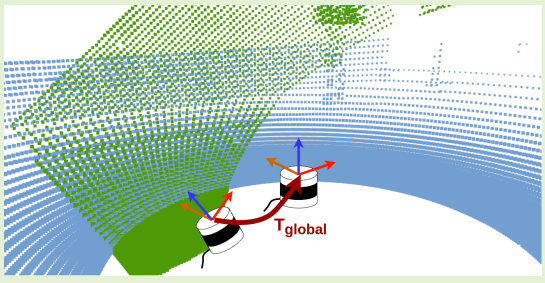
De Miguel, M. Á., Guindel, C., Al-Kaff, A., & García, F. (2023). High-Accuracy Patternless Calibration of Multiple 3D LiDARs for Autonomous Vehicles. *IEEE Sensors Journal*, 23(11), 12200-12208.

DOI: 10.1109/jsen.2023.3268338

High-Accuracy Patternless Calibration of Multiple 3D LiDARs for Autonomous Vehicles

Miguel Ángel de Miguel[1], *Member, IEEE*, Carlos Guindel[1], Abdulla Al-Kaff[1,2], *Member, IEEE*, and Fernando García[1,3], *Member, IEEE*

***Abstract*— This paper proposes a new method for estimating the extrinsic calibration parameters between any pair of multi-beam LiDAR sensors on a vehicle. Unlike many state-of-the-art works, this method does not use any calibration pattern or reflective marks placed in the environment to perform the calibration; in addition, the sensors do not need to have overlapping fields of view. An iterative ICP-based process is used to determine the values of the calibration parameters, resulting in better convergence and improved accuracy. Furthermore, a setup based on the CARLA simulator is introduced to evaluate the approach, enabling quantitative assessment with ground-truth data. The results show an accuracy comparable with other approaches that require more complex procedures and have a more restricted range of applicable setups. This work also provides qualitative results on a real setup, where the alignment between the different point clouds can be visually checked. The open-source code is available at https://github.com/midemig/pcd_calib.**



***Index Terms*— Autonomous driving, extrinsic calibration, iterative closest point, LiDAR, sensor fusion**

I. Introduction

LiDAR is currently one of the most commonly used sensor technologies in applications such as robotics and autonomous driving. This modality provides a point cloud with 3D spatial information that can be used for multiple tasks, including obstacle detection, mapping, and localization [1]. Early manufactured models tended to have a single plane, so they provided scarce information; however, this type of sensor can nowadays produce a high-resolution point cloud with up to 128 scan planes [2].

Although the increase in the number of layers involves a higher data resolution and, therefore, a more detailed representation of the environment, it also entails a significantly higher cost that does not translate into a wider field of view. For this reason, the current trend in applications requiring an extensive perception range (e.g., 360º around a vehicle) is more towards using several lower resolution LiDAR scanners placed in different positions and orientations to avoid blind spots [3]. In some cases, this approach leads to configurations where the sensors do not share a common field of view, as in the example shown in Fig. 1. However, some applications require merging the point clouds from these different LiDAR scanners to obtain a comprehensive view of the environment.

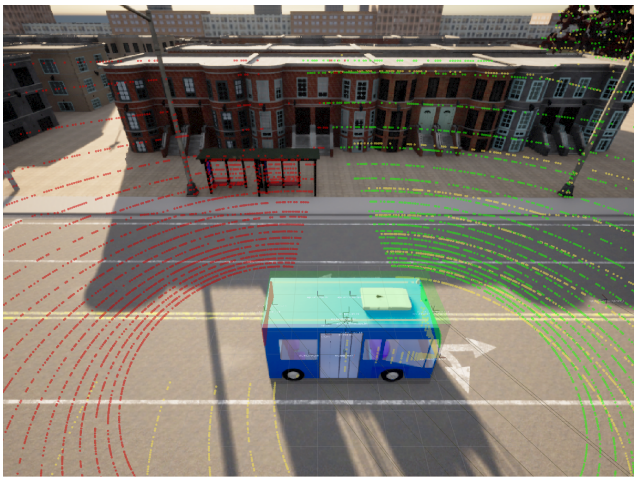


Fig. 1. Typical two-LiDAR configuration in a long vehicle where no scan points are shared between sensors as they are placed on the longitudinal ends of the vehicle roof.

In practice, this involves expressing the 3D points from all sensors in a joint coordinate system, which necessarily relies on the accurate calibration of the sensors involved.

Two types of sensor calibration are typically identified: intrinsic and extrinsic. When dealing with LiDAR, the first one consists of calculating the position of each 3D point with respect to the sensor coordinate system and is usually determined by the manufacturer. The second type of calibration, extrinsic calibration, aims to determine the relative position of

[1]The authors are with the Autonomous Mobility and Perception Lab (AMPL), University Carlos III of Madrid, Leganés, Madrid, 28911 Spain.
[2]Visiting Professor at Faculty of Engineering - German International University (GIU), Cairo, Egypt
[3]Visiting Professor at Smart Sensor Systems Group, The Hague University of Applied Sciences, Delft, the Netherlands

every sensor with respect to each other and is the main focus of this work.

This work presents an accessible yet effective method to obtain the extrinsic parameters of pairs of LiDAR scanners that is especially suitable for autonomous driving perception stacks. Our original contributions are listed below:

- We introduce a calibration pipeline based on the accumulation of point clouds from only one of the devices to be calibrated, thus allowing for a more hassle-free application of the necessary odometry algorithm.
- We propose and validate an iterative strategy to register clouds through the ICP method that minimizes the impact of its tolerance parameter, improving the accuracy of the results even when only a very rough initialization is available.

The method does not require a fiducial calibration pattern or distinctive reflective elements; instead, calibration can be performed on simple driving scenarios with a flat road and a few vertical elements (such as buildings). This removes the burden of manufacturing and handling such elements and creating complicated calibration scenarios. Besides, LiDAR scanners to be calibrated do not need to share a common field of view, contrary to what is usually the case with patternless methods based on the correlation between two point clouds. The approach requires a very limited amount of data to be carried out and can provide results in a short space of time.

Experimental results show that the accuracy of the approach is on par with other alternatives in the literature, being appropriate for use in automotive perception applications. However, it also has the advantages described above, which allow the range of applicability of the proposal to be extended beyond the more simple sensor placements and fulfill the requirements of cutting-edge setups.

The paper is structured as follows: Section II reviews state-of-the-art works for calibration, Section III describes the proposed method, Section IV shows the experiments and results to evaluate the method, and Section V presents the conclusions.

II. Related Work

Traditionally, extrinsic calibration of sensors in robotic platforms has been performed manually, either by determining the relative positions by hand through distance and angle measurement tools or by adjusting the parameters until visually pleasing results are obtained, in an iterative process that seeks to match the data from both sensors in the common field of view. However, these methods struggle to achieve the levels of accuracy required in perception, as even tiny errors in the estimation of the parameters can lead to substantial pairing errors (e.g., an error of 1° in rotation translates into a deviation of almost $1\,\mathrm{m}$ for measurements at a distance of $50\,\mathrm{m}$), with a consequent decline in the performance of algorithms using sensor data.

Focusing on the case of LiDAR scanners, these difficulties, coupled with the fact that manual calibration is usually a tedious and time-consuming process, have led to an intense research interest in recent years, resulting in numerous automatic calibration approaches able to provide the relative position of all LiDAR devices within the perception stack.

Sometimes, calibration is performed with respect to a coordinate system fixed to the robotic or automotive platform, from which the relative position among sensors can also be straightforwardly derived. That coordinate system can be located at the vehicle frame as in [4], where a method based on the iterative closest point (ICP) algorithm is used to determine the position of the sensor with respect to a robotic arm, or in [5], where a precise 3D model of an excavator is required to apply generalized ICP (GICP) to get the calibration parameters. Both methods report good calibration results, but they are oriented towards a specific type of vehicle, namely a robotic arm and an excavator with a moving arm, making them hardly generalizable to any ground vehicle.

The IMU sensor can also be established as a reference, as in [6], which uses an extended Kalman filter algorithm that exploits the motion-based calibration constraint for state update, or in [7], which tries to optimize the quality of a 3D point cloud produced by the LiDARs on the vehicle as it traverses an unknown environment.

Finally, calibration can also be performed with respect to sensors of other modalities, such as cameras. Such calibration methods frequently require a pattern with specific characteristics [8], [9] or are based on the correspondence between measurements provided by both sensors [10], [11]. Therefore, their application is inevitably limited to use cases where all LiDARs share part of their field of view with the camera, which is not always the case in automotive setups; e.g., in the configuration in Fig. 1, the measurements from the rear LiDAR will be outside the field of view of a hypothetical frontal camera.

Among the methods explicitly aimed at the calibration between multiple LiDARs, some present a similar problem to the one stated above; e.g., [12] analyzes the coincident planes between two LiDARs to obtain the calibration parameters.

Some methods accumulate successive LiDAR point clouds as the vehicle moves to overcome this limitation, enabling the superposition of points between the different sensors. In general, such methods seek to align the accumulated point clouds to obtain the translation and rotation of one cloud with respect to the other, which corresponds to the sensors' extrinsic parameters. Within this group of approaches, [13], [14] use reflective objects, easily detected by the LiDAR, as a reference in both point clouds to perform the alignment. Noteworthily, these methods rely on a good detection of the landmarks in both sensors' data, which can be challenging for some particular setups as it may require placing the targets at locations far from the vehicle if not outright inaccessible.

Closer to our work, other approaches dispense with the targets and apply ICP on point clouds from naturalistic scenes to determine the transformation between sensors. For example, [15] applies ICP to align each point cloud from a sensor with the point cloud from the other sensor that is closest in time. A LiDAR-based SLAM method is used to express the clouds in the same time reference. However, this method performs ICP using only separate point clouds, where there may be cases in which the closest clouds do not have enough

common points. Indeed, our tests using an implementation of this method have presented problems of convergence to an accurate solution multiple times.

On the other hand, Liu et al. [16] proposed a LiDAR-LiDAR and LiDAR-camera calibration method that uses point clouds retrieved at different poses and formulates the problem as a bundle adjustment problem where both the extrinsic parameters and the LiDAR poses are optimized concurrently. A custom adaptive voxelization strategy is applied to the input clouds to reduce the computational cost of the optimization, although it introduces the need for some manual parameter tuning depending on the properties of the LiDAR devices.

In conclusion, whereas a wide variety of methods has been proposed to solve the LiDAR extrinsic calibration problem, existing solutions either present limitations when applied outside the domain for what they were designed or have requirements that can be unfeasible for some particular sensor setups found in autonomous vehicles (e.g., with non-overlapping fields of view). This work aims to provide an accurate calibration solution applicable to a broader range of sensor configurations without requiring complex calibration scenarios, ad-hoc fiducial devices, or meticulous parameter tuning.

III. Method Overview

We propose a method to estimate the 3D transformation (i.e., the relative 6-DOF pose) between two LiDAR scanners based on their output data. Both devices are assumed to be attached to the body of a vehicle and placed in arbitrary positions and orientations.

The point clouds provided by each sensor at time step t are denoted by C_T^t and C_S^t, respectively. The subscript T designates the 'target' LiDAR, whereas S corresponds to the 'source' LiDAR; we will elaborate on this terminology later. Sensors must be synchronized so that both clouds are timestamped with respect to a shared time reference; i.e., the time offset between them must be known.

The method's outcome is a set of transformation parameters $T_{calib} = \{x, y, z, q_0, q_1, q_2, q_3\}$ comprising a three-axis translation (x, y, z) and a rotation given as a quaternion (q_0, q_1, q_2, q_3).

An overview of the approach is shown in Fig. 2. The different parts of the calibration pipeline will be described in the following sections.

A. Sensor Odometry

The method relies on sequences of LiDAR point clouds scanned while the vehicle is moving. In that way, the point clouds can be accumulated over time to generate a denser 3D reconstruction where better correspondences can be found.

This approach requires determining precisely the position of the target LiDAR at each time stamp. Therefore, the choice of the localization method is critical for the accuracy of the calibration, although the results will show later that the proposed method is remarkably robust to positioning errors within a wide range.

GNSS receivers are typically the method of choice for outdoor localization; however, using this kind of localization for the present task would require an extremely accurate calibration between the GNSS receiver and the target LiDAR so that the localization data could be effectively referenced to the LiDAR frame. In practice, that step could introduce a non-negligible bias that would affect the quality of the resulting LiDAR-LiDAR calibration.

In contrast, we propose employing a LiDAR odometry approach that avoids this intermediate transformation by directly providing the LiDAR scanner localization. To that end, different methods have been proposed in the literature; one of the most popular paradigms is LOAM [17], which has given rise to numerous variants, such as the lighter LegoLOAM [18]. Our setup uses LOAM to obtain the location of the target LiDAR scanner, thus benefiting from its high accuracy and robustness.

Let L_{LOAM}^t denote the location, expressed as a transformation, with respect to the origin of the target LiDAR point cloud C_T at time step t, for t ranging from 0 to T. For practical reasons, we set the origin at $L_{\text{LOAM}}^{T/2}$ and refer the rest of the locations to this point. Once the complete set of localizations provided by the LiDAR odometry is available, each of the target LiDAR point clouds can be transformed to the origin and concatenated to obtain the reconstruction C_T^{full}:

$$C_T^{\text{full}} = \bigcup_{t=0}^{T} (L_{\text{LOAM}}^{T/2})^{-1} \cdot L_{\text{LOAM}}^t \cdot C_T^t \tag{1}$$

Contrarily, point clouds from the source LiDAR are not accumulated. In that way, the need for accurate localization is reduced to only one of the devices, allowing the source scanner to be placed in unusual poses (i.e., extreme orientations) that would otherwise degrade the performance of odometry algorithms such as LOAM.

B. ICP Point Cloud Matching

The core of the proposed approach lies in the registration of pairs of point clouds, where each of them is obtained from one of the LiDAR scanners to be calibrated. This task is performed by employing the iterative closest point (ICP) algorithm [19], which estimates the transformation that minimizes the distance between two point clouds. We use the PCL implementation described in [20].

The way of applying ICP follows [15]; nonetheless, we propose performing the ICP registration between each point cloud from the source LiDAR, C_S^t, and the entire reconstruction from the target sensor, C_T^{full}, instead of using each of the instantaneous clouds C_T^t. In that way, the number of potential correspondences increases, and registration is made possible even in cases where the two LiDAR scanners do not share a common field of view.

In practice, ICP is carried out at each instant t using as inputs the clouds $C_T^{\text{full},t}$ and C_S^t, where $C_T^{\text{full},t}$ is the reconstructed target cloud C_T^{full} transformed in such a way that the origin of coordinates lies in the location represented by L_{LOAM}^t:

$$C_T^{\text{full},t} = \left((L_{\text{LOAM}}^{T/2})^{-1} \cdot L_{\text{LOAM}}^t \right)^{-1} \cdot C_T^{\text{full}} \tag{2}$$

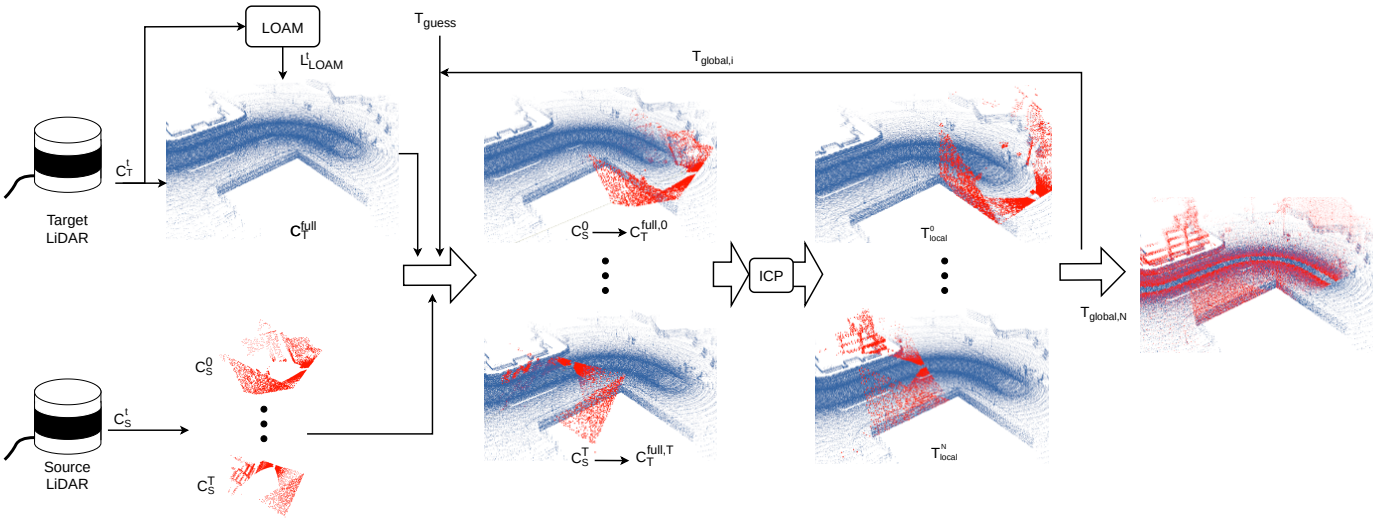


Fig. 2. Method overview. First, point clouds from the target LiDAR (in blue) are transformed and accumulated using the localization provided by the LOAM odometry. Later, an estimate of the transformation between LiDAR devices is found by using ICP to register each of the individual source LiDAR clouds (in red) to the accumulated target reconstruction. Then, the estimated transformation is obtained as the average of the transformations provided by ICP for each source cloud. Finally, this procedure is iteratively repeated using the output from each iteration as an initial guess for the next step, where the maximum correspondence distance (MCD) parameter of ICP is modified. The output from the last iteration can be used to accurately align the data from both sensors.

Hence, a local estimate T^t_{local} of the transformation between both LiDAR devices can be obtained through ICP:

$$T^t_{\text{local}} = \text{ICP}(C_T^{\text{full},t}, C_S^t, T_{\text{guess}}), \tag{3}$$

where T_{guess} is an initial guess of the transformation that must be provided as input along with both clouds.

Remarkably, the different executions of ICP corresponding to each time step (and, therefore, to each C_S^t cloud) can be parallelized, thus improving the method's efficiency.

However, in our preliminary experiments, we found that the accuracy of the final result provided by the ICP algorithm under these conditions was highly dependent on some configuration parameters. In particular, it proved closely related to the quality of the first estimation of the transformation, as well as to the value chosen as maximum correspondence distance (MCD), a parameter that controls the admissible distance threshold between two correspondent points. This parameter should be high when using rough initial guesses but lower with better-informed initializations. Because of this, in the next section, we introduce an iterative strategy to adapt the MCD value as available estimates improve.

C. Transformation Estimation

The procedure to estimate the transformation linking the LiDAR devices in space starts with the application of the ICP registration algorithm to pairs of clouds obtained over time, as described in the previous section. As a result, a set of local transformation estimates $\{T^t_{\text{local}}\}_{t=0}^{T}$ is obtained.

It should be noted that, as shown in (3), ICP requires an initial estimate of the aimed transformation, T_{guess}; in practice, this guess can be obtained by measuring by hand as it does not need to be very accurate. In our experiments, we allow errors up to $20\,\text{cm}$ and $15°$, values above the usual range when manually setting these parameters. A single T_{guess} is used as input in the computation of the different T^t_{local} transformations.

These local results are combined into an overall estimate T_{global} by averaging each of their components over time; that is, the component i of T_{global} is computed by:

$$T_{\text{global},i} = \overline{T^t_{\text{local, i}}} = \frac{1}{T} \cdot \sum_{t=0}^{T} T^t_{\text{local, i}} \tag{4}$$

Note that the transformations are composed of three components expressing the translation (x, y, z) and another four describing the rotation as a quaternion (q_0, q_1, q_2, q_3). Using the quaternion components instead of the Euler angles produces a better approximation to a rotation mean, as stated in [21].

To minimize the impact of the MCD parameter and improve the accuracy, this procedure is iteratively repeated, feeding each resulting T_{global} as T_{guess} for the next iteration. As each iteration improves the initial estimate, the MCD threshold value is reduced by a factor of γ. Empirically, we found that an initial value of $0.5\,\text{m}$ and $\gamma = 0.95$ yield satisfactory results.

The iterative process stops when the maximum number of iterations, set as a parameter N, is reached. The procedure's outcome is the result T_{global} obtained at the last iteration. Algorithm 1 provides a summary of the method.

IV. Experimental Results

The validation of the proposed method has been conducted in two different ways. On the one hand, experiments have been carried out in a virtual environment to obtain qualitative results with ground truth. On the other hand, the method has also been applied to a real vehicle with different LiDAR sensors, where the quality of the union between the two point clouds with the obtained calibration parameters has been qualitatively evaluated.

Algorithm 1: Proposed method.

input : $C_T^{\text{full},t}, C_S^t, T_{\text{guess}}, N, \delta, \gamma$
output: T_{cali}

$n \leftarrow 0$
$\Delta L \leftarrow 0$
$\text{MCD} \leftarrow 0.5$
while $n < N$ **do**
  $T_{\text{guess}-1} \leftarrow T_{\text{guess}}$
  for $t \leftarrow 0$ **to** T **do**
    $T_{\text{local}}^t \leftarrow \text{ICP}(C_T^{\text{full},t}, C_S^t, T_{\text{guess}}, \text{MCD})$
  end
  $T_{\text{guess}} \leftarrow \overline{T_{\text{local}}^t}$
  $n \leftarrow n + 1$
  $\text{MCD} \leftarrow \text{MCD} \times \gamma$
end
$T_{\text{cali}} \leftarrow T_{\text{guess}}$

A. Simulation Environment Setup

The evaluation of extrinsic calibration results is a non-trivial challenge, as it is impossible to measure the sensor positions with the necessary accuracy to obtain reliable ground-truth references. Most works in the literature merely compare the results with those obtained from manual measurements or perform a purely visual assessment.

In contrast, we propose using a simulation environment to evaluate our approach from a quantitative perspective. To that end, we use CARLA [22], a software specialized in the simulation of autonomous vehicles known for its realistic scenarios. CARLA can simulate different sensors, including LiDAR scanners, and allows tuning its parameters to adjust them according to a particular sensor model (e.g., number of layers, aperture, or noise).

In our experiments, the simulated sensor follows the specifications of Velodyne's HDL-32 model, which has an aperture of 40°, a standard deviation in distance measurements of 0.008 m, and a dropoff rate of 0.1. Data were obtained from recordings made in the simulator while the vehicle was driving through diverse areas of the virtual environment, featuring different setups (i.e., relative poses) of the LiDAR sensors. The collected data includes the point clouds from both sensors, their relative positions (i.e., the ground truth of the parameters being sought), the vehicle's location, and the timestamp at which each piece of data was collected.

The setups used in the experiments are shown in Table I. Configurations A-C are used in most of the experiments and correspond to the most usual positions used in autonomous vehicles, with one LiDAR (target) placed parallel to the ground plane and the other (source) sitting in positions similar to those used. It should be noted that, in these setups, the fields of view of both LiDAR scanners barely overlap; for example, configuration C is similar to the one shown in Fig. 1, where each LiDAR covers only 180° around one end of the vehicle.

On the other hand, configuration D is aimed to establish a fair comparison with a state-of-the-art alternative that requires field-of-view overlapping for its operation.

TABLE I
GROUND-TRUTH SOURCE LIDAR POSITION (METERS) AND ROTATION (DEGREES) REFERENCED TO THE TARGET LIDAR FOR THE THREE TESTED SCENARIOS.

Cfg.	X	Y	Z	Roll	Pitch	Yaw
A	1.0	0.0	0.4	0.0	40.0	0.0
B	0.0	0.5	0.2	0.0	45.0	90.0
C	5.0	0.0	0.0	0.0	0.0	180.0
D	0.2	1.0	0.4	10.0	0.0	0.0

B. Simulation Results

Experiments in the simulation environment mimic real situations where the vehicle equipped with the pair of LiDARs to be calibrated moves along a traffic scenario. The vehicle's position is known at all times. The calibration accuracy is assessed through the error between the output provided by the method and the ground-truth transformation in terms of both pure translation (distance) and rotation (angular error).

To remove the effect of external factors, we performed 50 different experiments for each setup, varying the vehicle's starting location in the virtual environment and the initial guess values, which were obtained by adding a random error to the calibration ground truth. This error reaches up to 20 cm in each translation component and 0.2 rad in each rotation component. When establishing comparisons, an identical seed was used for each method when generating the random errors, thus guaranteeing that the initial estimates were the same for each evaluated method.

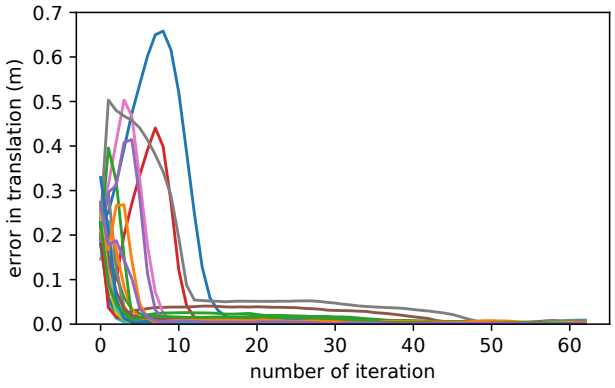
In each experiment, a total of 50 point clouds were processed for each sensor, collected every 0.5 s; therefore, the total duration of the sequence is 25 s, where the vehicle travels a distance of approximately 70 m through one of the city maps of the simulator.

First, to assess the proposed method and test the effectiveness of the iterative process, Fig. 3 shows the evolution of the calibration errors with the number of iterations for each of the experiments performed (i.e., different setups and different initializations). As can be seen, all tests converge to almost negligible errors, even in those rare cases in which peaks appear in early iterations.

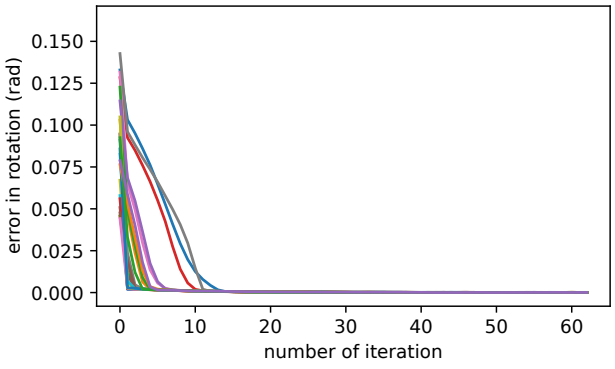
These results validate the iterative approach introduced in this work as opposed to the use of a single ICP iteration result, as proposed in [15]. As apparent, results for $N > 20$ consistently outperform the single-run alternative by a large margin.

The second set of experiments aims to evaluate how different factors affect the proposed method. Hence, the benefit of using a decay factor γ for the MCD parameter has been tested by performing the same experiments without decay (i.e., $\gamma = 1.0$). As shown in Fig. 4, the error without this factor increases since, in some cases, the algorithm does not converge to a result as accurate as the one obtained using the proposed value of $\gamma = 0.95$.

Another critical factor to be analyzed is the effect of localization accuracy, which will impact the target cloud accumulation and, therefore, the quality of the reconstruction C_T^{full}. Therefore, experiments to study the effect of the use of an imperfect source of odometry were also performed.



(a) Evolution of translation error



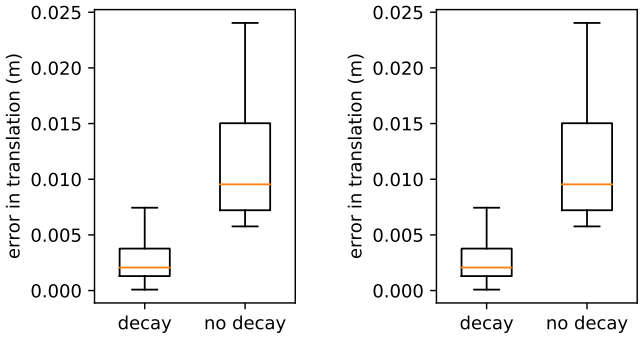
(b) Evolution of rotation error

Fig. 3. Evolution of the translation and rotation error with each iteration n of the proposed method. Each color corresponds to a different experiment (i.e., different setups and initializations).

Fig. 5 depicts the results of the proposed method using the ground-truth localization, a degraded localization generated by adding Gaussian noise to the ground-truth localization, and the actual LOAM odometry. As shown, although the introduction of odometry errors naturally deteriorates the accuracy of the result, the method is still robust to relatively large localization errors, and the effect of this factor becomes minimal when using the proposed LiDAR odometry method.

Next, a different set of experiments is devoted to benchmarking the performance of the proposed approach. Firstly, we compare our approach with a state-of-the-art extrinsic calibration method with an analogous set of features, MLCC [16]. Similar to our approach, MLCC is intended to calibrate LiDAR pairs with non-overlapping fields of view by performing cloud accumulation. However, it also includes additional procedures for pose optimization that are out of the scope of our work.

For fairness, our experiments feed the MLCC procedure with the pose estimation provided by LOAM, as is the case with our algorithm, and apply only the stages corresponding to the extrinsic parameter optimization and the final joint pose–extrinsics optimization. The values chosen for the parameters controlling the voxel size and the point cloud downsampling are identical to the ones used by the authors for the AVIA device. The results of the comparison are shown in Table II. They suggest that the accuracy of the parameters provided by our approach consistently surpasses the estimation by the two extrinsic parameter optimization steps from MLCC.



(a) Translation error. (b) Rotation error.

Fig. 4. Box plot comparison of the error made by the proposed method with and without the use of a decay factor.

TABLE II
TRANSLATION AND ROTATION ERROR COMPARISON BETWEEN THE PROPOSED METHOD AND THE EXTRINSIC PARAMETER OPTIMIZATION PROPOSED IN MLCC [16] USING CONFIGURATIONS A, B, AND C.

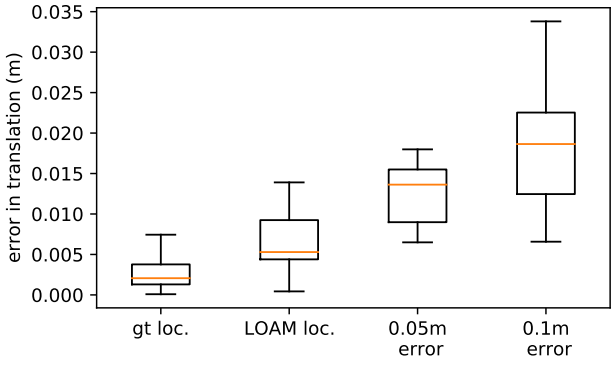
Cfg.	Method	Translation error (m)	Rotation error (rad)
A	MLCC (w/o. pose opt.)	0.05873	0.02510
	Ours	0.01431	0.00025
B	MLCC (w/o. pose opt.)	0.27862	0.03306
	Ours	0.00350	0.00052
C	MLCC (w/o. pose opt.)	0.09049	0.01622
	Ours	0.04891	0.00245

Lastly, the proposed method is compared with *velo2cam* [9], a state-of-the-art method that uses a fiducial pattern to perform an offline calibration procedure for any pair of sensors involving LiDARs or cameras. As stated before, configuration D is used in this comparison as the cited approach requires overlap between the sensors' fields of view. To perform this comparison, the calibration target used in that work was recreated in CARLA, and the same setup was employed to obtain results in the simulation environment with both approaches. Table III suggests that the accuracy of the results from both approaches is comparable, with the *velo2cam* approach achieving a lower translation error, whereas the proposed method yields a lower rotation error.

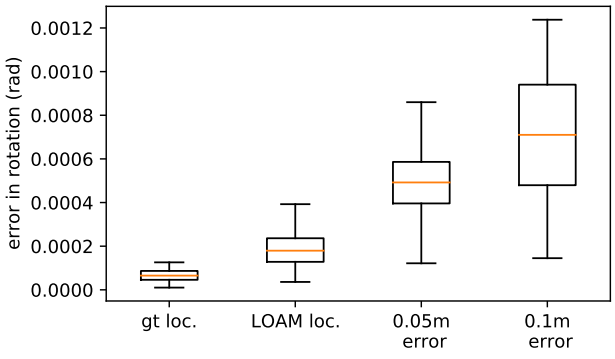
TABLE III
TRANSLATION AND ROTATION ERROR COMPARISON BETWEEN THE PROPOSED METHOD AND VELO2CAM USING CONFIGURATION D.

Method	Translation error (m)	Rotation error (rad)
Proposed method	0.00161	0.00012
velo2cam [9]	0.00017	0.00106

We note that, unlike the approach being compared, our method can perform the calibration while the vehicle moves without requiring a specific setup; furthermore, LiDAR scanners do not need to share a common field of view, which makes possible the calibration of configurations such as A, B, and C in Table I.



(a) Translation error



(b) Rotation error

Fig. 5. Box plot of the calibration errors for different sources of odometry: *gt loc.* represents the ground-truth localization, *LOAM loc.* the LOAM localization, and *0.05m error* and *0.1m error*, the ground-truth localization with additive Gaussian noise.

C. Real Environment Results

Results in the previous section quantitatively analyze the proposed method's performance in a simulation environment. However, for a complete assessment of the performance of the approach, we also carried out experiments in real environments. Although ground truth is unavailable in this case, the final result can still be qualitatively evaluated by analyzing the alignment between both point clouds when transformed into a common coordinate system using the calibration data.

To perform these experiments, two different platforms have been used. On the one hand, two LiDAR sensors were mounted in different positions on the autonomous vehicle platform shown in Fig. 6 and described in [3], [23]. On the other hand, we also obtained results from an automated minibus with a configuration similar to the one depicted in Fig. 1. In both cases, a 32-layer LiDAR was used as the source sensor, and a 16-layer LiDAR as the target sensor. In the first platform, the target device was tilted with respect to the source device, as shown in Fig. 6, whereas in the second platform, both LiDARs were aligned but placed at opposite ends of the bus. Both vehicles were naturalistically driven through urban environments while the data was being recorded. As with the simulation experiments, 50 clouds from each sensor, sampled with an interval of 0.5 s, were used.

The results of the subsequent calibration using these recordings are shown in Fig. 7, which shows the alignment of both clouds focusing on key points of the calibration, such as the ground, buildings' edges, or trees and streetlights. As can be seen, the accuracy of the calibration results enables the use of the combined information from both sensors to represent the scene, thus enhancing the performance of the perception algorithms downstream.

Fig. 6. Autonomous vehicle platform used to perform real experiments with LiDAR sensors attached.

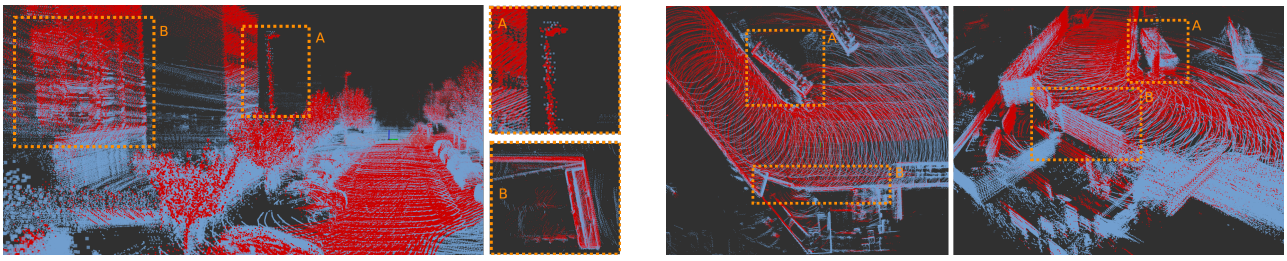
V. Conclusions

A new LiDAR sensor calibration method based on the iterative application of the ICP algorithm has been described. The results present accuracy levels compatible with the requirements of automotive applications, enabling proper sensor data alignment even at long distances.

The proposed method allows LiDAR calibration to be performed on a broad range of sensor setups, including those in which sensors do not share a common field of view, which is a feature that is not usual in comparable approaches. Another key advantage of the approach is its simplicity, as parameters can be obtained through a fast procedure that does not require specific calibration setups or fiducial markers. The scenario used to perform the calibration should ideally have elements that can serve as a reference, such as a flat road and several nearby objects or buildings that will help both the LOAM odometry and the ICP algorithm; nonetheless, these requirements can be easily fulfilled on a wide variety of typical traffic scenarios.

The experiments, both in simulated and real environments, have validated the method's usefulness. Furthermore, the source code of this algorithm has been publicly released as a Robot Operating System (ROS) package so that the community can replicate the results and apply the method to different use cases.

Future work will focus on the following lines: first, we will explore the possibilities of generalizing this method to work with different point-cloud-based sensors, such as radar or stereo cameras. On the other hand, we aim to determine the initial estimation of the sensors' poses automatically, thus removing the only manual intervention currently required in the procedure. Finally, optimizing the process will result in fewer requirements to execute the method, facilitating its use in all kinds of automotive platforms.



(a) (b)

Fig. 7. Real experiments results where the accumulated point clouds from both LiDAR sensors (each one in a different color) are transformed to a common frame. In 7(a), an overview is provided on the left side, and certain relevant details marked with A and B are cropped and zoomed in on the right side. In 7(b), two different views of the same scene are provided, with A and B pointing out details showing a good calibration in both views.

ACKNOWLEDGMENT

This work has been supported by the Madrid Government (Comunidad de Madrid-Spain) under the Multiannual Agreement with UC3M ("Fostering Young Doctors Research", APBI-CM-UC3M), and in the context of the V PRICIT (Research and Technological Innovation Regional Programme). Miguel Á. de Miguel acknowledges support by the "Ministerio de Universidades", the "Convocatoria de Ayudas para la recualificación del sistema universitario español para 2021-2023, de la Universidad Carlos III de Madrid, de 31 de Mayo de 2022", and Carlos Guindel acknowledges support by the "Ministerio de Universidades" and the "Convocatoria de la Universidad Carlos III de Madrid de Ayudas para la recualificación del sistema universitario español para 2021-2023, de 1 de julio de 2021, en base al Real Decreto 289/2021, de 20 de abril de 2021, por el que se regula la concesión directa de subvenciones a universidades públicas para la recualificación del sistema universitario español."

REFERENCES

[1] E. Marti, M. A. De Miguel, F. Garcia, and J. Perez, "A review of sensor technologies for perception in automated driving," *IEEE Intell. Transp. Syst. Mag.*, vol. 11, no. 4, pp. 94–108, 2019.

[2] J. Lambert, A. Carballo, A. M. Cano, P. Narksri, D. Wong, E. Takeuchi, and K. Takeda, "Performance analysis of 10 models of 3D LiDARs for automated driving," *IEEE Access*, vol. 8, pp. 131 699–131 722, 2020.

[3] M. Á. de Miguel, F. M. Moreno, F. García, J. M. Armingol, and R. E. Martin, "Autonomous vehicle architecture for high automation," in *Comput. Aided Syst. Theory – EUROCAST 2019*. Springer International Publishing, 2020, pp. 145–152.

[4] A. Peters, A. Schmidt, and A. C. Knoll, "Extrinsic calibration of an eye-in-hand 2D lidar sensor in unstructured environments using ICP," *IEEE Robot. Autom Lett.*, vol. 5, no. 2, pp. 929–936, 2020.

[5] N. Heide, T. Emter, and J. Petereit, "Calibration of multiple 3D LiDAR sensors to a common vehicle frame," in *Proc. 50th Int. Symp. Robot.*, 2018, pp. 372–379.

[6] S. Mishra, G. Pandey, and S. Saripalli, "Target-free extrinsic calibration of a 3D-lidar and an IMU," in *Proc. 2021 IEEE Int. Conf. Multisens. Fusion Integr. Intell. Syst.*, 2021.

[7] W. Maddern, A. Harrison, and P. Newman, "Lost in translation (and rotation): Rapid extrinsic calibration for 2D and 3D LIDARs," in *Proc. 2012 IEEE Int. Conf. Robot. Autom.*, 2012, pp. 3096–3102.

[8] C. Guindel, J. Beltrán, D. Martín, and F. García, "Automatic extrinsic calibration for lidar-stereo vehicle sensor setups," in *Proc. 2017 IEEE 20th Int. Conf. Intell. Transp. Syst.*, 2017, pp. 674–679.

[9] 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. 17 677–17 689, 2022.

[10] W. Wang, S. Nobuhara, R. Nakamura, and K. Sakurada, "SOIC: Semantic online initialization and calibration for LiDAR and camera," *arXiv preprint arXiv:2003.04260*, 2020.

[11] Y. Zhu, C. Li, and Y. Zhang, "Online camera-LiDAR calibration with sensor semantic information," in *Proc. 2020 IEEE Int. Conf. Robot. Autom.*, 2020, pp. 4970–4976.

[12] B. Shi, P. Yu, M. Yang, C. Wang, Y. Bai, and F. Yang, "Extrinsic calibration of dual LiDARs based on plane features and uncertainty analysis," *IEEE Sens. J.*, vol. 21, no. 9, pp. 11 117–11 130, 2021.

[13] B. Xue, J. Jiao, Y. Zhu, L. Zhen, D. Han, M. Liu, and R. Fan, "Automatic calibration of dual-LiDARs using two poles stickered with retro-reflective tape," in *Proc. 2019 IEEE Int. Conf. Imag. Syst. Tech.*, 2019.

[14] C. Gao and J. R. Spletzer, "On-line calibration of multiple LIDARs on a mobile vehicle platform," in *Proc. 2010 IEEE Int. Conf. Robot. Autom.*, 2010, pp. 279–284.

[15] Z. Gong, C. Wen, C. Wang, and J. Li, "A target-free automatic self-calibration approach for multibeam laser scanners," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 1, pp. 238–240, 2017.

[16] X. Liu, C. Yuan, and F. Zhang, "Targetless extrinsic calibration of multiple small FoV LiDARs and cameras using adaptive voxelization," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–12, 2022.

[17] J. Zhang and S. Singh, "LOAM: Lidar odometry and mapping in real-time." in *Robot.: Sci. Syst.*, vol. 2, no. 9, 2014.

[18] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in *Proc. 2018 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 4758–4765.

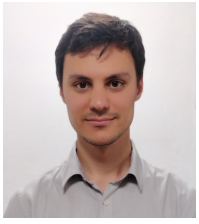
[19] P. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, 1992.

[20] C.-T. Hsieh, "An efficient development of 3D surface registration by Point Cloud Library (PCL)," in *Proc. 2012 Int. Symp. Intell. Signal Process. Commun. Syst.*, 2012, pp. 729–734.

[21] C. Gramkow, "On averaging rotations," *J. Math. Imag. Vis.*, vol. 15, no. 1, pp. 7–16, 2001.

[22] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. 1st Annu. Conf. Robot Learn.*, 2017, pp. 1–16.

[23] M. A. de Miguel, F. M. Moreno, P. Marin-Plaza, A. Al-Kaff, M. Palos, D. Martin, R. Encinar-Martin, and F. Garcia, "A research platform for autonomous vehicles technologies research in the insurance sector," *Appl. Sci.*, vol. 10, no. 16, p. 5655, 2020.



Miguel Ángel de Miguel received his Ph.D. degree in Electrical, Electronic and Automatic Engineering from Universidad Carlos III de Madrid in 2022 where he works as a post-doctoral researcher. His research interests focus on the areas of localization, path planning and sensors fusion applied for autonomous vehicles.

Carlos Guindel received his Ph.D. degree in Electrical Engineering, Electronics, and Automation in 2019 from the University Carlos III of Madrid, Spain, where he is currently a postdoctoral researcher. In 2022, he was visiting the Intelligent Vehicles and Traffic Technologies (INVETT) group from the University of Alcalá. His research interest focuses on the application of deep learning techniques to the intelligent transportation systems field, covering topics such as object detection, pose estimation, and sensor fusion.

Abdulla Al-Kaff (Member, IEEE) Assistant Professor and the Vice-Head of the Autonomous Mobility and Perception Lab (AMPL) at the Department of Systems Engineering and Automation, Universidad Carlos III de Madrid, and visiting professor at The German International University in Cairo. He graduated from the University of Aden in Computer Science and Engineering in 2007 and obtained his master's degree in Robotics and Automation in 2012 from Universidad Carlos III de Madrid, where he obtained his Ph.D. degree in Electrical Engineering, Electronics, and Automation in 2017. Dr. Abdulla was a recipient of the best Ph.D. thesis award by the IEEE Intelligent Transportation Systems Society - Spanish Chapter. He was the leader of the team that won the 2nd prize in the Academia International Category in the 2019 Dubai World Challenge for Self-Driving Transport. Dr. Abdulla is actively contributing to the scientific community of both intelligent transportation systems and robotics. He has authored or co-authored more than 60 papers in international journals and at international conferences, respectively. His research interests include Artificial Intelligence, Deep Learning, Reinforcement Learning, Data fusion, and image processing for autonomous ground and aerial vehicles. Dr. Abdulla is also actively working in the Intelligent Transportation Systems Society, where currently he is the secretary of the ITSS-Spanish Chapter, and as Associate Editor for the IEEE Open Journal of Intelligent Transportation Systems, and he is a guest editor for three special issues in prestigious journals.

Fernando García received his Ph.D. degree in Electrical, Electronic and Automatic Engineering from Universidad Carlos III de Madrid in 2012 where he works as Associate Professor. His research interests are perception and data fusion, mainly applied to vehicles and robotics. He is member of the Board of governors of the IEEE-ITS Society since 2017 and chair of the Spanish chapter for the period 2019-2020. He was recipient of IEEE ITS Young Researcher/Engineer Award 2021.