remote sensing

MDPI

Article

Online Extrinsic Calibration on LiDAR-Camera System with LiDAR Intensity Attention and Structural Consistency Loss

Pei An [1,2,†], Yingshuo Gao [2,†], Liheng Wang [1], Yanfei Chen [1] and Jie Ma [2,*]

[1] School of Electrical and Information Engineering, Wuhan Institute of Technology, Wuhan 430072, China; anpei@hust.edu.cn (P.A.); wlhhust@wit.edu.cn (L.W.); 05090408@wit.edu.cn (Y.C.)
[2] School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430072, China; yingshuogao@hust.edu.cn
* Correspondence: majie@hust.edu.cn
† These authors contributed equally to this work.

Abstract: Extrinsic calibration on a LiDAR-camera system is an essential task for the advanced perception application for the intelligent vehicle. In the offline situation, a calibration object based method can estimate the extrinsic parameters in high precision. However, during the long time application of LiDAR-camera system in the actual scenario, the relative pose of LiDAR and camera has small and accumulated drift, so that the offline calibration result is not accurate. To correct the extrinsic parameter conveniently, we present a deep learning based online extrinsic calibration method in this paper. From Lambertian reflection model, it is found that an object with higher LiDAR intensity has the higher possibility to have salient RGB features. Based on this fact, we present a LiDAR intensity attention based backbone network (LIA-Net) to extract the significant co-observed calibration features from LiDAR data and RGB image. In the later stage of training, the loss of extrinsic parameters changes slowly, causing the risk of vanishing gradient and limiting the training efficiency. To deal with this issue, we present the structural consistency (SC) loss to minimize the difference between projected LiDAR image (i.e., LiDAR depth image, LiDAR intensity image) and its ground truth (GT) LiDAR image. It aims to accurately align the LiDAR point and RGB pixel. With LIA-Net and SC loss, we present the convolution neural network (CNN) based calibration network LIA-SC-Net. Comparison experiments on a KITTI dataset demonstrate that LIA-SC-Net has achieved more accurate calibration results than state-of-the-art learning based methods. The proposed method has both accurate and real-time performance. Ablation studies also show the effectiveness of proposed modules.

Keywords: LiDAR-camera system; extrinsic calibration; mutual information; deep learning

Citation: An, P.; Gao, Y.; Wang, L.; Chen, Y.; Ma, J. Online Extrinsic Calibration on LiDAR-Camera System with LiDAR Intensity Attention and Structural Consistency Loss. *Remote Sens.* **2022**, *14*, 2525. https://doi.org/10.3390/rs14112525

Academic Editors: Stephan Nebiker, Pierre Grussenmeyer, Tania Landes and Grazia Tucc

Received: 19 April 2022
Accepted: 23 May 2022
Published: 25 May 2022

1. Introduction

The system that consists of Light Detection and Ranging (LiDAR) and optical camera has played an important role in the recent years [1]. This system is called a LiDAR-camera system, which is the core multi-sensor system for the intelligent vehicle [2]. The main advantage of LiDAR-camera system is that it provides both structural (3D LiDAR point cloud) and textural (2D RGB image) information of the surrounding [3], which can be used to improve the performance of the advanced perception tasks (i.e., 3D object detection [4], 3D point cloud semantic segmentation [5]) for the autonomous driving car.

To utilize LiDAR point cloud and RGB image efficiently [6], it is essential to find the alignment of point p in LiDAR point cloud and its corresponding pixel I in RGB images, to make sure that p and I satisfy the projection constraint [7]. Theoretically, this alignment problem can be solved by point cloud and image registration [8]. Considering the background of autonomous driving, the mechanical structure of LiDAR and camera is fixed. In the ideal condition, the relative pose of LiDAR and camera is also constant. It means that point cloud and image registration can be simplified and solved via calibrating the

LiDAR-camera system. This task aims to calibrate all the parameters of the LiDAR-camera system, such as (i) the intrinsic parameters of camera, (ii) the intrinsic parameters of LiDAR, and (iii) the extrinsic parameters of LiDAR-camera system. At first, the intrinsic parameters of camera consist of intrinsic matrix $\mathbf{K}$ and lens distortion parameters $\mathbf{D}$. In the offline situation, using the rectangular board with the chessboard patterns as a calibration object, the user can place the camera at the different positions to observe the calibration object, and exploit the nonlinear optimization to estimate $\mathbf{K}$ and $\mathbf{D}$ via minimizing the reprojection errors of observed corner points extracted from the chessboard patterns [7]. $\mathbf{K}$ and $\mathbf{D}$ of the camera are basically constant during the online application. Second, the intrinsic parameters of LiDAR are more complex than the camera because they are related with the mechanical structure of LiDAR. These parameters are generally carefully calibrated by the LiDAR manufacturer. The intrinsic parameters of LiDAR are also constant in the actual application. Third, the extrinsic parameters of LiDAR-camera system are the rigid transformation of the LiDAR coordinate system and the camera coordinate system, represented by the rotation matrix $\mathbf{R}$ and the translation vector T. Thanks to the rapid development of computer vision and LiDAR manufacturing technique, the intrinsic parameters of LiDAR and camera can be calibrated with high precision. However, due to the limited resolution angle of LiDAR, the LiDAR point cloud is sparse, causing the extrinsic parameter calibration on the LiDAR-camera system to be a challenging problem, for the co-observed feature is difficult to find from the sparse LiDAR point cloud and dense RGB image [9,10]. Therefore, we mainly discuss the extrinsic calibration on the LiDAR-camera system in this paper.

The key of extrinsic calibration on LiDAR-camera system is to find the co-observed constraints from both LiDAR point cloud and RGB image [11]. To resist the sparsity of LiDAR point cloud, a simple and effective approach is to design a specific target or calibration object [12]. The calibration object has a salient structure, and its contour and edge features can be clearly found in the sparse LiDAR point cloud and RGB image. These features serve as the co-observed constraints for calibration. This kind of method is called the calibration object based calibration method. The man-made calibration object is convenient to find [13] in the offline case. However, in the online situation (i.e., the intelligent vehicle drives in the open scene), there is nearly no specific calibration object so that the calibration object based extrinsic calibration method cannot work. Due to the complexity of object contour and the sparsity of LiDAR point cloud, it is a challenging task to find the co-observed constraint in the open scene [10]. Therefore, it is difficult to calibrate the extrinsic parameters of LiDAR-camera system in the online situation.

With the background of intelligent driving, one may wonder whether it is necessary to calibrate the LiDAR-camera system in the online situation. Ref. [14,15] points out that, in the long time application of LiDAR-camera system, the extrinsic parameter has small but accumulated drift (represented by $\Delta\mathbf{R}$ and ΔT) due to the mechanical vibrations or temperature changes. This issue causes extrinsic calibration results $\mathbf{R}_0$ and T_0 in the offline case to not be accurate at this time. Online extrinsic calibration aims to estimate $\Delta\mathbf{R}$ and ΔT, and obtain the accurate extrinsic parameters $\mathbf{R}$ and T as:

$$\begin{pmatrix} \mathbf{R} & T \\ 0^T & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R}_0 & T_0 \\ 0^T & 1 \end{pmatrix} \cdot \begin{pmatrix} \Delta\mathbf{R} & \Delta T \\ 0^T & 1 \end{pmatrix} \tag{1}$$

In this paper, we present a novel convolution neural network (CNN) based calibration network with a LiDAR intensity attention based backbone network (LIA-Net) and structural consistency (SC) loss. The proposed calibration network is called LIA-SC-Net, which aims to solve the problem of online extrinsic calibration on the LiDAR-camera system. From the Lambertian reflection model [16], it is found that the object with high LiDAR intensity has the higher possibility with the salient texture feature in the RGB image. This kind of object has the potential co-observed constraint. As the object in the open scene has complex contour, it is difficult to extract the robust primary co-observed features [17]. We exploit the technique of deep learning [18], and attempt to learn the co-observed calibration (CoC) feature via the deep CNN architecture. As the object with the higher LiDAR intensity has

more potential information for calibration, we refer the visual attention mechanism [19] in deep learning, and then propose LIA-Net to learn the salient CoC feature. In the later stage of training, the regression loss of extrinsic parameters tends to be stable and close to zero. On the one hand, it has the risk of vanishing gradient [20], which limits the efficiency of supervised learning. On the other hand, according to the camera pinhole model [7], even if the rotation error is smaller than 1 deg, the alignment pixel error of the projected LiDAR point and its corresponding pixel in the RGB image might be larger than 10.0 pixels. To deal with this issue, we present SC loss to reduce the alignment pixel error by minimizing the difference between a projected LiDAR image (i.e., LiDAR depth image, LiDAR intensity image) and its GT projected LiDAR image. With combining the advantages of LIA-Net and SC loss, we then propose LIA-SC-Net to estimate the extrinsic parameters. To evaluate the performance of the proposed method, extensive experiments are both conducted in the public dataset KITTI [2] collected by the self-made LiDAR-camera system. Ablation studies are also provided to show the effectiveness of LIA-Net and SC loss. We believe that the proposed method benefits the advanced application of the LiDAR-camera system.

In conclusion, three main contributions are provided in this paper:

(i) Considering that the object with higher LiDAR intensity has more salient co-observed constraint, LIA-Net is proposed to utilize LiDAR intensity as the attention feature map to generate the salient CoC features;
(ii) To prevent the risk of vanishing gradient in the later training stage, SC loss is presented to approximately reduce the alignment error from LiDAR point cloud and RGB image by minimizing the difference of the projected and its GT projected LiDAR images;
(iii) Taking the advantages of both LIA-Net and SC loss, deep learning based extrinsic calibration method LIA-SC-Net is presented to estimate the accurate extrinsic parameters of a LiDAR-camera system.

The remainder of this paper is organized as follows: At first, related works of extrinsic calibration on LiDAR-camera system are provided in Section 2. In the next, the proposed online extrinsic calibration method is illustrated in Section 3. After that, experimental results in the public dataset are shown in Section 4. Discussions of the proposed method and its performance in the actual application are presented in Section 5. Finally, our work is concluded in Section 6.

2. Related Works

As for the calibration technique, current extrinsic calibration methods can be classified as three categories: (i) calibration object based extrinsic calibration method, (ii) information fusion based extrinsic calibration method, and (iii) deep learning based extrinsic calibration method.

2.1. Calibration Object Based Extrinsic Calibration Method

The key of calibration object based method is to design the specific target with the significant structure. The corner point of the calibration object can be measured from the LiDAR point cloud and RGB image, and 3D-2D, 3D-3D point corresponding constraints can be established from the corner point [9]. From a 3D-2D point corresponding constraint, perspective-n-point (PnP) algorithm [21,22], direct linear transformation (DLT) method [23], and bundle adjustment (BA) algorithm [24] can be used to estimate $\mathbf{R}$ and T. From a 3D-3D point corresponding constraint, the iterative closest point (ICP) method [25,26] can be exploited to calibrate the extrinsic parameters. A common calibration object can be classified as three groups: (i) 2D calibration object; (ii) 3D calibration object; and (iii) combined calibration object. 2D calibration objects are the polygonal planar board [11,12], the planar board with circular holes [27], or chessboard patterns [28,29]. Chessboard patterns are useful in the extrinsic calibration on the LiDAR-camera system, for the corner points of chessboard patters are easily extracted from RGB image [28]. As LiDAR intensity values of chessboard patterns change regularly, this fact can be utilized to extract the corner points of chessboard patters from LiDAR point cloud [28,30]. If the relative position

of chessboard patterns and the planar board is known, the corner points of chessboard patterns can also provide 3D-2D point corresponding constraints [31]. 3D calibration objects are the trihedron [32], the spherical ball [33], the cubic box [13,34], and the cubic box with patterns [35]. Corner point of the trihedron and cube, or center point of the sphere can be used to establish the point corresponding constraint. To extract more calibration constraints, researchers try to combine the 2D and 3D calibration objects, and propose the combined calibration object. An et al. [9] proposed a combined calibration object that consists of the main and auxiliary calibration objects. Kummerle and Kuhner [36] combined the spherical ball and the planar board with patterns. As time goes by, the structure of calibration object tends to be complex. In fact, from the viewpoint of actual application, it is an interesting question to balance the calibration accuracy and the production cost of the calibration object.

2.2. Information Fusion Based Extrinsic Calibration Method

As discussed in Section 1, under the background of the intelligent vehicle, it is essential to calibrate the LiDAR-camera system in the target-less open scene [14,15]. Before the rise of deep learning in the area of computer vision, the information fusion based extrinsic calibration method is presented to automatically find the co-observed calibration features from camera image and LiDAR data. This so called co-observed calibration feature can be (i) corresponding edge feature [14], (ii) corresponding point cloud [37], and (iii) corresponding odometry [38]. To make use of the corresponding edge feature, researchers converts the extrinsic calibration as the multi-source image registration problem [14]. With the initial extrinsic parameters $\mathbf{R}_0$ and T_0, the LiDAR point cloud can be projected into the image plane as the projected LiDAR depth image $\mathcal{D}_0$ and the projected LiDAR intensity image $\mathcal{I}_0$. $\Delta\mathbf{R}$ and ΔT can be estimated via solving the image registration of $\mathcal{D}_0$ or $\mathcal{I}_0$ and RGB image $\mathcal{R}$. Mutual information (MI) is a useful tool to measure the similarity of two multi-source images, which has been successfully used in the field of remote sensing [39]. With the MI based calibration loss, Pandey et al. [14] established the nonlinear optimization problem to estimate $\Delta\mathbf{R}$ and ΔT in an iterative scheme. To obtain the robust registration result, Wolcott and Eustice [40] exploited the normalized MI for registration, and Irie et al. [41] proposed bagged least-squares MI (BLSMI) as the calibration loss. Zhu et al. [42] considered that the edge feature in the RGB image is sensitive to color variations and noise, and they used the semantic segmentation map predicted from RGB image for registration. Xiao et al. [43] obtained $\Delta\mathbf{R}$ and ΔT by minimizing the image feature errors of LiDAR points in the different frames. To make use of the corresponding point cloud, researchers convert the extrinsic calibration as the 3D point cloud registration problem [37]. The key of this approach is to generate the point cloud $\mathcal{P}_c$ from RGB images. After that, $\mathbf{R}$ and T are estimated by solving the point cloud registration of $\mathcal{P}_c$ and LiDAR point cloud $\mathcal{P}_l$. For the LiDAR-stereo system, the stereo camera can generate the depth image via the disparity principle, and $\mathcal{P}_c$ is obtained via back-projecting the depth image into the 3D space [44]. For the common LiDAR-camera system, using the sequence of RGB images, a 3D point cloud $\mathcal{P}_c$ can be reconstructed via the technique of structure from motion (SFM) [45] and simultaneous localization and mapping (SLAM) [46]. Nagy and Benedek [3] exploited the semantic segmentation on the point clouds $\mathcal{P}_c$ and $\mathcal{P}_l$ to increase the accuracy of the ICP algorithm [25]. To make use of the corresponding odometry, researchers convert the extrinsic calibration as the classic hand-eye calibration problem [47]. This approach needs to move the LiDAR-camera system. With the ICP algorithm [25], the odometry of LiDAR is estimated as $\mathcal{O}_l$. Using the techniques of visual odometry (VO) [48], SFM, and SLAM, the odometry of camera is estimated as $\mathcal{O}_c$. A hand-eye calibration constraint can be established from $\mathcal{O}_l$ and $\mathcal{O}_c$, and then extrinsic parameters are estimated [49]. Ishikawa et al. [38] presented the two-stage odometry based extrinsic calibration method. In the first stage, they obtain the initial results $\mathbf{R}_0$ and T_0 from the odometry information via the hand-eye calibration approach [47]. In the second stage, they exploited the MI based extrinsic method [14] to obtain $\Delta\mathbf{R}$ and ΔT. The extrinsic parameters are finally estimated

via Equation (1). Unlike calibration object based extrinsic calibration method, the information fusion based extrinsic method tends to convert the extrinsic calibration as some known and classical problems, such as multi-source image registration [14], point cloud registration [37], and hand-eye calibration [38].

2.3. Deep Learning Based Extrinsic Calibration Method

As for the information fusion based extrinsic calibration method, it needs the extra techniques, such as MI computation [39], 3D reconstruction [45,46], and odometry estimation [25,48]. Calibration procedure is more complex than the calibration object based method, and calibration precision depends on the performance of these techniques. With the success of deep learning in the field of robotic vision [50], some researchers attempt to design the deep learning based extrinsic calibration method, and aims to extract the adaptive co-observed feature to regress the extrinsic parameters [15]. The key problem of this method is how to align the structure feature from LiDAR point cloud $\mathcal{P}_l$ and RGB image $\mathcal{R}$. As $\mathcal{P}_l$ and $\mathcal{R}$ have the different data representations, a naive approach is to project $\mathcal{P}_l$ into the image plane as $\mathcal{D}_0$ and $\mathcal{I}_0$ with $\mathbf{R}_0$ and T_0, and extract the co-observed feature from $\mathcal{D}_0$, $\mathcal{I}_0$, and $\mathcal{R}$. Most of the researchers select to extract the CoC feature from $\mathcal{D}_0$ and $\mathcal{R}$ [51–53] because most of the edge features are caused by the discontinuous of depth [54], which means that $\mathcal{D}_0$ and $\mathcal{R}$ exist the common structure feature. RegNet [51] is the first deep learning based calibration approach. It uses the network in network (NIN) module [55] to extract CoC features from $\mathcal{D}_0$ and $\mathcal{R}$, and exploit fully connected (FC) layers to regress $\Delta\mathbf{R}$ and ΔT. Iyer et al. [52] proposed a calibration network CalibNet, in which ResNet [56] is used as a backbone network to extract the feature maps from $\mathcal{D}_0$ and $\mathcal{R}$. In CalibNet, they exploited the geometric and photometric consistency of the $\mathcal{P}_l$ and $\mathcal{R}$ as the supervision loss. Based on the deep learning based optical flow [57], Cattaneo et al. [53] aimed to extract the CoC feature from the difference of $\mathcal{D}_0$ and $\mathcal{R}$. Recently, Wu et al. [15] proposed a coarse-to-fine method CalibRank where the image retrieval algorithm is used to estimate $\mathbf{R}_0$ and T_0. Yuan et al. [58] considered Riemannian geometry and present a module tolerance regularizer for extrinsic calibration. Ye et al. [59] designed a point weighting layer to predict the sparse keypoint correspondences with the weighted scores, and exploited the PnP algorithm [21] to estimate $\Delta\mathbf{R}$, ΔT with these correspondences. A deep learning based extrinsic calibration method is in development now. It is a challenging task to increase the generalization ability for various online calibration scenes. In addition, more discussion of this kind of method is presented in the following.

2.4. Analysis of the Deep Learning Based Extrinsic Calibration Method

As for the online extrinsic calibration on LiDAR-camera system, compared with the information fusion based method, the deep learning based method is the end-to-end approach. It does not require any extra techniques, and is convenient to use in the actual application. The general pipeline of deep learning based extrinsic calibration method is shown in Figure 1 [51–53]. It consists of three modules: (i) backbone networks, (ii) co-observed feature extraction, and (iii) parameters regression. To increase the calibration accuracy, two issues need to be focused on. First, how to learn the salient CoC feature from multi-source images is an important problem. To solve this issue, we exploit the visual attention mechanism from LiDAR intensity and present a novel backbone network LIA-Net. Second, it is essential to increase the learning efficiency of calibration network. To deal with it, SC loss is proposed to accurately align the LiDAR points and RGB pixels. In the next section, the proposed LIA-Net and SC loss are discussed with details.

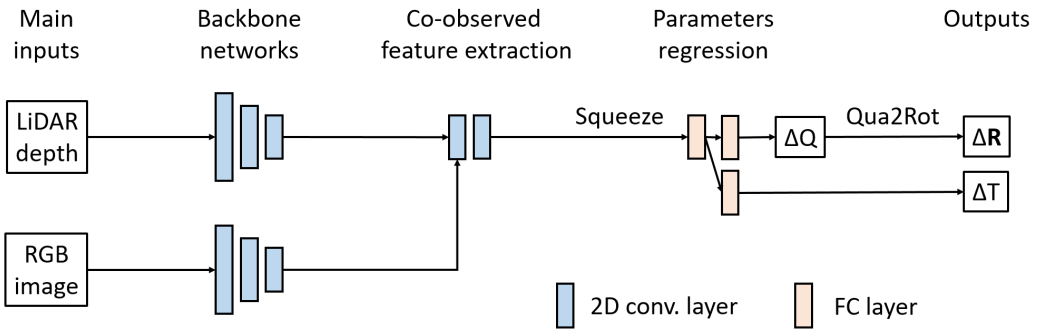

Figure 1. The general pipeline of deep learning based extrinsic calibration method. Inputs are $\mathcal{D}_0$ and $\mathcal{R}$. $\mathcal{D}_0$ are obtained by projecting $\mathcal{P}_l$ into the image plane with $\mathbf{K}$, $\mathbf{R}_0$, and T_0. Outputs are $\Delta\mathbf{R}$ and ΔT. Squeeze is the operation to convert the 2D feature map into a 1D feature vector. Function Qua2Rot converts the quaternion ΔQ as the rotation matrix $\Delta\mathbf{R}$. Conv. and FC mean convolution and fully connected, respectively.

3. Proposed Method

3.1. Problem Statement and Method Overview

Extrinsic calibration on the LiDAR-camera system aims to estimate the rigid transformation between LiDAR coordinate system $O_L - X_L Y_L Z_L$ and the camera coordinate system $O_C - X_C Y_C Z_C$, represented by $\mathbf{R}$ and T, shown in Figure 2a. As illustrated in Section 1, with the known and initial $\mathbf{R}_0$ and T_0, online extrinsic calibration aims to estimate $\Delta\mathbf{R}$ and ΔT. Compared with Figure 2b and Figure 2c, the alignment of LiDAR point cloud and RGB image is more accurate after online extrinsic calibration.

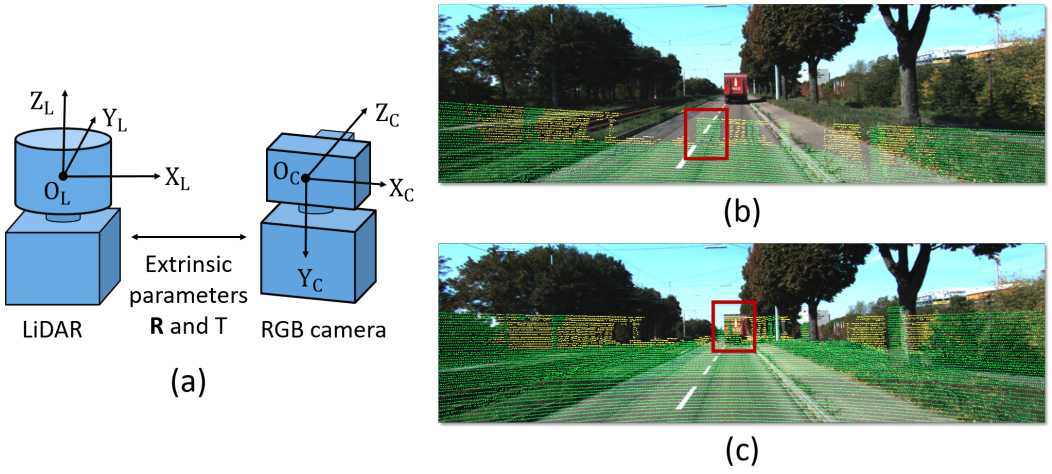

Figure 2. The task and significance of online extrinsic calibration on LiDAR-camera system. (**a**) the model of LiDAR-camera system; (**b**) inaccurate alignment of LiDAR point cloud and RGB image with $\mathbf{R}_0$ and T_0; (**c**) LIA-SC-Net predicts accurate $\Delta\mathbf{R}$ and ΔT, thus obtaining the precise alignment result. The projected LiDAR point cloud with pseudo color dependent on LiDAR intensity is used for visualization. After calibration, RGB pixels and LiDAR points of the vehicle in the red box are correctly aligned.

In this paper, we propose an efficient deep learning based online extrinsic calibration method LIA-SC-Net. The pipeline is presented in Figure 3. Its inputs are LiDAR point cloud $\mathcal{P}_l$, RGB image $\mathcal{R}$, and initial extrinsic parameters $\mathbf{R}_0$ and T_0. Its outputs are $\Delta\mathbf{R}$ and ΔT. Using the pinhole projection [7], the initial LiDAR depth $\mathcal{D}_0$ and LiDAR intensity $\mathcal{I}_0$ are obtained with $\mathcal{P}_l$, $\mathbf{K}$, and $\mathbf{R}_0$ and T_0. Projection detail is shown in Appendix A.1. $\mathcal{R}$, $\mathcal{D}_0$, and $\mathcal{I}_0$ are $[W, H, 3]$, $[W, H, 1]$, $[W, H, 1]$ tensors where W, H are the width and height of the image. The architecture of LIA-SC-Net is the extension of the general pipeline in Figure 1.

Unlike the common pipeline, LIA-Net in the proposed method utilizes the natural attention mechanism from LiDAR intensity, and SC-loss focuses on the alignment of LiDAR point cloud and RGB image, thus increasing the extrinsic calibration accuracy.

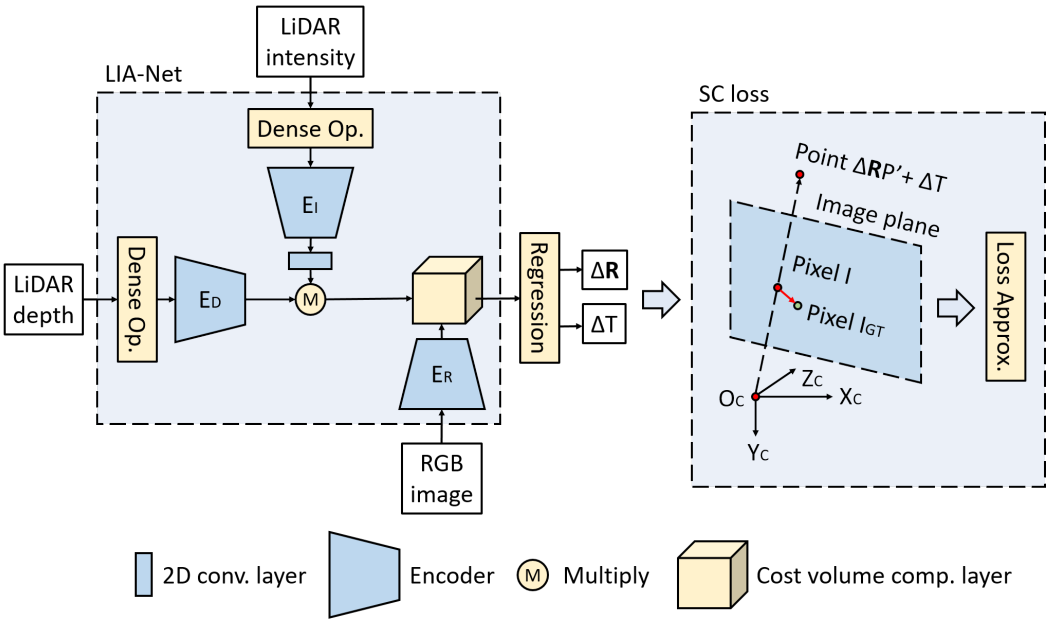

Figure 3. The pipeline of LIA-SC-Net. It mainly consists of LIA-Net and SC loss. Abbreviations Op. and approx. are operation and approximation, respectively. SC loss aims to reduce the alignment error by minimizing the re-projection error of the point $P' = \mathbf{R}_0 P + T_0$.

3.2. LiDAR Intensity Attention Based Backbone Network

LIA-Net is presented to generate salient CoC feature $\mathcal{F}_{CoC}$ from $\mathcal{R}$, $\mathcal{D}_0$, and $\mathcal{I}_0$. It has two procedures: (i) dense operation and (ii) CoC feature generation. At first, dense operation is discussed. It is a fact that the number of point cloud generated by the mechanical LiDAR ($\approx 10^5$) is far smaller than image pixels $W \cdot H$ ($\approx 10^6$), so that $\mathcal{D}_0$ and $\mathcal{I}_0$ are sparser than $\mathcal{R}$ (seen in Figure 4a,b). From the computation theory of CNN layer [18,56], sparse feature map has the risk of destroying the quality of the structural information of the original data. Thus, it is essential to densify $\mathcal{D}_0$ and $\mathcal{I}_0$. Inspired by the work [60], based on the classical image processing techniques, a simple and fast pipeline is presented as:

$$\mathcal{X}_d = \text{inv}(F_{\text{median}}(F_{\text{hole}}(F_{\text{dilate}}(\text{inv}(\mathcal{X}), k)), k)), \mathcal{X} = \{\mathcal{D}_0, \mathcal{I}_0\}, \mathcal{X}_d = \{\mathcal{D}_{0d}, \mathcal{I}_{0d}\} \quad (2)$$

$$\text{inv}(\mathcal{X}) = \max(\mathcal{X}) \cdot \mathbf{1} - \mathcal{X} \quad (3)$$

Operators $F_{\text{median}}(\cdot, k)$ and $F_{\text{dilate}}(\cdot, k)$ are median filter and dilation operation with the kernel size of $k \times k$. If the k set is too large, densification results $\mathcal{D}_{0d}$ and $\mathcal{I}_{0d}$ have distortion and are unreal. An exact estimation of k requires the intrinsic parameters of both LiDAR and camera. It is discussed in Appendix A.2. $F_{\text{hole}}(\cdot)$ is the operation to estimate the empty depth using the dilation with the large kernel size [60]. $\text{inv}(\cdot)$ is the inverse operation for the single channel image. $\mathbf{1}$ in Equation (3) is a full-one tensor with the same size of $\mathcal{X}$. Results of the dense operation are shown in Figure 4c.

After that, CoC feature generation is discussed. From the Lambertian reflection model [16], it is found that the object with higher LiDAR intensity has the potential with the salient structure feature in both LiDAR point cloud and RGB image, helpful for generating the significant CoC feature. It is discussed in the following. As most of the object surfaces in the open scene are rough [61], the Lambertian reflection model [16] is used to characterize LiDAR intensity r_l:

$$r_l \propto \rho_l \cdot \frac{n_l^T n_{\text{LiDAR}}}{{D_l}^2} \cdot P_{\text{LiDAR}} \quad (4)$$

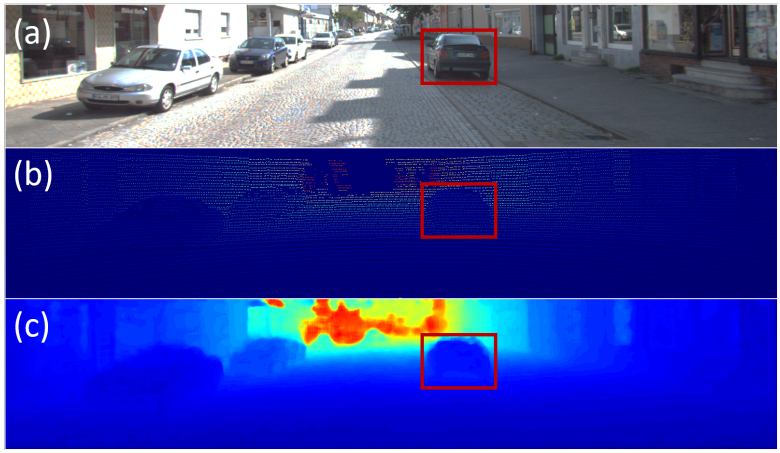

Figure 4. Visualization of dense operation. (**a**) RGB image; (**b**) LiDAR depth and intensity; (**c**) dense LiDAR depth and intensity after dense operation. Edge structure of the vehicle inside the red box in the dense LiDAR feature map is clearer than the raw LiDAR feature map, benefiting the extraction of the significant calibration feature.

LiDAR intensity is determined by the material coefficient of laser ρ_l, object orientation n_l, LiDAR orientation n_{LiDAR}, object distance D_l, and laser power P_{LiDAR}. As laser is the active light source, $P_{\text{LiDAR}} \propto D_l^{-2}$. If LiDAR intensity of one object is larger, it might be the fact that $n_l^T n_{\text{LiDAR}} \approx 1$ and D_l is smaller, which means that this object is close to LiDAR and object surface exactly faces LiDAR. According to the mechanism of the mechanical LiDAR, this kind of object commonly has dense LiDAR points, and its structural feature in LiDAR point cloud is clear. It is noted that the LiDAR-camera system equipped in the intelligent vehicle, so that $T \ll D_l$. It means that this kind of object is also close to the camera, which might have salient textural feature in the RGB image. In conclusion, the object with high LiDAR intensity might have salient features in both LiDAR point cloud and RGB image (shown in Figure 5), which is helpful for generating the representative CoC feature.

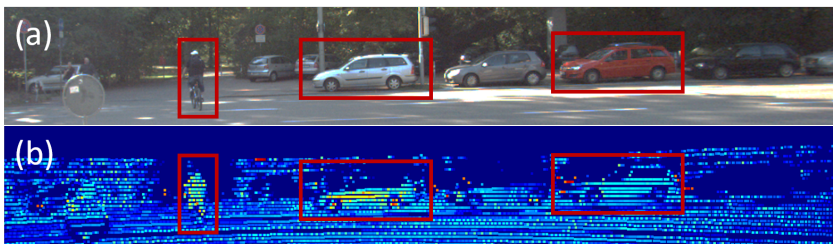

Figure 5. Visualization of (**a**) RGB image and (**b**) LiDAR intensity. Observing the red bounding boxes, the objects with higher LiDAR intensity have the clearer structural features in both RGB image and LiDAR point cloud.

Based on the above analysis, visual attention is exploited from LiDAR intensity $\mathcal{I}_{0d}$. With the efficient encoder ResNet [56], the attention feature map $\mathcal{A}$ is obtained as Equation (5). $E_I(\cdot)$ is the encoder with the first four convolution layers of ResNet-18. The reason for selecting ResNet-18 as an encoder is that it is light and efficient to extract the image feature. $\text{Conv}_{2d}(\cdot)$ is one 2D convolution layer with the single channel. Its activation function is Softmax to make sure the attention value in $\mathcal{A}$ lies in $[0, 1]$. $\mathcal{A}$ is $[W/8, H/8, 1]$ tensor. After that, with $\mathcal{A}$, an attentional LiDAR map feature $\mathcal{F}_{\text{LiDAR}}$ is extracted as Equation (6). $E_D(\cdot)$ is the encoder of LiDAR depth, as the same architecture with $E_I(\cdot)$. $\mathcal{F}_{\text{LiDAR}}$ is $[W/8, H/8, 256]$ tensor:

$$\mathcal{A} = \text{Conv}_{2d}(E_I(\mathcal{I}_{0d})) \tag{5}$$

$$\mathcal{F}_{\text{LiDAR}} = \mathcal{A} \cdot E_D(\mathcal{D}_{0d}) \tag{6}$$

Referring the co-observed feature extraction in CMR-Net [53], the CoC feature $\mathcal{F}_{\text{CoC}}$ is extracted as:

$$\mathcal{F}_{\text{CoC}} = \text{CVC}(\mathcal{F}_{\text{LiDAR}}, \mathcal{F}_{\text{Camera}}) \tag{7}$$

$$\mathcal{F}_{\text{Camera}} = E_R(\mathcal{R}) \tag{8}$$

$\text{CVC}(\cdot)$ is the cost volume computation (CVC) module [53,57], discussed in Appendix A.3. $E_R(\cdot)$ is the encoder of RGB image, as the same architecture with $E_I(\cdot)$. $\mathcal{F}_{\text{Camera}}$ is $[W/8, H/8, 256]$ tensor. $\mathcal{F}_{\text{CoC}}$ is $[d^2, W/8, H/8]$ tensor. d is the pixel range, set as the default value in the literature [53].

3.3. Parameters Regression

The parameters regression module aims to estimate $\Delta\mathbf{R}$ and ΔT. Referring the literature [51,53], the proposed network regresses the unit quaternion $\Delta\mathbf{Q}$ instead of $\Delta\mathbf{R}$, for regressing $\Delta\mathbf{R}$ needs to consider two constraints: (i) $\Delta\mathbf{R}^T\Delta\mathbf{R} = \mathbf{I}$ and (ii) $det(\Delta\mathbf{R}) = 1$, whereas $\Delta\mathbf{Q}$ only has one constraint: $\|\Delta\mathbf{Q}\|_2 = 1$. $\Delta\mathbf{Q}$ and ΔT are obtained as:

$$f_{\text{CoC}} = \text{FC}_f(\text{Squeeze}(\mathcal{F}_{\text{CoC}})) \tag{9}$$

$$\Delta\mathbf{Q} = \text{FC}_Q(f_{\text{CoC}}), \Delta T = \text{FC}_T(f_{\text{CoC}}) \tag{10}$$

$\text{Squeeze}(\cdot)$ is the operation that converts the feature map into 1D vector. FC_f is two FC layers with neurons of 1024 and 512. FC_Q is two FC layers with neurons of 256 and 4. FC_T is two FC layers with neurons of 256 and 3. As $\Delta\mathbf{Q}$ is a unit quaternion, normalization should be done in Equation (11). Finally, $\Delta\mathbf{R}$ is extracted from $\Delta\mathbf{Q}$ in Equation (12). $\text{Qua2Rot}(\cdot)$ is the operation that converts the unit quaternion as a rotation matrix, discussed in Appendix A.4:

$$\Delta\mathbf{Q} = \frac{\Delta\mathbf{Q}}{\|\Delta\mathbf{Q}\|_2} \tag{11}$$

$$\Delta\mathbf{R} = \text{Qua2Rot}(\Delta\mathbf{Q}) \tag{12}$$

3.4. Structural Consistency Loss

Regular calibration loss [51,53] is illustrated at first. Ground truths' extrinsic parameters are marked as $\Delta\mathbf{R}_{\text{gt}}$ and ΔT_{gt}. $\Delta\mathbf{Q}_{\text{gt}} = \text{Qua2Rot}(\Delta\mathbf{R}_{\text{gt}})$. Regular loss L_{reg} is presented in Equation (13). $L_{\text{smooth-L1}}(\cdot)$ is smooth $L1$ function [50]. Let Θ denote all learn-able parameters in the proposed calibration network. $\|\Theta\|_2$ is an $L2$ norm of all learn-able parameters. $\gamma > 0$ is $L2$ regulation weight, set as the default value in the literature [53]:

$$L_{reg} = L_{\text{smooth-L1}}(\Delta T, \Delta T_{\text{gt}}) + L_{\text{smooth-L1}}(\Delta\mathbf{Q}, \Delta\mathbf{Q}_{\text{gt}}) + \gamma\|\Theta\|_2 \tag{13}$$

The challenge of L_{reg} is then discussed. In the late period of the training stage, although the regression results $\Delta\mathbf{R}$ and ΔT might be close to the ground truths, it cannot promise that the alignment error of LiDAR point cloud and RGB image is small enough. An example is taken. Supposed that the focal length of the camera is $f_c = 800$ pixels and $T \ll D_l$. If the error of the rotation angle is close to $e_r = 1.0$ deg, the re-projection error of LiDAR point in the image plane is $e_p \geq \frac{\pi}{180} f_c e_r = 13.95$ pixels [7], which is larger than 8.0 pixels as the calibration tolerance error [9,10]. Therefore, we need to design a new loss term for the alignment of LiDAR point cloud and RGB image.

A naive loss term is designed to directly minimize the alignment error as Equation (14). $P_{l,i}$ is the coordinate of the i-th LiDAR point in the LiDAR coordinate system. $\mathbf{S}_{\text{FOV}}$ is the index set of LiDAR points that are fallen into FOV of the camera. $\mathbf{1}(i \in \mathbf{S}_{\text{FOV}})$ is the indicator function. It outputs 1 if the i-th point is fallen into FOV of the camera. It outputs 0 if not. $\pi(\cdot)$ is the operation of pinhole camera projection, discussed in Appendix A.1. $I_{i,\text{GT}}$ is the pixel in the image corresponding to the i-th LiDAR point:

$$L_{\text{align}} = \sum_{i=1}^{N} \|I_{i,\text{GT}} - I_i\|_2^2 \cdot \mathbf{1}(i \in \mathbf{S}_{\text{FOV}}), I_i = \pi(P'_{l,i}; \Delta\mathbf{R}, \Delta T), P'_{l,i} = \mathbf{R}_0 P_{l,i} + T_0 \quad (14)$$

However, L_{align} cannot work in the actual applications for two reasons: (i) it is difficult to find the corresponding pixel I_i for the arbitrary LiDAR point; (ii) it is also difficult to determine $\mathbf{S}_{\text{FOV}}$ with the initial and inaccurate $\mathbf{R}_0$ and T_0. Thus, L_{align} needs to be approximated for the actual application. That is the reason why loss approximation is needed in Figure 3. With $\Delta\mathbf{R}_{\text{gt}}$ and ΔT_{gt}, ground truth LiDAR depth $\mathcal{D}_{\text{gt}}$ and LiDAR intensity $\mathcal{I}_{\text{gt}}$ are obtained with the same procedure in Section 3.1. With $\Delta\mathbf{R}$ and ΔT, $\mathcal{D}(\Delta\mathbf{R}, \Delta T)$ and $\mathcal{I}(\Delta\mathbf{R}, \Delta T)$ are also obtained. It is noted that the depth and intensity of $I_{i,\text{GT}}$ and I_i are close if $\|I_{i,\text{GT}} - I_i\|_2$ is small enough. Therefore, we aim to maximize the structural consistency between $\mathcal{D}_{\text{gt}}$ and $\mathcal{D}(\Delta\mathbf{R}, \Delta T)$, or between $\mathcal{I}_{\text{gt}}$ and $\mathcal{I}(\Delta\mathbf{R}, \Delta T)$. Inspired the photo-metric loss in CalibNet [52], using L2 norm, SC loss is designed as:

$$L_{\text{SC,raw}} = \|\mathcal{D}_{\text{gt}} - \mathcal{D}(\Delta\mathbf{R}, \Delta T)\|_2^2 + \|\mathcal{I}_{\text{gt}} - \mathcal{I}(\Delta\mathbf{R}, \Delta T)\|_2^2 \quad (15)$$

Due to the sparsity of LiDAR point cloud, minimizing the sparse feature map would easily cause the discontinuity of the SC loss. Dense operation in Section 3.2 is exploited to densify the ground truths and estimated LiDAR projection maps. SC loss is revised as Equation (16). The calibration loss of LIA-SC-Net is shown in Equation (17):

$$L_{\text{SC}} = \|\mathcal{D}_{\text{gtd}} - \mathcal{D}_d(\Delta\mathbf{R}, \Delta T)\|_2^2 + \|\mathcal{I}_{\text{gtd}} - \mathcal{I}_d(\Delta\mathbf{R}, \Delta T)\|_2^2 \quad (16)$$

$$L_{\text{calib}} = L_{\text{reg}} + L_{\text{SC}} \quad (17)$$

3.5. Iterative Inference Scheme

Following the methods [52,53], the iterative inference scheme is exploited to increase the accuracy of LIA-SC-Net. Let m be the iteration time ($m = 1, ..., M$). In addition, M is the maximum iterative time. The m-th outputs are $\Delta\mathbf{R}_m$ and ΔT_m. Then, $\mathbf{R}_0$ and T_0 are updated with $\Delta\mathbf{R}_m$ and ΔT_m via Equation (1), and used as the initial parameters for the $m + 1$ iteration. In the M-th iteration, updated $\mathbf{R}_0$ and T_0 are output as the iterative inference calibration results. The selection of M is discussed in the experiment section.

4. Experimental Results

4.1. Experiment Configuration

4.1.1. Dataset and Preparations

The performance of the proposed method is evaluated on a KITTI autonomous driving dataset [2]. In the KITTI outdoor dataset, the LiDAR-camera system consists of Velodyne HDL-64E LiDAR (10 Hz, 64 laser beams) and PointGray Flea2 video RGB camera (10 Hz) [2]. RGB image and LiDAR point cloud are used to prepare our dataset. It contains 7481 samples for training and 7518 samples for validation. The training and validation datasets are randomly selected from all 28 different outdoor scenes. As the size of image in the KITTI dataset is $[1242, 375]$, all images are padded to $[1280, 384]$ to meet the CNN architecture requirement (width and height multiple of 64). Although the KITTI dataset was established in 2012, it is still the main public benchmark for mainstream applications for the autonomous driving [17]. To simulate the situation of online extrinsic calibration, we follow the approach of the current deep learning based extrinsic calibration methods [51–53] to generate $\Delta\mathbf{R}$ and ΔT. $\Delta\mathbf{R}$ can be represented with yaw–pitch–roll Euler form with three angles $(\Delta\theta_y, \Delta\theta_p, \Delta\theta_r)^T$. These angles are randomly sampled from a uniform distribution in the range of $[-\theta, \theta]$ (Unit: degree). ΔT has three elements $(\Delta x, \Delta y, \Delta z)^T$, and they are randomly sampled from a uniform distribution in the range of $[-D, D]$ (Unit: meter).

4.1.2. Implementations

To verify the deep learning based methods in different situations, six noise levels are presented in Table 1. According to the classical learning based method [15], for the

practical application of the LiDAR-camera system, the drifted rotation error is smaller than 20 deg, and the drifted translation error is smaller than 1.50 m. For the fair comparison, all learning based methods are trained with level-3 calibration situation, and then tested with level-0 to level-5 calibration situations. Level-0 is used to test the stability of the learning based method. Based on an open-source optical flow network PWC-Net [57], CMR-Net [53] and LIA-SC-Net are implemented with the PyTorch library. Open-source CalibNet [52] is implemented with Tensorflow library. LiDAR depth and reflected intensity are normalized to the range of $[0, 1]$. $\gamma = 0.004$. Batch size is set as 1 for training. LIA-SC-Net, CMR-Net, and CalibNet are trained for 40 epochs using an SGD optimizer on a single Nvidia GTX 1080ti.

Table 1. Different calibration situations for verification on deep learning based methods.

Levels	**0**	**1**	**2**	**3**	**4**	**5**
θ/ deg	0.0	4.0	8.0	12.0	16.0	20.0
D/m	0.0	0.30	0.60	0.90	1.20	1.50
Training	×	×	×	✓	×	×
Testing	✓	✓	✓	✓	✓	✓

4.1.3. Evaluation Metrics

GT rotation matrix $\Delta\mathbf{R}_{\text{gt}}$ is represented with yaw–pitch–roll Euler form using three angles $(\Delta\theta_{y,gt}, \Delta\theta_{p,gt}, \Delta\theta_{r,gt})^T$. GT translation vector ΔT_{gt} is denoted as $(\Delta t_{x,gt}, \Delta t_{y,gt}, \Delta t_{z,gt})^T$. Calibration metrics are presented as Equations (18) and (19) for evaluation. The average errors in the validation dataset are used to evaluate the performance of learning based calibration method:

$$E_\theta = \frac{|\Delta\theta_{y,gt} - \Delta\theta_y| + |\Delta\theta_{p,gt} - \Delta\theta_p| + |\Delta\theta_{r,gt} - \Delta\theta_r|}{3} \tag{18}$$

$$E_t = \frac{|\Delta t_{x,gt} - \Delta t_x| + |\Delta t_{y,gt} - \Delta t_y| + |\Delta t_{z,gt} - \Delta t_z|}{3} \tag{19}$$

4.2. Verification of the Proposed Method

To evaluate the LIA-Net backbone network and SC loss in LIA-SC-Net, five baselines are designed in Table 2. Encoder means that the calibration network only uses the encoders E_D and E_R to extract CoC features. Calibration errors are presented in Figure 6. Error curves from Baselines-1 to Baselines-5 are marked as deep blue, light blue, green, yellow, and pink, respectively. Average and standard deviation errors of all baseline methods are presented in Table 3.

Table 2. Five baselines for the verification of LIA-SC-Net.

Methods	**Dense Operation**	**Encoder**	**LIA-Net**	L_{reg}	L_{SC}
baseline-1	×	✓	×	✓	×
baseline-2	✓	✓	×	✓	×
baseline-3 (LIA+reg)	✓	×	✓	✓	×
baseline-4 (LIA+SC)	✓	×	✓	×	✓
baseline-5 (LIA-SC-Net)	✓	×	✓	✓	✓

Table 3. Average and standard deviation of rotation and translation errors for five baseline methods in the validation dataset. Gain of the average calibration error is also provided.

Methods	Rotation Error/deg	Translation Error/m
Baseline-1	0.869/0.062	0.0693/0.0087
Baseline-2	0.812/0.054	0.0648/0.0074
Baseline-3	0.786/0.052	0.0601/0.0068
Baseline-4	0.768/0.047	0.0538/0.0064
Baseline-5	0.751/0.045	0.0447/0.0061
Gain	15.71%	55.03%

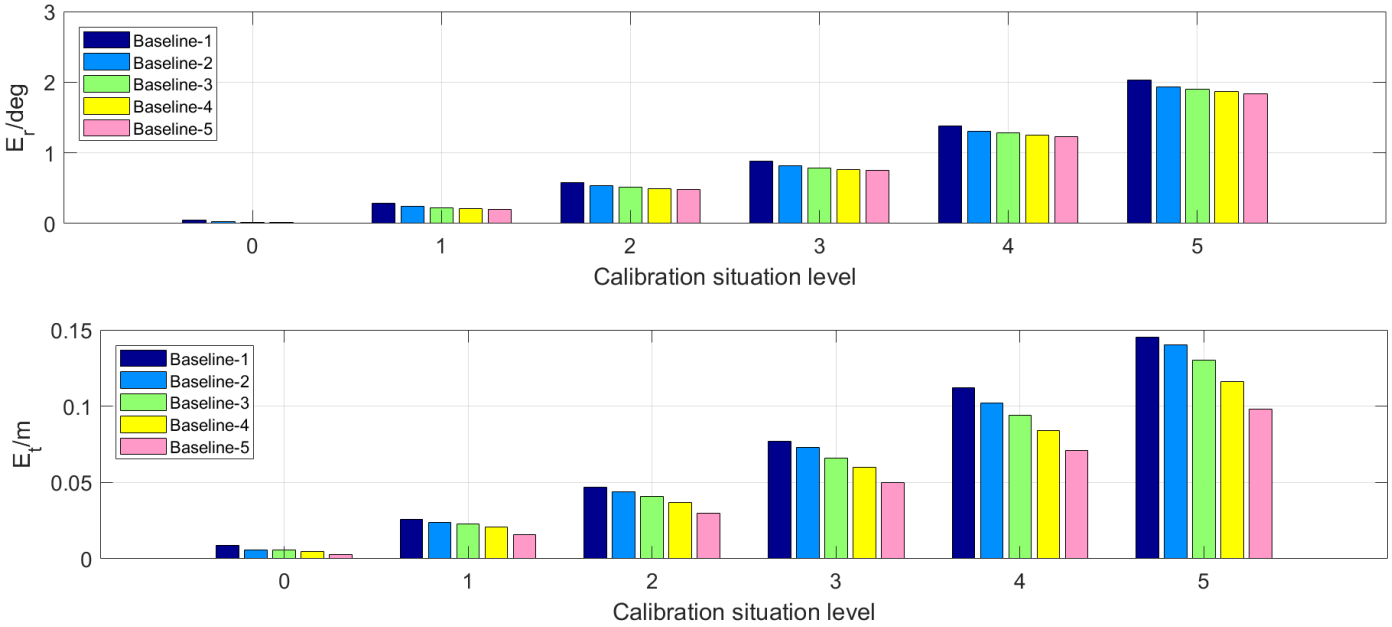

Figure 6. Calibration performance of five baseline methods in the validation dataset.

4.2.1. Verification on Dense Operation

This experiment investigates the performance of dense operation for LiDAR depth. From Table 3, compared with baseline-1 and baseline-2, it is found that the rotation and translation errors are decreased by 0.057 deg and 0.45 cm with the dense operation. As presented in Figure 4, a dense LiDAR map has more salient object structural information than the sparse LiDAR map. Thus, the representative CoC feature can be generated from the dense LiDAR map and RGB image. It is concluded that dense operation is effective for the extrinsic online calibration for the sparse LiDAR point cloud.

4.2.2. Verification on LiDAR Intensity Attention

This experiment investigates the performance of LiDAR intensity attention. From Figure 6 and Table 3, compared with baseline-2 and baseline-3, it is found that baseline-3 has smaller calibration errors from levels 1 to 5. The average rotation and translation errors are decreased by 0.026 deg and 0.47 cm. Standard deviation is also decreased. Compared with the encoder, LIA-Net exploits LiDAR intensity as visual attention, which can adaptively select the area in the LiDAR depth for the salient calibration feature extraction. It means that LiDAR intensity attention is helpful for CoC feature generation.

4.2.3. Verification on Structural Calibration Loss

This experiment investigates the performance of the proposed SC loss. From Figure 6 and Table 3, compared with baseline-3, baseline-4, and baseline-5, the translation errors

are decreased significantly with the calibration levels. Compared with baseline-1, rotation and translation errors of baseline-5 are dropped by 15.71% and 55.03%. Visualizations of baseline-1 and baseline-5 are shown in Figure 7. It means that combining the regular loss and the proposed SC loss improves the training efficiency of the calibration network.

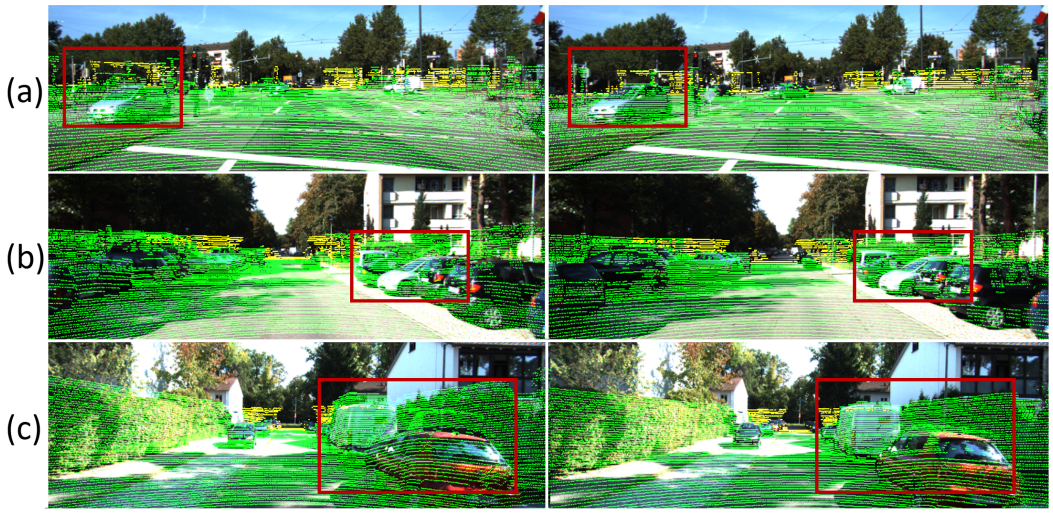

Figure 7. Comparisons with the baseline-1 (**left**) and baseline-5 (**right**) in the validation dataset for the various outdoor scenes (**a**–**c**). Comparing the alignment of LiDAR points and RGB pixels of the objects in the red boxes, baseline-5 has the smaller alignment errors. The projected LiDAR point cloud with pseudo color depended on LiDAR intensity is used for visualization.

4.2.4. Verification on Iterative Inference and Time-Consuming Test

This experiment investigates the performance of iterative inference and time-consuming. With the iterative time M increasing from 1 to 5, the calibration performance is presented in Table 4. Calibration accuracy increases with M. When $M \geq 4$, the gains of calibration errors are not obvious because the calibration network has difficultly generating more significant calibration features so that LIA-SC-Net considers the current results as accurate enough. We also investigate the performance of time efficiency of the proposed method. CPU-GPU transfer time was not considered. Time consumption of each module is presented in Table 5. Frame per second (FPS) of the proposed method in the non-iterative inference mode is nearly 34, higher than the fresh rate of the mechanical LiDAR (nearly 10 Hz) and camera (nearly 30 Hz). In the case that $M = 5$, the proposed method runs 146 ms (nearly 6.8 FPS). In conclusion, LIA-SC-Net works for the online extrinsic calibration and iterative inference strategy is useful in the actual application.

Table 4. Performance of iterative inference of LIA-SC-Net in the validation dataset.

M	**1**	**2**	**3**	**4**	**5**
E_r / deg	0.751	0.654	0.538	0.525	0.525
E_t / m	0.0447	0.0402	0.0396	0.0396	0.0396

Table 5. Runtime of LIA-SC-Net in the validation dataset (Unit: ms).

Module	**Preparation**	**LIA-Net**	**Regression**	**Total**
Time	7.6	13.5	8.1	29.2

4.3. Comparison Results

4.3.1. Comparisons with State-of-the-Art Methods

For the calibration performance evaluation, our method is compared with state-of-the-art methods, such as CalibNet [52] and CMR-Net [53]. As discussed in Section 4.1.1, these

methods are all trained on the same validation dataset, with the same epochs and optimizer. For the fair comparisons on all methods, iterative inference strategy is used for the best calibration results. Rotation and translation errors of each calibration level are presented in Figure 8. All methods have nearly the same accuracy for levels 0 to 2. For the large initial calibration errors, it is found that LIA-SC-Net outperforms other methods. Average calibration errors of all levels are provided in Table 6. LIA-SC-Net has the smaller average rotation and translation errors than other methods. In addition, the runtime comparison results in the non-iterative inference mode are shown in Table 7. Although the inference time of LIA-SC-Net is not the fastest, it satisfied the requirement of the real-time online calibration. Therefore, the proposed calibration method meets the need of accurate and real-time extrinsic calibration on the LiDAR-camera system.

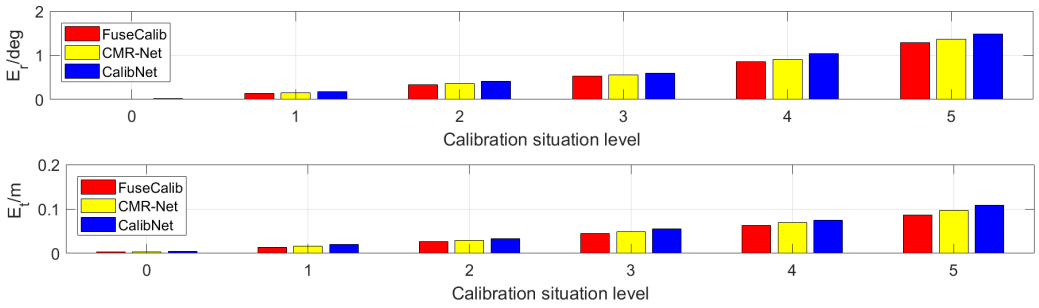

Figure 8. Calibration performance of deep learning based methods in the validation dataset.

Table 6. Average calibration errors of deep learning based methods in the validation dataset.

Methods	Rotation Error/deg	Translation Error/m
CalibNet	0.621	0.0495
CMR-Net	0.557	0.0437
LIA-SC-Net (Our)	0.525	0.0396

Table 7. Runtime comparison with deep learning based methods in the validation dataset (Unit: ms).

Methods	CalibNet	CMR-Net	LIA-SC-Net
Time	36.8	14.7	29.2

4.3.2. Calibration Visualization

Visualization results of LIA-SC-Net and CMR-Net [53] on the large initial calibration case are provided in Figure 9. For better visualization, RGB image and LiDAR dense depth are mixed. Red dotted lines are the borderline of LiDAR depth. It is found that LIA-SC-Net has better performance than CMR-Net [53] in these scenes. With the iterative inference strategy, calibration results of our method are close to the ground truths. However, in the first scene, results of LIA-SC-Net with $M = 5$ seem to have relatively large rotation error. The reason might be that few salient objects with the low depth and high LiDAR reflected intensity are founded in this scene. Therefore, for precise calibration, our method requires the region close to the LiDAR-camera system with higher LiDAR reflected intensity.

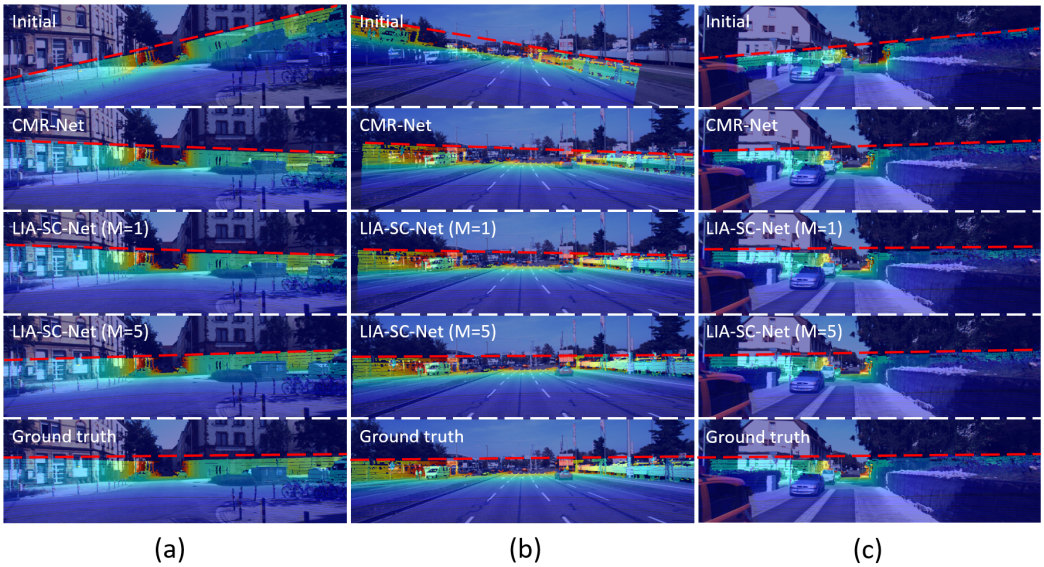

Figure 9. Calibration visualization of different learning based calibration methods in the various scenes (**a**–**c**). Images mixed with LiDAR depth and RGB image are used for visualization.

5. Discussion

The proposed method LIA-SC-Net has large potential advantages for the online extrinsic calibration on the LiDAR-camera system. Compared with the calibration object based method [11,12], it does not exploit any specific calibration object. Among the target-less based method, the proposed method does not need to compute the odometry from LiDAR point cloud and RGB image [38], or reconstruct 3D point cloud from an RGB image [37]. Due to the sparsity of LiDAR point cloud, LiDAR depth and reflected intensity are also sparse (shown in Figure 4), causing the inaccuracy of MI features [40,41]. Compared with the other deep learning based calibration methods [51–53], the proposed method notices the relation of the co-observed features and LiDAR intensity, thus proposing LIA-Net as the novel backbone network to generate more salient CoC features. In future work, we attempt to refine LIA-Net with both spatial and channel attention [62] to achieve more accurate calibration results.

The proposed method LIA-SC-Net provides the aligned LiDAR point cloud and RGB image for the advanced applications on the LiDAR-camera system, such as depth completion [61] and 3D object detection [63]. As LiDAR depth is sparse in Figure 4b, depth completion can use the texture feature in the aligned RGB image as guidance [64] to generate a dense depth map. As LiDAR point cloud and RGB image are aligned, the multi-sensor fusion feature can be generated to regress the accurate 3D localization of targeted object. It means that LIA-SC-Net has wide applications on the LiDAR-camera system.

Experiments demonstrate that LIA-SC-Net has stable calibration performance and outperforms current methods. It means that the proposed LIA-Net module and SC loss are useful to ensure that LIA-SC-Net generates the robust and precise calibration results. A limitation of our method is the compatibility of the real-time and accurate performance. From the analysis in Section 4.2.4, $M \geq 4$ achieves better results, but FPS is lower than 10. Therefore, we would speed up the proposed method in the future work.

6. Conclusions

In this paper, to solve the accumulated extrinsic parameter errors of the LiDAR-camera system, we present a novel and efficient extrinsic calibration network LIA-SC-Net. From the Lambertian reflection model, it is found that the object with higher LiDAR intensity has the potential of providing the salient co-observed feature. LIA-Net is presented to exploit LiDAR intensity as the attention feature map to generate the representative CoC feature. To increase the training efficiency of calibration network, SC loss is proposed to minimize

the alignment errors of LiDAR point cloud and RGB image. With LIA-Net and SC loss, the proposed method achieves the accurate and robust performance in the dataset experiment. It demonstrates the effectiveness of LIA-SC-Net.

Author Contributions: Methodology and writing—original draft preparation, P.A.; software and validation, Y.G.; investigation, L.W.; formal analysis, Y.C.; supervision, resources, and funding acquisition, J.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (U1913602, 61991412). Equipment Pre-Research Project (41415020202, 41415020404, 305050203).

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: http://www.cvlibs.net/datasets/kitti/ accessed on 18 April 2022.

Acknowledgments: The authors thank Siying Ke, Bin Fang, Junfeng Ding, Zaipeng Duan, and Zhenbiao Tan from Huazhong University of Science and Technology, and anonymous reviewers for providing many valuable suggestions.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

LiDAR	Light detection and ranging
MI	Mutual information
CNN	Convolution neural network
LIA-Net	LiDAR intensity attention based backbone network
SC	Structural consistency
FOV	Field of view
FPS	Frame per second
GT	Ground truth
Net	Network
CoC	Co-observed calibration
PnP	Perspective-n-point
DLT	Direct linear transformation
BA	Bundle adjustment
ICP	Iterative closest point
BLSMI	Bagged least-squares mutual information
VO	Visual odometry
FC	Fully connected
CVC	Cost volume computation
FPS	Frame per second

Appendix A

Appendix A.1. Projection Details

Let $P_l = (x_l, y_l, z_l)^T \in \mathcal{P}_l$ denote the LiDAR coordinate of one LiDAR laser point. r_l is its LiDAR reflected intensity. With the extrinsic parameters $\mathbf{R}$ and T, P_l can be projected into the image plane via camera pinhole model [7]:

$$d_l \cdot I_l = \mathbf{K} \cdot (\mathbf{R}P_l + T) \tag{A1}$$

$$\mathbf{K} = \begin{pmatrix} f_u & f_s & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{pmatrix} \tag{A2}$$

$\mathbf{K}$ is the camera intrinsic matrix. $(u_0, v_0, 1)^T$ is the pixel coordinates of the principle point. f_u and f_v are the focal lengths of horizontal and vertical axes on the image plane. f_s is the skew ratio. $I = (u, v, 1)^T$ is the pixel coordinates of laser point P_l. If this laser point falls into the camera field of view (FOV), it has positive depth $d_l > 0$. Projecting all LiDAR

points fallen into the FOV of the camera via Equation (A1), collecting I, d_l, and r_l, LiDAR depth $\mathcal{D}$ and LiDAR intensity $\mathcal{I}$ intensity images are generated. They are $[W, H, 1]$ tensors where W and H are the width and height of the image. For simplicity, Equation (A1) can be rewritten as:

$$I_l = \pi(P_l; \mathbf{R}, T) \tag{A3}$$

Table A1. Intrinsic parameters of LiDAR-camera system in the KITTI outdoor dataset [2].

Parameter	Values
f_u	721.537 pixel
f_v	721.537 pixel
f_s	0.0 pixel
u_0	609.559 pixel
v_0	216.379 pixel
θ_h	0.08 deg
θ_v	0.40 deg

Appendix A.2. Selection of k in the Dense Operation

Kernel size k of dilation is selected to make sure that each dilated pixel overlaps with its neighbored dilated pixels. Let θ_h and θ_v be the horizontal and vertical resolution angles of LiDAR. From Equation (A3), k should approximately satisfy the following constraints:

$$\frac{\pi}{180}\theta_h \leq \frac{k}{f_u} \leq 2\frac{\pi}{180}\theta_h, \ \frac{\pi}{180}\theta_v \leq \frac{k}{f_v} \leq 2\frac{\pi}{180}\theta_v \tag{A4}$$

$$\frac{\pi}{180} \cdot \frac{1}{2}(\theta_v f_u + \theta_h f_v) \leq k \leq \frac{\pi}{180} \cdot (\theta_v f_u + \theta_h f_v) \tag{A5}$$

For the mechanical LiDAR, θ_v is far larger than θ_h, and a compromise selection scheme is exploited as Equation (A5). As the experiments are conducted in the KITTI dataset [2], according to the parameters of LiDAR-camera system in KITTI dataset [2] (Table A1), k is set as 5.

Appendix A.3. Details of the Cost Volume Computation Module

To extract co-observed features from $\mathcal{F}_{\text{LiDAR}}$ and $\mathcal{F}_{\text{Camera}}$ for calibration, the matching cost computation module [53,57] is exploited. The CoC feature $\mathcal{F}_{\text{CoC}}$ is computed as:

$$\mathcal{F}_{\text{CoC}}(p_1, p_2) = \frac{1}{c}(\mathcal{F}_{\text{LiDAR}}(p_1) - \bar{\mathcal{F}}_{\text{LiDAR}})^T(\mathcal{F}_{\text{Camera}}(p_2) - \bar{\mathcal{F}}_{\text{Camera}}) \tag{A6}$$

$$\bar{\mathcal{F}}_{\text{LiDAR}} = \text{Mean}(\mathcal{F}_{\text{LiDAR}}), \bar{\mathcal{F}}_{\text{Camera}} = \text{Mean}(\mathcal{F}_{\text{Camera}}) \tag{A7}$$

$p_i = (u_i, v_i)$ is the pixel index of the 2D feature map. $\mathcal{F}_{\text{LiDAR}}(p_1)$ and $\mathcal{F}_{\text{Camera}}(p_2)$ are both $[1, c]$ vectors where $c = 256$ in this paper (as shown in Section 3.2). To extract the robust CoC feature from images with different sources, normalization is used. Mean$(\cdot)$ generates the average feature from a given 2D feature map. $\bar{\mathcal{F}}_{\text{LiDAR}}$ and $\bar{\mathcal{F}}_{\text{Camera}}$ are both $[1, c]$ vectors. T is the vector transpose operator. For the computation efficiency [57], we only compute a partial cost volume with a limited range of d pixel ($\|p_1 - p_2\|_\infty \leq d/2$). As $\mathcal{F}_{\text{LiDAR}}$ and $\mathcal{F}_{\text{Camera}}$ are $[W/8, H/8, 256]$ tensors, $\mathcal{F}_{\text{CoC}}$ is a $[d^2, W/8, H/8]$ tensor.

Appendix A.4. Relation of the Quaternion and Rotation Matrix

Let $\mathbf{R} = (r_{ij})_{3\times3}$ be a rotation matrix. Let $\mathbf{Q} = (q_0, q_1, q_2, q_3)^T$ be a unit quaternion, which is also a 4×1 unit vector. $\mathbf{R}$ can be converted from $\mathbf{Q}$ as:

$$\mathbf{R} = \begin{pmatrix} 1-2q_2^2-2q_3^2 & 2q_1q_2-2q_0q_3 & 2q_1q_3+2q_0q_2 \\ 2q_1q_2-2q_0q_3 & 1-2q_1^2-2q_3^2 & 2q_2q_3-2q_0q_1 \\ 2q_1q_3-2q_0q_2 & 2q_2q_3+2q_0q_1 & 1-2q_1^2-2q_2^2 \end{pmatrix} \tag{A8}$$

Q can be converted from **R** as:

$$q_0 = \frac{\sqrt{tr(\mathbf{R})+1}}{2}, q_1 = \frac{r_{23}-r_{32}}{4q_0}, \tag{A9}$$

$$q_2 = \frac{r_{31}-r_{13}}{4q_0}, q_3 = \frac{r_{12}-r_{21}}{4q_0} \tag{A10}$$

References

1. Qi, C.R.; Liu, W.; Wu, C.; Su, H.; Guibas, L.J. Frustum PointNets for 3D Object Detection From RGB-D Data. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 918–927.
2. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.
3. Nagy, B.; Benedek, C. On-the-Fly Camera and Lidar Calibration. *Remote Sens.* **2020**, *12*, 1137. [CrossRef]
4. Ku, J.; Mozifian, M.; Lee, J.; Harakeh, A.; Waslander, S.L. Joint 3D Proposal Generation and Object Detection from View Aggregation. In Proceedings of the International Conference on Intelligent Robots and Systems, Madrid, Spain, 1–5 October 2018; pp. 1–8.
5. Zhuang, Z.; Li, R.; Jia, K.; Wang, Q.; Li, Y.; Tan, M. Perception-Aware Multi-Sensor Fusion for 3D LiDAR Semantic Segmentation. In Proceedings of the IEEE Conference on International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 16260–16270.
6. Yin, L.; Luo, B.; Wang, W.; Yu, H.; Wang, C.; Li, C. CoMask: Corresponding Mask-Based End-to-End Extrinsic Calibration of the Camera and LiDAR. *Remote Sens.* **2020**, *12*, 1925. [CrossRef]
7. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1330–1334. [CrossRef]
8. Liu, W.; Wang, C.; Chen, S.; Bian, X.; Lai, B.; Shen, X.; Cheng, M.; Lai, S.; Weng, D.; Li, J. Y-Net: Learning Domain Robust Feature Representation for ground camera image and large-scale image-based point cloud registration. *Inf. Sci.* **2021**, *581*, 655–677. [CrossRef]
9. An, P.; Ma, T.; Yu, K.; Fang, B.; Zhang, J.; Fu, W.; Ma, J. Geometric calibration for LiDAR-camera system fusing 3D-2D and 3D-3D point correspondences. *Opt. Express* **2020**, *28*, 2122–2141. [CrossRef]
10. An, P.; Gao, Y.; Ma, T.; Yu, K.; Fang, B.; Zhang, J.; Ma, J. LiDAR-camera system extrinsic calibration by establishing virtual point correspondences from pseudo calibration objects. *Opt. Express* **2020**, *28*, 18261–18282. [CrossRef]
11. Park, Y.; Yun, S.; Won, C.; Cho, K.; Um, K.; Sim, S. Calibration between color camera and 3D LIDAR instruments with a polygonal planar board. *Sensors* **2014**, *14*, 5333–5353. [CrossRef]
12. Dhall, A.; Chelani, K.; Radhakrishnan, V.; Krishna, K.M. LiDAR-Camera Calibration using 3D-3D Point correspondences. *arXiv* **2017**, arXiv:1705.09785.
13. Pusztai, Z.; Hajder, L. Accurate Calibration of LiDAR-Camera Systems using Ordinary Boxes. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Venice, Italy, 22–29 October 2017; pp. 394–402.
14. Pandey, G.; Mcbride, J.R.; Savarese, S.; Eustice, R.M. Automatic Targetless Extrinsic Calibration of a 3D Lidar and Camera by Maximizing Mutual Information. In Proceedings of the Twenty-Sixth Conference on Artificial Intelligence, Toronto, ON, Canada, 22–26 July 2012 ; pp. 1–7.
15. Wu, X.; Zhang, C.; Liu, Y. Calibrank: Effective Lidar-Camera Extrinsic Calibration by Multi-Modal Learning to Rank. In Proceedings of the IEEE International Conference on Image Processing, Abu Dhabi, United Arab Emirates, 25–28 October 2020; pp. 3189–3193.
16. Oren, M.; Nayar, S.K. Generalization of the Lambertian model and implications for machine vision. *Int. J. Comput. Vis.* **1995**, *14*, 227–251. [CrossRef]
17. Yoo, J.H.; Kim, Y.; Kim, J.; Choi, J.W. 3D-CVF: Generating Joint Camera and LiDAR Features Using Cross-View Spatial Feature Fusion for 3D Object Detection. In Proceedings of the ECCV, Glasgow, UK, 23–28 August 2020; pp. 1–16.
18. Glorot, X.; Bordes, A.; Bengio, Y. Deep Sparse Rectifier Neural Networks. In Proceedings of the ICAIS, Klagenfurt, Austria, 6–8 September 2011; pp. 315–323.
19. Mnih, V.; Heess, N.; Graves, A.; Kavukcuoglu, K. Recurrent Models of Visual Attention. In Proceedings of the NeurIPS, Montreal, QC, Canada, 8–13 December 2014; pp. 2204–2212.
20. Ribeiro, A.H.; Tiels, K.; Aguirre, L.A.; Schön, T.B. Beyond exploding and vanishing gradients: Analysing RNN training using attractors and smoothness. In Proceedings of the AISTATS, virtually, 18 September 2020; Volume 108, pp. 2370–2380.
21. Lepetit, V.; Noguer, F.M.; Fua, P. EPnP: An accurate O(n) solution to the PnP problem. *Int. J. Comput. Vis.* **2009**, *81*, 155–166. [CrossRef]

22. Gao, X.; Hou, X.; Tang, J.; Cheng, H. Complete Solution Classification for the Perspective-Three-Point Problem. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 930–943.
23. Abdel-Aziz, Y.I.; Karara, H.M. Direct linear transformation into object space coordinates in close-range photogrammetry. In Proceedings of the Symposium on Close-Range Photogrammetry, Urbana, IL, USA, 26–29 January 1971; pp. 1–18.
24. Triggs, B.; Mclauchlan, P.F.; Hartley, R.I.; Fitzgibbon, A.W. Bundle Adjustment—A Modern Synthesis. In Proceedings of Workshop on Vision Algorithms, Corfu, Greece, 21–22 September 2000; pp. 298–372.
25. Ge, Y.; Maurer, C.R.; Fitzpatrick, J.M. Surface-based 3D image registration using the iterative closest point algorithm with a closest point transform. *Proc. SPIE- Int. Soc. Opt. Eng.* **1996**, *2710*, 358–367.
26. Horn, B.K.P.; Hilden, H.M.; Negahdaripour, S. Closed-form Solution of Absolute Orientation Using Orthonormal Matrices. *J. Opt. Soc. Am. A* **1988**, *5*, 1127–1135. [CrossRef]
27. Guindel, C.; Beltrán, J.; Martín, D.; Garcia, F. Automatic Extrinsic Calibration for Lidar-Stereo Vehicle Sensor Setups. In Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), Yokohama, Japan, 16–19 October 2017; pp. 1–6.
28. Zhou, L.; Deng, Z. Extrinsic calibration of a camera and a lidar based on decoupling the rotation from the translation. In Proceedings of the IEEE Intelligent Vehicles Symposium, Madrid, Spain, 3–7 June 2012; pp. 642–648.
29. Wang, W.; Sakurada, K.; Kawaguchi, N. Reflectance Intensity Assisted Automatic and Accurate Extrinsic Calibration of 3D LiDAR and Panoramic Camera Using a Printed Chessboard. *Remote Sens.* **2017**, *9*, 851. [CrossRef]
30. Cui, J.; Niu, J.; Ouyang, Z.; He, Y.; Liu, D. ACSC: Automatic Calibration for Non-repetitive Scanning Solid-State LiDAR and Camera Systems. *arXiv* **2020**, arXiv:2011.08516.
31. Ou, J.; Huang, P.; Zhou, J.; Zhao, Y.; Lin, L. Automatic Extrinsic Calibration of 3D LIDAR and Multi-Cameras Based on Graph Optimization. *Sensors* **2022**, *22*, 2221. [CrossRef]
32. Gong, X.; Lin, Y.; Liu, J. 3D LIDAR-Camera Extrinsic Calibration Using an Arbitrary Trihedron. *Sensors* **2013**, *13*, 1902–1918. [CrossRef]
33. Lee, G.; Lee, J.; Park, S. Calibration of VLP-16 Lidar and multi-view cameras using a ball for 360 degree 3D color map acquisition. In Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, Daegu, Korea, 16–18 November 2017; pp. 64–69.
34. Pusztai, Z.; Eichhardt, I.; Hajder, L. Accurate Calibration of Multi-LiDAR-Multi-Camera Systems. *Sensors* **2018**, *18*, 2139. [CrossRef]
35. Chai, Z.; Sun, Y.; Xiong, Z. A Novel Method for LiDAR Camera Calibration by Plane Fitting. In Proceedings of the IEEE International Conference on Advanced Intelligent Mechatronics, Auckland, New Zealand, 9–12 July 2018; pp. 286–291.
36. Kümmerle, J.; Kühner, T. Unified Intrinsic and Extrinsic Camera and LiDAR Calibration under Uncertainties. In Proceedings of the IEEE International Conference on Robotics and Automation, Paris, France, 31 May–31 August 2020; pp. 6028–6034.
37. Caselitz, T.; Steder, B.; Ruhnke, M.; Burgard, W. Monocular camera localization in 3d lidar maps. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Daejeon, Korea, 9–14 October 2016; pp. 1926–1931.
38. Ryoichi, I.; Takeshi, O.; Katsushi, I. LiDAR and Camera Calibration using Motion Estimated by Sensor Fusion Odometry. In Proceedings of the IEEE International Conference on Intelligence Robots and Systems, Madrid, Spain, 1–5 October 2018; pp. 7342–7349.
39. Chen, S.; Li, X.; Zhao, L. Multi-source remote sensing image registration based on sift and optimization of local self-similarity mutual information. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Beijing, China, 10–15 July 2016; pp. 2548–2551.
40. Wolcott, R.W.; Eustice, R.M. Visual localization within LIDAR maps for automated urban driving. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 176–183.
41. Irie, K.; Sugiyama, M.; Tomono, M. Target-less camera-LiDAR extrinsic calibration using a bagged dependence estimator. In Proceedings of the IEEE International Conference on Automation Science and Engineering, Fort Worth, TX, USA, 21–25 August 2016; pp. 1340–1347.
42. Zhu, Y.; Li, C.; Zhang, Y. Online Camera-LiDAR Calibration with Sensor Semantic Information. In Proceedings of the International Conference on Robotics and Automation, Paris, France, 31 May–31 August 2020; pp. 4970–4976.
43. Xiao, Z.; Li, H.; Zhou, D.; Dai, Y.; Dai, B. Accurate extrinsic calibration between monocular camera and sparse 3D Lidar points without markers. In Proceedings of the Intelligent Vehicles Symposium, Los Angeles, CA, USA, 11–14 June 2017; pp. 424–429.
44. John, V.; Long, Q.; Liu, Z.; Mita, S. Automatic calibration and registration of lidar and stereo camera without calibration objects. In Proceedings of the IEEE International Conference on Vehicular Electronics and Safety, Yokohama, Japan, 5–7 November 2015; pp. 231–237.
45. Schönberger, J.L.; Frahm, J. Structure-from-Motion Revisited. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4104–4113.
46. Mur-Artal, R.; Tardós, J.D. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [CrossRef]
47. Fassi, I.; Legnani, G. Hand to sensor calibration: A geometrical interpretation of the matrix equation $AX = XB$. *J. Field Robot.* **2005**, *22*, 497–506. [CrossRef]
48. Ban, X.; Wang, H.; Chen, T.; Wang, Y.; Xiao, Y. Monocular Visual Odometry Based on Depth and Optical Flow Using Deep Learning. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–19. [CrossRef]

49. Chien, H.; Klette, R.; Schneider, N.; Franke, U. Visual odometry driven online calibration for monocular LiDAR-camera systems. In Proceedings of the IEEE International Conference on Pattern Recognition, Cancun, Mexico, 4–8 December 2016; pp. 2848–2853.
50. Girshick, R.B. Fast R-CNN. In Proceedings of the IEEE Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
51. Schneider, N.; Piewak, F.; Stiller, C.; Franke, U. RegNet: Multimodal sensor registration using deep neural networks. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017.
52. Ganesh, I.; R, K.R.; Krishna, M.J.; et al. CalibNet: Self-Supervised Extrinsic Calibration using 3D Spatial Transformer Networks. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018.
53. Cattaneo, D.; Vaghi, M.; Ballardini, A.L.; Fontana, S.; Sorrenti, D.G.; Burgard, W. CMRNet: Camera to LiDAR-Map Registration. In Proceedings of the IEEE International Conference on Intelligent Transportation Systems (ITSC), Auckland, New Zealand, 27–30 October 2019.
54. Neubert, P.; Schubert, S.; Protzel, P. Sampling-based methods for visual navigation in 3D maps by synthesizing depth images. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Vancouver, BC, Canada, 24–28 September 2017; pp. 2492–2498.
55. Lin, M.; Chen, Q.; Yan, S. Network In Network. In Proceedings of the International Conference on Learning Representations, Banff, Canada, 14–16 April 2014; pp. 1–10.
56. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
57. Sun, D.; Yang, X.; Liu, M.; Kautz, J. PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8934–8943.
58. Yuan, K.; Guo, Z.; Wang, Z.J. RGGNet: Tolerance Aware LiDAR-Camera Online Calibration With Geometric Deep Learning and Generative Model. *IEEE Robot. Autom. Lett.* **2020**, *5*, 6956–6963. [CrossRef]
59. Ye, C.; Pan, H.; Gao, H. Keypoint-Based LiDAR-Camera Online Calibration With Robust Geometric Network. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–11. [CrossRef]
60. Ku, J.; Harakeh, A.; Waslander, S.L. In defense of classical image processing: Fast depth completion on the CPU. In Proceedings of the IEEE Conference on Computer and Robot Vision, Toronto, ON, Canada, 8–10 May 2018; pp. 16–22.
61. An, P.; Fu, W.; Gao, Y.; Ma, J.; Zhang, J.; Yu, K.; Fang, B. Lambertian Model-Based Normal Guided Depth Completion for LiDAR-Camera System. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5. [CrossRef]
62. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual Attention Network for Scene Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Computer Vision Foundation, Long Beach, CA, USA, 15–20 June 2019; pp. 3146–3154.
63. An, P.; Liang, J.; Yu, K.; Fang, B.; Ma, J. Deep structural information fusion for 3D object detection on LiDAR-camera system. *Comput. Vis. Image Underst.* **2022**, *214*, 103295. [CrossRef]
64. Mukherjee, S.; Mohana, R.; Guddeti, R. A Hybrid Algorithm for Disparity Calculation From Sparse Disparity Estimates Based on Stereo Vision. In Proceedings of the International Conference on Signal Processing and Communications, Bangalore, India, 22–25 July 2014; pp. 1–6.