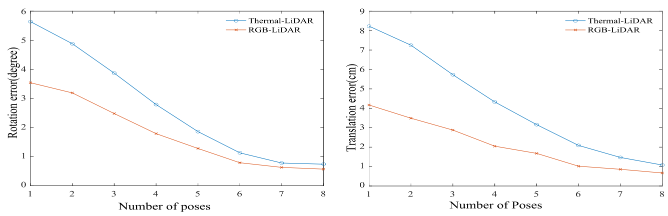
From the table, it can be observed that overall, the algorithm provides more accurate and robust estimates for rotation compared to translation. Both rotation and translation accuracy vary among different sensor combinations, with the LiDAR–RGBD combination being the more precise and the LiDAR–thermal combination showing the lowest accuracy. We also set identical camera parameters for both thermal and RGB cameras with resolutions of 320 × 240 and 640 × 480 in the Gazebo simulation. The FoV of all cameras is set as 60° × 47°. Table IV presents the experimental results. It is evident from the table that the reprojection error remains consistent across all control groups, as the PnL algorithm

$$\frac{\partial \boldsymbol{e}}{\partial \delta \boldsymbol{\xi}} = -\boldsymbol{n}_i^{\top} \begin{bmatrix} \dfrac{f_x}{Z'} & 0 & -\dfrac{f_x X'}{Z^2} & -\dfrac{f_x XY}{Z^2} & f_x + \dfrac{f_x X^2}{Z^2} & -\dfrac{f_x Y}{Z} \\ 0 & \dfrac{f_y}{Z} & -\dfrac{f_y Y}{Z^2} & -f_y - \dfrac{f_y Y^2}{Z^2} & \dfrac{f_y XY}{Z^2} & \dfrac{f_y X}{Z} \end{bmatrix}. \tag{15}$$

TABLE IV

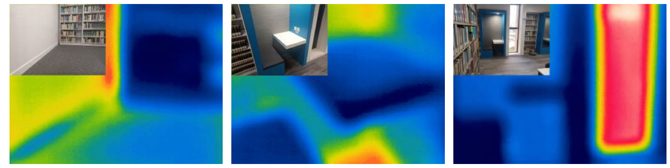
CALIBRATION ACCURACY COMPARISON USING CAMERAS WITH DIFFERENT RESOLUTIONS IN GAZEBO SIMULATION

Sensor	Resolution	e_t(cm)	e_r(deg)	e_{res}(pix)
Thermal	320 × 240	8.42	4.43	1.2
Thermal	640 × 480	3.78	3.26	1.3
RGB	320 × 240	8.26	4.39	1.1
RGB	640 × 480	3.85	3.01	1.2



(a) (b)

Fig. 9. Calibration result affected by the number of poses. (a) Rotation error. (b) Translation error.



(a) (b) (c)

Fig. 10. Example calibration scenes. (a) Library corner. (b) Library seat. (c) Library window.

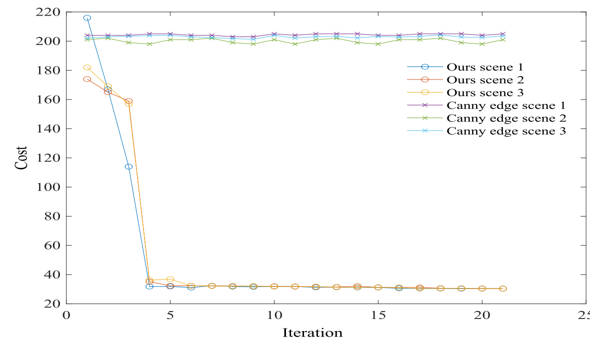


Fig. 11. Cost comparison using the proposed thermal edge detector and the Canny edge detector of LiDAR–thermal calibration in different scenes.

minimizes the reprojection error in pixel values. However, due to the difference in camera resolution, with the same reprojection error, the translation error is nearly twice as large for small-resolution cameras.

In conclusion, the results obtained from single-frame calibration demonstrate the effectiveness of the algorithm and lay the foundation for multiframe calibration. In addition, the comparison between different resolutions indicates that, with the same reprojection error, the absolute error is influenced by the camera resolution. The higher the resolution, the more precise the absolute transformation accuracy that can be achieved.

2) Multipose Experiment: The accuracy of the calibration results in the context of multipose experiments can be influenced by various sources of noise inherent to the method, such as sensor noise and feature detection errors. When the sensor rig is posed at different positions and data is collected using diverse approaches, including features with varying depths and noncoplanar lines, additional constraints can be obtained by providing new poses. This process has the potential to decrease ambiguity and enhance accuracy.

The rotation and translation errors decreased with the increase of the frames used in the calibration process as shown in Fig. 9. As can be seen, the rotation error reaches a relatively stable state when the number of input frames is more than 6, while translation estimation is more prone to noise and reaches a high accuracy when the number of input frames exceeds 7.

B. Real-World Experiment

In this section, our system is evaluated in real-world experiments, both quantitatively and qualitatively. The ground-truth extrinsic parameters are obtained by manually selecting corresponding feature points between ten frames and applying the PnP algorithm based on these correspondences.

We test our method in a variety of scenes, as depicted in Fig. 10. Data collection is guided by the analysis in Section IV-A, considering the thermal properties and geometric edge distributions to ensure alignment between thermal edges and geometric edges.

1) Convergence Validation: To assess the convergence of the proposed method for estimating extrinsic parameters, experiments are conducted under three distinct scenarios as depicted in Fig. 10. A randomized initialization approach is employed to ensure robustness, with initial values sampled within a range of ±5° in rotation and 5 cm in translation. Each scenario is executed 20 times to yield statistically significant results. The cost function is evaluated for each run, and the statistical analysis of the results is presented in Fig. 11. Our findings demonstrate successful convergence of the proposed algorithm in all scenarios, with final estimates exhibiting minimal cost. These results indicate the reliability and effectiveness of our method in accurately estimating extrinsic parameters across diverse scenes.

2) Precision and Accuracy: To evaluate the precision of the proposed method, we calculate the calibration error relative to the mean value of extrinsic statistics to test the variance of the proposed method. All results are obtained running batch optimization on eight frames according to the simulation results that the extrinsic error stability is observed upon surpassing the threshold of eight frames of data. We compared our method with several others, including an indirect calibration method as a control group, that is, $T_{\text{ThLiDAR}} = T_{\text{ThC}} \cdot T_{\text{CLiADR}}$, as used by Fu et al. [23]. Although the error in T_{CLiDAR} by Fu et al.'s [23] method is similar to other methods, the overall error of T_{ThLiDAR} is larger due to the accumulated uncertainty of the indirect process. The boxplot in Fig. 12 shows that in the six axes for rotation and translation, our method achieved the smallest variance in five directions out of six axes in LiDAR–RGB calibration, and the smallest variance in four directions out of six axes in LiDAR–thermal calibration. In the yaw direction, our mean error is closer to zero.

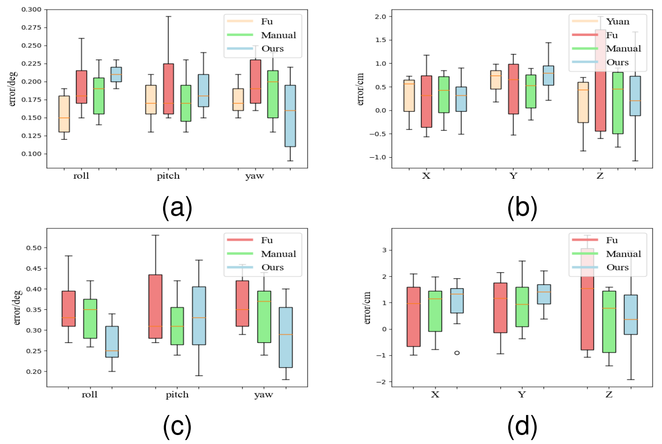


Fig. 12. Comparison experiment of calibration accuracy. (a) LiDAR–RGB rotation error. (b) LiDAR–RGB translation error. (c) LiDAR–thermal rotation error. (d) LiDAR–thermal translation error.

TABLE V
EXTRINSIC CALIBRATION COMPARISON

Method	LiDAR-RGB			LiDAR-thermal		
	e_t(cm)	e_r(deg)	e_{res}(pix)	e_t	e_r	e_{res}
Yuan's [13]	**2.75**	**0.17**	1.3	-	-	-
Fu's [23]	2.81	0.23	1.9	4.39	0.48	2.3
Ours	2.77	0.21	**1.2**	**3.86**	**0.46**	**1.3**
Manual	2.30	0.19	1.3	4.20	0.37	1.8

To further validate the effectiveness of our proposed method, a comparative analysis is conducted with the extrinsic from the CAD model as ground truth. In real experiments, we do not have the absolute ground truth as in simulations. Therefore, we use a CAD model as the ground truth. However, it should be noted that using CAD for extrinsic calibration can introduce large reprojection errors. Table V presents the comparison results of calibration errors. It is evident that our method achieves superior results in terms of reprojection error among autonomous–targetless extrinsic calibration compared to Yuan et al. [13] and Fu et al. [23] for both LiDAR–RGB and LiDAR–thermal extrinsic calibration. In terms of translation and rotation accuracy, our method achieves results comparable to other methods. In addition, it performs on par with the manual method. Particularly, our method demonstrates remarkable accuracy in both LiDAR–RGB calibration and LiDAR–thermal calibration, with overall translation errors of 2.77 and 3.86 cm, and overall rotation errors of 0.21° and 0.46°, respectively. Moreover, our method enables the direct and autonomous–targetless calibration of both the thermal camera and LiDAR. Previously, no existing method offered the capability to directly calibrate the combination of LiDAR and thermal sensors in such a manner.

3) Fusion Results: Fig. 13 shows the fusion result of the LiDAR point cloud and the RGB image in the stereo camera. It can be seen from the fusion result that the LiDAR point cloud edges are in good alignment with the RGB image edges. LiDAR points near the LiDAR center are represented by red color and points far from the LiDAR are represented by blue color. The trend of color changes in the image is consistent with reality.

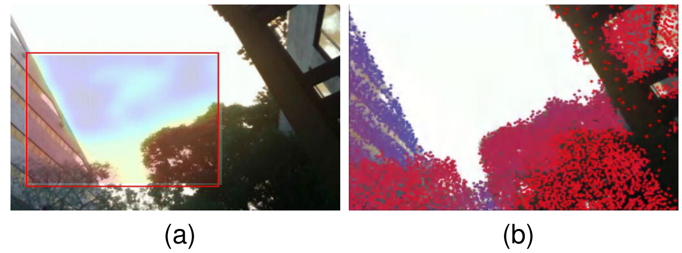


Fig. 13. Image fusion result of the RGB image and the LiDAR point cloud. (a) Raw RGB image, the FoV of the thermal camera is shown in the red box. (b) RGB image with LiDAR point cloud projection.

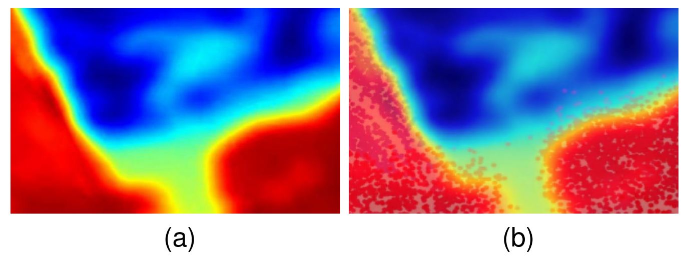


Fig. 14. Image fusion result of the thermal image and the LiDAR point cloud. The FoV of the thermal camera is shown in the red box in Fig. 13. (a) Thermal image. (b) Thermal image with LiDAR point cloud projection.

Fig. 14 shows a qualitative fusion result of the LiDAR point cloud and the thermal image at the same scene as depicted in Fig. 13, using the parameters obtained by our calibration method. Since the FOV of the thermal camera is much smaller than the RGBD camera and LiDAR, the thermal information only covers a part of the other two sensors. Still, it demonstrates the feasibility of the proposed method.

VI. CONCLUSION

In this article, we have presented an autonomous–targetless calibration method for LiDAR–RGB and LiDAR–thermal calibration, representing the first autonomous–targetless approach capable of directly calibrating a thermal camera and a LiDAR, as supported by the literature review. The method extracts line features from 2-D thermal images and 3-D point clouds and solves the extrinsic parameters based on the line feature correspondences. We have developed a simulation calibration suite on the Gazebo platform, incorporating a thermal camera. The performance of the proposed method has been rigorously assessed through systematic estimation using synthetic data and qualitative validation using real-world data. In both LiDAR–RGB calibration and LiDAR–thermal calibration, the method has demonstrated exceptional accuracy, with overall translation errors of 2.77 and 3.86 cm and overall rotation errors of 0.21° and 0.46°, respectively. These results are comparable to those achieved by traditional target-based manual methods. Different thermal scenes have been analyzed based on the thermography process. The presence of distinguishable and well-aligned edges in different modalities (thermal–RGB–LiDAR) has been identified as the key to obtaining ideal calibration results. As part of future work, we plan to explore autonomous bad scene rejection mechanisms and enhance calibration accuracy through the utilization of a more precise thermal edge detection method.

References

[1] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[2] J. Beltrán, C. Guindel, A. de la Escalera, and F. García, “Automatic extrinsic calibration method for LiDAR and camera sensor setups,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 17677–17689, Oct. 2022.

[3] S. Vidas, R. Lakemond, S. Denman, C. Fookes, S. Sridharan, and T. Wark, “A mask-based approach for the geometric calibration of thermal-infrared cameras,” *IEEE Trans. Instrum. Meas.*, vol. 61, no. 6, pp. 1625–1635, Jun. 2012.

[4] J. Zhang, R. Zhang, Y. Yue, C. Yang, M. Wen, and D. Wang, “SLAT-Calib: Extrinsic calibration between a sparse 3D LiDAR and a limited-FOV low-resolution thermal camera,” in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2019, pp. 648–653.

[5] A. Khurana and K. S. Nagla, “Extrinsic calibration methods for laser range finder and camera: A systematic review,” *MAPAN*, vol. 36, no. 3, pp. 669–690, Sep. 2021.

[6] X. Li, Y. Xiao, B. Wang, H. Ren, Y. Zhang, and J. Ji, “Automatic targetless LiDAR–camera calibration: A survey,” *Artif. Intell. Rev.*, vol. 56, no. 9, pp. 1–39, 2022.

[7] J.-Y. Bouguet. (2004). *Camera Calibration Toolbox for MATLAB*. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/

[8] M. Rufli, D. Scaramuzza, and R. Siegwart, “Automatic detection of checkerboards on blurred and distorted images,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2008, pp. 3121–3126.

[9] D. Feng et al., “Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges,” *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1341–1360, Feb. 2020.

[10] J. Maye, P. Furgale, and R. Siegwart, “Self-supervised calibration for robotic systems,” in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, pp. 473–480.

[11] R. Unnikrishnan and M. Hebert, “Fast extrinsic calibration of a laser rangefinder to a camera,” Robot. Inst., Pittsburgh, PA, USA, Tech. Rep. CMU-RI-TR-05-09, 2005.

[12] Z. Pusztai and L. Hajder, “Accurate calibration of LiDAR-camera systems using ordinary boxes,” in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 394–402.

[13] C. Yuan, X. Liu, X. Hong, and F. Zhang, “Pixel-level extrinsic self calibration of high resolution LiDAR and camera in targetless environments,” *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7517–7524, Oct. 2021.

[14] Y. Lin, V. Larsson, M. Geppert, Z. Kukelova, M. Pollefeys, and T. Sattler, “Infrastructure-based multi-camera calibration using radial projections,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2020, pp. 327–344.

[15] L. Zhou, Z. Li, and M. Kaess, “Automatic extrinsic calibration of a camera and a 3D LiDAR using line and plane correspondences,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 5562–5569.

[16] T. Tóth, Z. Pusztai, and L. Hajder, “Automatic LiDAR-camera calibration of extrinsic parameters using a spherical target,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 8580–8586.

[17] J. Levinson and S. Thrun, “Automatic online calibration of cameras and lasers,” in *Robotics: Science and Systems*, vol. 2. Berlin, Germany: Citeseer, 2013, p. 7.

[18] G. Pandey, J. R. McBride, S. Savarese, and R. M. Eustice, “Automatic targetless extrinsic calibration of a 3D LiDAR and camera by maximizing mutual information,” in *Proc. 26th AAAI Conf. Artif. Intell.*, 2012, pp. 2053–2059.

[19] M. R. U. Saputra, C. X. Lu, P. P. B. de Gusmao, B. Wang, A. Markham, and N. Trigoni, “Graph-based thermal–inertial SLAM with probabilistic neural networks,” *IEEE Trans. Robot.*, vol. 38, no. 3, pp. 1875–1893, Jun. 2022.

[20] S. Prakash, P. Y. Lee, T. Caelli, and T. Raupach, “Robust thermal camera calibration and 3D mapping of object surface temperatures,” in *Thermosense XXVIII*, vol. 6205. Bellingham, WA, USA: SPIE, 2006, pp. 182–189.

[21] S. Vidas, P. Moghadam, and M. Bosse, “3D thermal mapping of building interiors using an RGB-D and thermal camera,” in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 2311–2318.

[22] J. T. Lussier and S. Thrun, “Automatic calibration of RGBD and thermal cameras,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2014, pp. 451–458.

[23] T. Fu, H. Yu, W. Yang, Y. Hu, and S. Scherer, “Targetless extrinsic calibration of stereo, thermal, and laser sensors in structured environments,” *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–11, 2022.

[24] J. Zhang, P. Siritanawan, Y. Yue, C. Yang, M. Wen, and D. Wang, “A two-step method for extrinsic calibration between a sparse 3D LiDAR and a thermal camera,” in *Proc. 15th Int. Conf. Control, Autom., Robot. Vis. (ICARCV)*, Nov. 2018, pp. 1039–1044.

[25] J. Zhang, Y. Liu, M. Wen, Y. Yue, H. Zhang, and D. Wang, “$L^2V^2T^2$Calib: Automatic and unified extrinsic calibration toolbox for different 3D LiDAR, visual camera and thermal camera,” in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2023, pp. 1–7.

[26] J. Sola, J. Deray, and D. Atchuthan, “A micro lie theory for state estimation in robotics,” 2018, *arXiv:1812.01537*.

[27] W. Xu and F. Zhang, “FAST-LIO: A fast, robust LiDAR-inertial odometry package by tightly-coupled iterated Kalman filter,” *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3317–3324, Apr. 2021.

[28] J. Canny, “A computational approach to edge detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[29] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[30] D. W. Marquardt, “An algorithm for least-squares estimation of nonlinear parameters,” *J. Soc. Ind. Appl. Math.*, vol. 11, no. 2, pp. 431–441, Jun. 1963.

[31] 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, Aug. 2022.