2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

LiDAR-camera Online Calibration by Representing Local Feature and Global Spatial Context

SeongJoo Moon[1,2], Sebin Lee[1], Dong He[2], and Sung-Eui Yoon[1*]

***Abstract*— LiDAR-camera calibration plays a crucial role in autonomous driving. However, operation-induced factors such as physical vibrations and temperature variations degrade the pre-deployment calibration accuracy, leading to the environmental perception performance deterioration. Recent re-calibration methods have achieved online calibration without a target board by leveraging the relative attributes of LiDAR and camera. Nevertheless, we proposes a novel framework for LiDAR-camera online calibration which employs a Transformer network to learn crucial interactions between cameras and LiDAR sensors. Additionally, our novel framework design enables the effective calibration by utilizing correspondence point information between the two sensors. This allows the utilization of global spatial context and achieves high performance by integrating information across modalities. Experimental results indicate that our method demonstrates superior performance compared to state-of-the-art benchmarks.**

I. Introduction

Sensor fusion has recently drawn great research interest in modern autonomous driving. In particular, the fusion of camera and LiDAR has demonstrated promising results in autonomous driving tasks like object detection [1]–[3], and semantic segmentation [4]–[6] and object tracking [7]–[9] according to the leaderboard rankings [10]. Such success is attributed to their complementary nature: a LiDAR provides precise spatial information, while a camera offers dense context information. Accumulative operation-induced factors like physical vibrations and sensor temperature fluctuations can subtly shift their extrinsic parameters. However, the accurate extrinsic transformation is the foundation of LiDAR-camera data fusion.

Currently, the industry standard for LiDAR-camera calibration primarily relies on manual offline tuning using target boards such as checkerboards and polygonal boards. To address such limitation, extensive research has been conducted on the development of automatic extrinsic calibration methods in the driving environment, namely LiDAR-camera online calibration. In traditional approaches, [11]–[13] propose the information theory based methods, which determine the extrinsic parameters by maximizing the similarity transformation. Various studies [14]–[16] estimate extrinsic parameters by identifying and matching features derived from the static pair of LiDAR and RGB camera data or Structure from Motion (SfM). Recently, owing to advancements in deep learning, LiDAR-camera online calibration has demonstrated significant performance improvements. RegNet [17] is the pioneer by integrating feature extraction, feature matching, and global regression into a CNN network. The following works employ various networks and strategies to improve performance [18]–[21]. Unlike prior work, we propose a novel framework for LiDAR-camera online calibration by leveraging the spatial context of cross-modal sensor fusion. Notably, it employs a Transformer-based module to learn the critical interaction feature between correspondence points from camera and LiDAR, as shown in Fig.1. This module enables the utilization of global spatial context, leading to high performance.

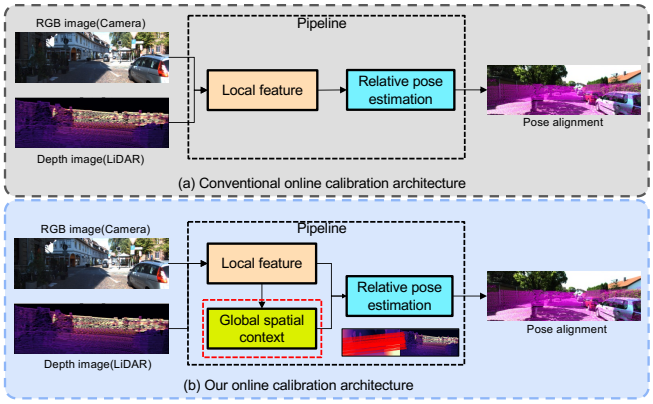

Fig. 1. The pipelines of LiDAR-camera online calibration. (a) Conventional online calibration only utilizes local feature for estimating relative pose. (b) In contrast, our online calibration uses global spatial context (correspondence matching points) for predicting relative pose.

Our framework adopts the encoder-decoder architecture of the Transformer. The encoder, as a self-attention network, extracts local feature from the LiDAR depth image and the camera image. The decoder, as a cross-attention network, estimates the correlating points between the LiDAR depth image and camera image, which uses a specific point query from the image with integrating the local features derived from the encoder. This integration of local feature information from the encoder and spatial information from the decoder benefits the extrinsic prediction network with a richer feature representation.

- **Addressing cross-modality divergence:** We tackle the challenge of cross-modality divergence between two different sensors by proposing a novel learning method that leverages a Transformer network. To achieve more accurate predictions of corresponding points between

[1]Authors from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea. {sj.moon, seb.lee, sungeui}@kaist.ac.kr
[2]Authors from the SAPEON Korea Inc., Seongnam-si, South Korea. {sj.moon, dhe}@sapeon.com
*Prof. Sung-Eui Yoon is a corresponding author.

cameras and LiDAR sensors, we design the learning process to organically incorporate the information from both sensors. By pre-computing the similarity between the query point information from the camera and the ground truth information from LiDAR, our Transformer captures local features and predicted correspondence information in a mutually reinforcing manner.

- **Utilizing spatial information:** Overcoming the absence of spatial information, we train the Transformer decoder's output feature on camera-LiDAR correspondence. This enables the utilization of global spatial context information. We calculate calibration-flow [21], [22] information from the correspondence details, enhancing the incorporation of diverse spatial information. Going beyond the use of spatial information alone, we enhance performance by combining the local feature information from the Transformer encoder with spatial information, surpassing previous research.
- **Unified modality learning:** Through ablation studies, we propose and analyze a method that compares the performance of using only local feature information to learning with a unified modality. Our findings reveal that our proposed approach exhibits performance improvements equivalent to or surpassing those achieved by unified modality learning, providing valuable insights into the calibration process.

To validate the proposed calibration framework, we conduct extensive experiments using the KITTI Raw dataset [23]. The results demonstrate significant improvements in both translation and rotation accuracy, point cloud alignment, and feature matching, affirming the efficacy of our approach.

Our main contributions are summarized as follows: (1) We leverage a Transformer in our calibration framework to tackle cross-modality divergence between LiDAR and camera. (2) We train the network by utilizing the global spatial information from camera-LiDAR correspondence. (3) Our proposed calibration framework shows promising performance enhancements compared to single-modality learning methods, as demonstrated by the results obtained in our experiments.

II. Related work

A. LiDAR-camera calibration

According to the requirement of a target board or not, the existing LiDAR-camera calibration methods are grouped into 1) target-based calibration or 2) target-less calibration. The target-based calibration is commonly used to detect both intrinsic and extrinsic parameters with a target board during the offline manufacturing stage. Several studies simplify the calibration process and improve the precision [24], [25]. In contrast, target-less calibration aims to estimate extrinsic parameters in real-time, also known as online calibration. The mainstream of online calibration includes information theory based methods, feature based methods and learning based methods.

Information theory based methods aim to find the optimized statistical model that maximizes the similarity between the joint histograms of several common attributes, such as reflectivity, intensity, gradient magnitude and orientation extracted from LiDAR and camera [11]–[13], [26]. Information theory methods are highly dependent on selecting appropriate attributes, However, their accuracy is affected by environmental factors such as occlusion and shadows.

Feature-based methods leverage the extracted geometric, semantic, or motion features to establish the correspondences between point clouds and images [14]–[16], [27]. While these methods show promising performance by adopting well-established feature extraction methods, their reliance on representations from LiDAR and camera can lead to misalignment. Meanwhile, LiDAR and camera capture distinct characteristics of the environment and the feature extraction can be affected by noise and occlusion.

Given the growing volume of training data for cameras and LiDAR, along with the possibility of end-to-end inference, recent research is utilizing deep learning to enhance the accuracy of pose parameters in target-less online calibration. Early works leverage the extraction of features from both 2D projected LiDAR images and camera imagery in a bid to predict poses [17], [18], [28]. Despite these strides, they fell short in performance when compared with non-deep learning methodologies and necessitated cumbersome fine-tuning and iterative enhancements to produce acceptable results from CNN-based models. Evolving from these early research methodologies, a method to extract similarity between the features of LiDAR and cameras has been proposed [19], [29]. These papers also explicitly use the similarity relationship between the features of the two sensors for calibration. However, there is still a limitation of losing intermediate features extracted CNN-based backbone network. Contemporary trends in LiDAR-camera calibration have shifted towards addressing the multi-modal problem. This is achieved by predicting the depth of the camera image first, followed by the prediction of pose parameters via feature matching with sparse LiDAR depth images [30], [31].

Unlike previous works, our work is the first to integrate a Transformer encoder-decoder architecture to explicitly utilize both local feature information and global spatial information. We leverage the Transformer network to not only capture local feature attributes in the encoder but also predict corresponding points in the LiDAR depth image that align with the query point in the Transformer decoder. This approach enables us to utilize a greater amount of features when predicting pose parameters, thereby yielding superior results compared to previous works.

B. Correspondence matching using Transformer

Originally designed for natural language processing, Transformer networks [32] are now used in various fields, including image processing. They have proven effective in identifying image correspondences. Transformers excel in image processing due to their ability to consider the context of the entire input. This allows each pixel to be understood within the image's overall structure, aiding in identifying image correspondences [33]–[35].

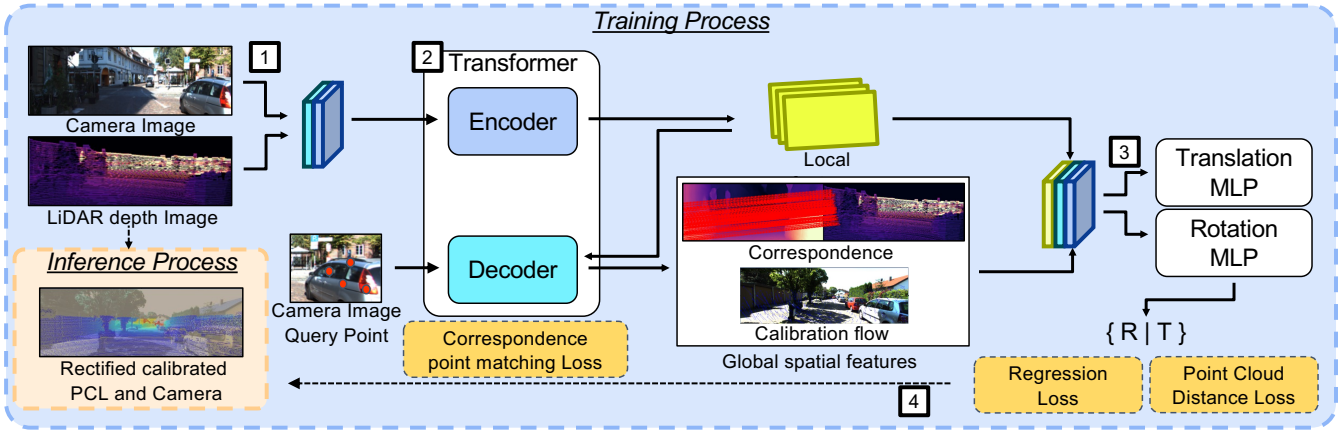

Fig. 2. Overview of our LiDAR-camera online calibration. 1) The input of the Transformer encoder consists of camera data (image form) and LiDAR data (converted into 2D depth information). 2) The transformer encoder processes these inputs and outputs local feature information. This output, along with corresponding point query information, is then inputted into the decoder, which predicts corresponding point pairs between the two images. 3) The regressor network learns relative positioning information using the local feature information (output from the Transformer encoder) and the global spatial information (output from the Transformer decoder). 4) Finally, the point cloud data from the LiDAR is adjusted using the learned relative positioning information.

The COTR methodology [36] uses a Transformer to find correspondence points between different images effectively. The study demonstrates that the Transformers can identify correspondences more quickly and accurately than traditional methods. Inspired by COTR [36], our work utilizes an algorithm within the Transformer decoder to predict corresponding points in two images. This approach is crucial for accurate correspondence identification in LiDAR-camera auto-calibration, as it plays a significant role in calibrating extracted geometrical constraints.

Extending the COTR methodology, our objective was to strategically extract queries within the Transformer framework. Through the preliminary identification of similarity relations between desired query points from camera imagery and mis-calibrated points from LiDAR, we leveraged these as the respective queries and key-value pairs within the Transformer. This methodology not only significantly improved prediction accuracy but also efficiently mitigated the challenge of cross-modality divergence.

III. Method

We propose a novel approach for LiDAR-camera online calibration, as illustrated in Fig.2. In this section, we dive into the details of design components, focusing on the representation strategy in encoder, the architecture of decoder, and the method to aggregate local feature and global spatial context.

A. Preliminaries

For robotic or autonomous systems, we consider a setup comprising both a camera and a LiDAR sensor, crucial for perception and mapping tasks. The relative spatial relationship between those sensors is defined by an initial calibration matrix $M_{init} \in SE(3)$. The initial calibration progressively misaligns during operation due to several factors, such as vibration, temperature changes, and mechanical stress.

To quantify the miscalibration, we employ a decalibration matrix $M_{decal} \in SE(3)$ that delineates the deviations from the initial calibration [37]. Then, a corrected calibration matrix is defined by the composite transformation with the initial calibration and decalibration matrix, *i.e.*, $M_{init}M_{decal}$, indicating the actual transformation between the sensors. We learn a calibration network f_θ to estimate the decalibration matrix M_{decal}. The calibration network f_θ takes an RGB image I and a depth image D as inputs, captured from the camera and LiDAR, estimating the decalibration matrix: $\hat{M}_{decal} := f_\theta(I, D)$. The depth image D is computed by projecting LiDAR points to the image plane with the transformation $M_{init}M_{decal}$ and a camera intrinsic matrix [37].

To train the calibration network with diverse training samples, we randomly generate decalibration matrices M_{decal} as labels. We encourage the calibration network to minimize the discrepancy between the predicted calibration matrix $M_{init}\hat{M}_{decal}$ and the actual calibration matrix $M_{init}M_{decal}$, *i.e.*, $\hat{M}_{decal} \approx M_{decal}$.

B. Representation of local feature and global spatial context

Point cloud 2D-view augmentation: To address the limitations of sparse depth maps resulting from projecting a 3D point cloud onto a 2D view, we employ a bilateral filter [38] on LiDAR data. This filter enhances the depth map density by considering texture information and spatial distance, effectively smoothing the map while preserving edges. The bilateral filtering algorithm demonstrates to be a robust approach for improving the quality of depth information.

Transformer encoder: The Transformer network integrates the dense depth map, represented as D and acquired through bilateral filtering, with the original camera image I, to form the input data. This fusion is mathematically articulated as:

$$\text{Input}_{\text{concat}} = \text{Concat}(D, I), \tag{1}$$

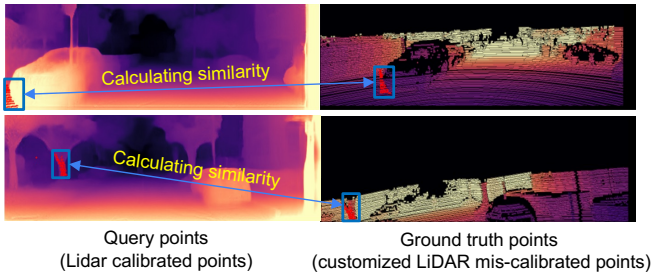

Fig. 3. Extraction of camera query points context. We calculate distance between camera points (u, v, z) and mis-calibrated LiDAR 2D points (u', v', z') and then generate high-similarity query points for the Transformer decoder.

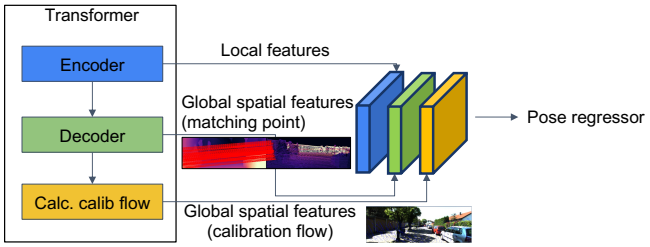

Fig. 4. Fusion module of local feature and global spatial features. Concatenation of local feature, global spatial features (matching point), and global spatial features (calibration flow) as the fused feature for the pose regressor(Translation/Rotation MLP).

where the function Concat$(\cdot)$ denotes the concatenation operation between the two inputs.

Subsequently, the Transformer network employs self-attention on the combined input to extract local feature information, as depicted by the expression:

$$E_{\text{out}} = \text{Self-Attn}\left(\text{Input}_{\text{concat}}\right), \tag{2}$$

where the term Self-Attn$(\cdot)$ references the self-attention mechanism [33], [36], and E_{out} denotes outputs of Transformer encoder. This process enables the Transformer network to effectively extract local features from the combined input, encompassing information from both the LiDAR depth image and the camera image.

Transformer decoder: The Transformer decoder takes the output from the encoder E_{out} as its input and simultaneously utilizes the features of query points extracted from the camera image. The primary objective is to predict corresponding points in the dense depth map for the given camera image query points. As in Fig. 3, we extracted reliable query points with high similarity from both the camera's image and the mis-calibrated LiDAR depth image. The details of query points extraction are as follows:

1) We use the ground truth calibration parameters to transform the LiDAR point cloud into the camera coordinate system, where points $\mathbf{P}(u, v, z)$ are considered as the reference. 2) We empirically select a subset of points, $\mathbf{P}_{\text{sub}} = \{\mathbf{p}_i(u, v, z) \in \mathbf{P}(u, v, z) \mid z_i \leq z_{\text{threshold}}\}$, based on their depth values. 3) We then apply the intended mis-calibration parameters to the LiDAR point cloud and obtain the corresponding points $\mathbf{P}'(u', v', z')$. From this set, we select the subset of points $\mathbf{P}'_{\text{sub}}(u', v', z')$. 4) From the set of corresponding point pairs $(\mathbf{P}_{\text{sub}}(u, v, z), \mathbf{P}'_{\text{sub}}(u', v', z'))$, we select the top K point pairs, $\mathbf{P}_{\text{sub\_topK}}(u, v, z)$ and $\mathbf{P}'_{\text{sub\_topK}}(u', v', z')$, with the smallest Euclidean distance between them. 5) Here, $\mathbf{P}_{\text{sub\_topK}}(u, v, z)$ represents the query points, while $\mathbf{P}'_{\text{sub\_topK}}(u', v', z')$ serves as the ground truth. This allows for reducing the depth error of corresponding points, thereby increasing the accuracy of the prediction.

As we mentioned in Equation 2, E_{out} represents the output from the encoder, and Q_{p} denotes the features of query points extracted from the camera image. The attention mechanism [33] first computes the dot product between the query matrix Q_p and the transpose of the output embeddings matrix E_{out}. This result is scaled by the square root of d_k, where d_k is the dimensionality of the query and key vectors. The combination of these inputs for the attention of decoder is expressed as:

$$\text{Attn} = \text{softmax}\left(\frac{Q_{\text{p}} E_{\text{out}}^T}{\sqrt{d_k}}\right) E_{\text{out}}. \tag{3}$$

The Transformer decoder, through attention and subsequent layers, processes this combined input to predict corresponding points in the dense depth image from camera query points. This process uses the Transformer's capability to focus on relevant information from the encoder output and query points, facilitating the prediction of associated points in the dense depth image.

Local-global aggregation: As shown in Fig.4, we leverage the local feature extracted by the Transformer encoder and the corresponding point pair information predicted by the Transformer decoder. Furthermore, we incorporated the feature of the calibration flow [22] to provide a global spatial context. we define calibration flow which represented as the deviation from camera points (u, v, z) to LiDAR points (u', v', z'), serves as a crucial step in aligning LiDAR and camera data. Contrary to other studies [21], [22] that directly predict calibflow, our approach of predicting corresponding points and calculating deviation allows for a more comprehensive and sophisticated global spatial context.

After aggregating local feature and global spatial context, we employ a Multi-Layer Perceptron (MLP) network to predict calibration parameters. The MLP uses the concatenated local feature, corresponding points, and calibration flow as input. It then estimates translation and rotation adjustments, effectively refining and optimizing local feature and global spatial context.

By combining the strengths of Transformer-based attention mechanisms with MLP regression, our proposed online calibration network demonstrates robustness in learning and predicting intricate relationships between local features, corresponding point pairs, and translation/rotation parameters essential for accurate sensor fusion and calibration. The comprehensive integration of these components contributes to the advancement of sensor calibration methodologies in 3D perception systems.

C. Loss functions

The overall loss function L is defined as a weighted combination of different components, each addressing specific aspects of the calibration framework:

$$L = L_{pos} + L_P + L_C, \tag{4}$$

where, the components are delineated as follows:

- **Regression loss (L_{pos}):** This component encapsulates the regression loss associated with the translation and rotation parameters of MLP network output.
- **Point cloud distance loss (L_P):** This component corresponds to the loss related to the distance between predicted and ground truth point clouds.
- **Correspondence loss (L_C):** This component involves the distance loss associated with the prediction of correspondence points of the Transformer decoder.

Regression loss: The total regression combines the translation and rotation losses:

$$L_{pos} = \lambda_T L_T + \lambda_R L_R, \tag{5}$$

where L_T is translation loss and L_R is rotation loss. λ_T and λ_R denotes weight of translation loss and rotation loss respectively.

For the translation loss L_T, the smooth L1 loss is applied to ensure a stable optimization process. Unlike the L1 loss, the smooth L1 loss addresses the issue of non-uniqueness in the derivative at zero, enhancing training convergence. This loss function exhibits smooth behavior, especially in the vicinity of zero, attributed to the incorporation of the square function.

$$L_T = \text{SmoothL1}(t_{gt}, t_{pred}), \tag{6}$$

where t_{gt} denotes translation vector of ground truth and t_{pred} denotes predicted translation vector.

Concerning the rotation loss L_R, quaternions inherently represent directional information, and using Euclidean distance may not accurately capture the distinction between two quaternions. Therefore, we opt for angular distance to quantify the dissimilarity between quaternions, expressed as $L_R = D_a(q_{\text{gt}}, q_{\text{pred}})$, where q_{gt} denotes the ground truth quaternions, q_{pred} is the predicted quaternions, and D_a represents the angular distance function.

Point cloud distance loss: The Point Cloud Distance Loss ($\lambda_P L_P$) is defined as the dissimilarity between the ground truth LiDAR point cloud gt and the point cloud obtained by re-projecting the predicted point cloud through the inverse transformation matrix of the predicted extrinsic parameters. Mathematically, this loss term can be expressed as:

$$L_P = \lambda_P \cdot \|gt - P(\hat{P}(gt))\|, \tag{7}$$

where P represents the projection function, $\hat{P}$ is the re-projection function using the inverse transformation matrix of the predicted extrinsic parameters, and λ_P is the associated weight for this loss term. This formulation captures the distance between the ground truth and the re-projected predicted point clouds, contributing to the overall calibration framework loss.

Correspondence loss: Building upon prior research [36], we utilize the following expressions for the correspondence matching distance:

$$L_{\text{corr}} = \|x_0 - F_\Gamma(x|I, D)\|_2^2 \tag{8}$$

$$L_{\text{cycle}} = \|x - F_\Gamma(F_\Gamma(x|I, D)|I, D)\|_2^2, \tag{9}$$

where L_{corr} quantifies errors in correspondence estimation, I, D represents camera image and LiDAR mis-calibrated depth image respectively, x, x_0 denotes query point and ground truth of query point of camera image (image I) respectively, F_Γ represents the re-projection function. Similarly, L_{cycle} enforces cycle-consistency among correspondences. Combining these terms, the correspondence matching distance is expressed as:

$$L_C = \lambda_C(L_{\text{corr}} + L_{\text{cycle}}). \tag{10}$$

The weights λ_T, λ_R, λ_P, and λ_C enable fine-tuning the emphasis placed on each component during the optimization process. The comprehensive combination of these loss components contributes to the effective training of the calibration framework.

IV. EXPERIMENTS

A. Dataset preparation

Following [31], [37], we train and evaluate our model on a subset of the KITTI Raw 09_26 dataset, consisting of 24,000 samples. To simulate mis-calibration, we randomly modify translation with ± 0.25 meters and rotation with ± 10 degrees. The performance test is then conducted on divergent test sets including:

- **Test set No.1 (T1):** KITTI raw 09_26 (6000 samples);
- **Test set No.2 (T2):** KITTI raw 09_30 (1600 samples);
- **Test set No.3 (T3):** nuScenes v1.0-test (6008 samples).

T2 assesses the generalization capability and robustness of our model across different scenes within the KITTI raw dataset. Additionally, T3 verifies the robustness of our model on the nuScenes dataset. These test sets consistently indicate that our model demonstrates stable performance across diverse conditions and environments.

Fig. 5. visualizes the qualitative results and ground truth values from the KITTI dataset. It also displays the intermediate inference of correspondence points. The improvement in calibration parameter inference results is significantly attributed to these correspondence points, as inferred by the Transformer decoder.

B. Training details

We train our model on a single NVIDIA RTX 3090 GPU with a batch size of 10. The input image sizes are resized to 512 $\times$ 256. The initial learning rate is set to 1e-4, and it decays by half every 30 epochs. We utilize the AdamW optimizer [39] for training. The loss weights for translation, rotation, point cloud distance, and correspondence matching distance are set as 2.0, 1.0, 0.5, and 1.0, respectively. The total training epoch is set to 250.

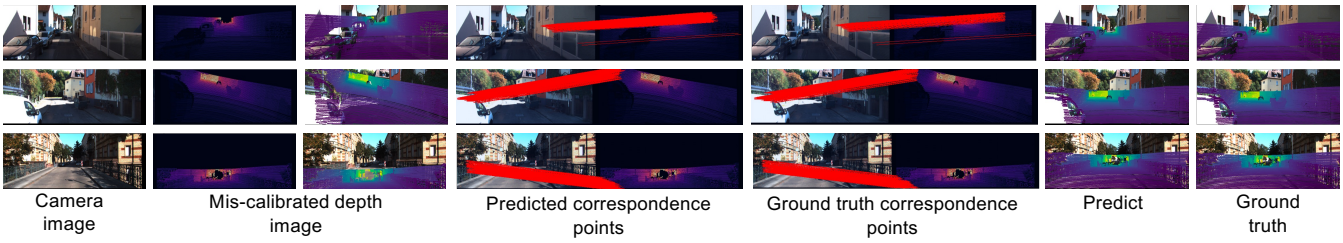

Fig. 5. Qualitative results of predicted correspondence points and results.

TABLE I
CALIBRATION RESULTS (T1)

Method	Translation(cm)				Rotation(deg)			
	mean	X	Y	Z	mean	Roll	Pitch	Yaw
RegNet [17]	6	7	7	4	0.28	0.24	0.25	0.36
Calibnet [28]	7.82	12.10	3.49	7.87	0.410	0.150	0.900	0.180
CalibRCNN [19]	5.30	6.20	4.30	5.40	0.428	0.199	0.640	0.446
CalibDNN [20]	5.07	3.80	1.80	9.60	0.210	0.110	**0.350**	0.180
CalNet [37]	3.03	3.65	**1.63**	3.80	0.200	**0.100**	0.380	0.120
Ours	**2.91**	**1.64**	5.79	**1.30**	**0.178**	0.105	0.397	**0.034**

TABLE II
CALIBRATION RESULTS (T2)

Method	Translation(cm)				Rotation(deg)			
	mean	X	Y	Z	mean	Roll	Pitch	Yaw
CalibRCNN [19]	5.73	7.80	3.20	6.20	0.97	0.21	2.21	0.50
CalibDNN [20]	6.10	5.50	3.20	9.60	0.450	0.15	0.99	0.20
CalNet [37]	4.95	4.84	2.59	7.42	0.400	0.15	0.91	**0.15**
CalibDepth [31]	4.75	6.66	**1.12**	6.48	0.348	0.180	0.682	0.181
Ours	**3.81**	**3.24**	3.93	**4.25**	**0.240**	**0.086**	**0.361**	0.280

C. Metrics

To measure the accuracy of the translation vector, we employ the Euclidean distance between vectors, defining the absolute error as:

$$E_t = \|\mathbf{t}_{\text{pred}} - \mathbf{t}_{\text{gt}}\|_2, \tag{11}$$

where, $\|\cdot\|_2$ denotes the 2-norm of a vector. Furthermore, we evaluate the absolute error of the translation vector separately in the X, Y, and Z directions.

Rotation representation is facilitated through quaternions, serving as directional indicators. Utilizing quaternion angle distance, we delineate disparities between quaternion representations. For the assessment of angle errors in the extrinsic rotation matrix across three dimensions, we transform the rotation matrix into Euler angles. These combined metrics provide a comprehensive assessment of the auto-calibration performance.

D. Experimental results

The quantitative results of T1 are shown in Table I. We compares the accuracy results between proposed model(ours) and the auto-calibration algorithms proposed in prior research based on deep learning models.

Table II presents the accuracy comparison results T2. The results of T2 indicate that our approach achieves higher accuracy in translation error(mean), with a precision of 0.94 cm, and in rotation angle(mean) with a precision of 0.108 degrees than CalibDepth [31].

TABLE III
CALIBRATION RESULTS (T3:NUSCENES)

Method	Translation(cm)				Rotation(deg)			
	mean	X	Y	Z	mean	Roll	Pitch	Yaw
CalibDepth [31]	12.78	7.57	**21.13**	9.64	7.02	8.88	4.69	7.49
Ours	**12.55**	**6.29**	23.28	**8.08**	**4.19**	**5.92**	**4.43**	**2.21**

Indeed, Table III determines the robustness of the model through the nuScenes dataset. Table III measures the inference performance of the nuScenes data through the pretrained model of CalibDepth [31] and our proposed model. Through these experiments, it was confirmed that the inference performance on different datasets is also superior to other models.

E. Ablation study

We conduct the ablation study to investigate two aspects: **1) Discerning the impact of Transformer decoder data on performance:** we aim to understand how the inclusion of data from the Transformer decoder influences performance. **2) Evaluating the performance impact of the Transformer network in resolving multi-modality issues:** our goal is to assess how the Transformer network contributes to addressing challenges related to multi-modality. The ablation study results are shown in Table IV.

In case 1, we evaluated the model performance with only local feature encoder. In case 2, we employ both global and local features while using LiDAR range-view depth images directly without image depth estimation. The improved performance demonstrates the importance of utilizing both global and local features in the model. Additionally, we adopt the image depth estimation, and it turns out the translation performance degrades comparing to ours. In other words, the LiDAR range-view depth images can replace the image depth estimation without performance loss.

F. Limitation analysis

While our Transformer-based method outperforms existing CNN-based methods in overall performance, it exhibits worse in predicting Y-axis translations. To explain this discrepancy, we conducted a thorough analysis including the characteristics of the input images and the properties of Transformers. We first observed that images in driving

TABLE IV
ABLATION STUDIES RESULTS

Case	Image Depth Estimation	Transformer Encoder (local feature)	Transformer Decoder (global spatial feature)	Translation (E_t)	Rotation (E_R)
1		✓		4.35	0.40
2 (ours)		✓	✓	**4.18**	0.39
3	✓	✓	✓	4.93	**0.37**

scenarios have unique inherent characteristics: the variance of pixel responses along the X-axis is greater than along the Y-axis, which is also mentioned in [40]. The self-attention network, functioning as a low-pass filter, enhances the characteristics of local features but also has the potential to distort outlier information [41]. These two factors combined lead to relatively larger Y-axis translation errors. Notably, this limitation is a significant finding revealed in our experimental results, and presents a potential research opportunity.

V. CONCLUSION

By employing a Transformer encoder and decoder, our method can select and learn both local features and global spatial context, effectively enhancing learning performance. This approach opens new possibilities in deep learning-based calibration automation technology, particularly in achieving more accurate calibration results by considering both local features and global spatial context. Moreover, the Transformer-based algorithm improves proving valuable for online calibration scenarios. Future work will apply this method to more robust network that reflects the characteristics of image features and explore various strategies to boost performance. These efforts will undoubtedly contribute to the advancement of calibration automation technology, reinforcing the importance and potential of this research.

ACKNOWLEDGEMENT

This work was supported by Korea Evaluation Institute of Industrial Technology (KEIT) grant funded by the Korea government (MOTIE) (No.20017851, Development of scalable AI Accelerator processor with 200TOPS for Real-time AI Inference). In addition, this project received support from Institute of Information & communications Technology Planning & Evaluation (IITP) and National Research Foundation of Korea (NRF), funded by the Korea government (MSIT) (RS-2023-00237965 and RS-2023-00208506(2024)). Prof. Sung-Eui Yoon is a corresponding author.

REFERENCES

[1] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3d proposal generation and object detection from view aggregation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–8.

[2] S. Pang, D. Morris, and H. Radha, "Clocs: Camera-lidar object candidates fusion for 3d object detection," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 10 386–10 393.

[3] Y. Li, A. W. Yu, T. Meng, B. Caine, J. Ngiam, D. Peng, J. Shen, Y. Lu, D. Zhou, Q. V. Le, *et al.*, "Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 182–17 191.

[4] Z. Zhuang, R. Li, K. Jia, Q. Wang, Y. Li, and M. Tan, "Perception-aware multi-sensor fusion for 3d lidar semantic segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 16 280–16 290.

[5] L. Zhao, H. Zhou, X. Zhu, X. Song, H. Li, and W. Tao, "Lif-seg: Lidar and camera image fusion for 3d lidar semantic segmentation," *IEEE Transactions on Multimedia*, 2023.

[6] J. Cen, S. Zhang, Y. Pei, K. Li, H. Zheng, M. Luo, Y. Zhang, and Q. Chen, "Cmdfusion: Bidirectional fusion network with cross-modality knowledge distillation for lidar semantic segmentation," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 771–778, 2023.

[7] K. Huang and Q. Hao, "Joint multi-object detection and tracking with camera-lidar fusion for autonomous driving," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 6983–6989.

[8] X. Wang, C. Fu, Z. Li, Y. Lai, and J. He, "Deepfusionmot: A 3d multi-object tracking framework based on camera-lidar fusion with deep association," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8260–8267, 2022.

[9] C. Zhang, C. Zhang, Y. Guo, L. Chen, and M. Happold, "Motiontrack: End-to-end transformer-based multi-object tracking with lidar-camera fusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 151–160.

[10] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.

[11] G. Pandey, J. McBride, S. Savarese, and R. Eustice, "Automatic targetless extrinsic calibration of a 3d lidar and camera by maximizing mutual information," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 26, no. 1, 2012, pp. 2053–2059.

[12] Z. Taylor, J. Nieto, and D. Johnson, "Automatic calibration of multi-modal sensor systems using a gradient orientation measure," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1293–1300.

[13] C. Guindel, J. Beltrán, D. Martín, and F. García, "Automatic extrinsic calibration for lidar-stereo vehicle sensor setups," in *2017 IEEE 20th international conference on intelligent transportation systems (ITSC)*. IEEE, 2017, pp. 1–6.

[14] 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.

[15] R. Ishikawa, T. Oishi, and K. Ikeuchi, "Lidar and camera calibration using motions estimated by sensor fusion odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 7342–7349.

[16] C. Park, P. Moghadam, S. Kim, S. Sridharan, and C. Fookes, "Spatiotemporal camera-lidar calibration: A targetless and structureless approach," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1556–1563, 2020.

[17] N. Schneider, F. Piewak, C. Stiller, and U. Franke, "Regnet: Multimodal sensor registration using deep neural networks," in *2017 IEEE intelligent vehicles symposium (IV)*. IEEE, 2017, pp. 1803–1810.

[18] K. Yuan, Z. Guo, and Z. J. Wang, "Rggnet: Tolerance aware lidar-camera online calibration with geometric deep learning and generative model," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6956–6963, 2020.

[19] J. Shi, Z. Zhu, J. Zhang, R. Liu, Z. Wang, S. Chen, and H. Liu, "Calibrcnn: Calibrating camera and lidar by recurrent convolutional neural network and geometric constraints," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 10 197–10 202.

[20] G. Zhao, J. Hu, S. You, and C.-C. J. Kuo, "Calibdnn: multimodal sensor calibration for perception using deep neural networks," in *Signal Processing, Sensor/Information Fusion, and Target Recognition XXX*, vol. 11756. SPIE, 2021, pp. 324–335.

[21] X. Jing, X. Ding, R. Xiong, H. Deng, and Y. Wang, "Dxq-net: differentiable lidar-camera extrinsic calibration using quality-aware flow," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 6235–6241.

[22] X. Lv, S. Wang, and D. Ye, "Cfnet: Lidar-camera registration using calibration flow network," *Sensors*, vol. 21, no. 23, p. 8112, 2021.

[23] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[24] F. Itami and T. Yamazaki, "An improved method for the calibration of a 2-d lidar with respect to a camera by using a checkerboard target," *IEEE Sensors Journal*, vol. 20, no. 14, pp. 7906–7917, 2020.

[25] J. Domhof, J. F. Kooij, and D. M. Gavrila, "A joint extrinsic calibration tool for radar, camera and lidar," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 3, pp. 571–582, 2021.

[26] Z. Taylor, J. Nieto, and D. Johnson, "Multi-modal sensor calibration using a gradient orientation measure," *Journal of Field Robotics*, vol. 32, no. 5, pp. 675–695, 2015.

[27] H. Yu, W. Zhen, W. Yang, and S. Scherer, "Line-based 2-d–3-d registration and camera localization in structured environments," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 11, pp. 8962–8972, 2020.

[28] G. Iyer, R. K. Ram, J. K. Murthy, and K. M. Krishna, "Calibnet: Geometrically supervised extrinsic calibration using 3d spatial transformer networks," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1110–1117.

[29] X. Lv, B. Wang, Z. Dou, D. Ye, and S. Wang, "Lccnet: Lidar and camera self-calibration using cost volume network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2894–2901.

[30] A. Zhu, Y. Xiao, C. Liu, and Z. Cao, "Robust lidar-camera alignment with modality adapted local-to-global representation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 1, pp. 59–73, 2022.

[31] J. Zhu, J. Xue, and P. Zhang, "Calibdepth: Unifying depth map representation for iterative lidar-camera online calibration," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 726–733.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[33] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[34] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 4938–4947.

[35] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "Loftr: Detector-free local feature matching with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8922–8931.

[36] W. Jiang, E. Trulls, J. Hosang, A. Tagliasacchi, and K. M. Yi, "Cotr: Correspondence transformer for matching across images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6207–6217.

[37] Y. Fu, W. Li, S. Fan, Y. Jiang, and H. Bai, "Cal-net: Conditional attention lightweight network for in-orbit landslide detection," *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[38] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*. IEEE, 1998, pp. 839–846.

[39] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[40] H. Zhou, Z. Ge, Z. Li, and X. Zhang, "Matrixvt: Efficient multi-camera to bev transformation for 3d perception," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 8548–8557.

[41] Z. Wang, H. Luo, P. Wang, F. Ding, F. Wang, and H. Li, "Vtc-lfc: Vision transformer compression with low-frequency components," *Advances in Neural Information Processing Systems*, vol. 35, pp. 13 974–13 988, 2022.