2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Cross-Modal Registration Using Adaptive Modeling in Infrastructure-based Vehicle Localization*

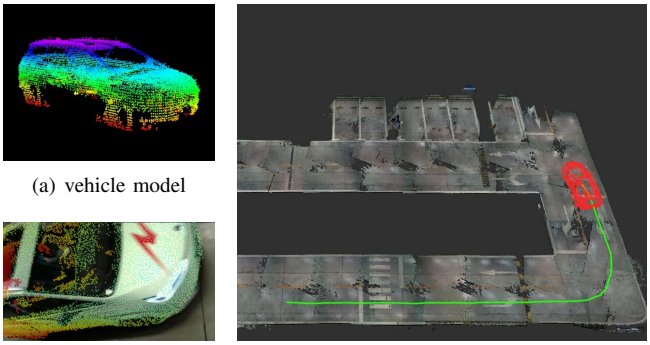
Fei Wang, Yuesheng He, Hanyang Zhuang, Chenxi Yang, Ming Yang

Abstract— **Infrastructure-based vehicle localization, in comparison to single-agent approaches, offers several advantages including reduced system cost, extended perception range, enhanced data fusion capabilities, and energy savings. Many conventional approaches impose limitations on the types of objects due to the need for specific object-end modifications, such as applying perceptual markers like color-labeled plates and reflective balls. LiDAR presents a solution in terms of object arbitrariness, as it addresses the challenges of feature-free object modeling and continuous registration. However, achieving complete environmental coverage with LiDAR remains prohibitively expensive, particularly in extensive areas. Hence, this study proposes a cross-modal localization approach using adaptive modeling, employing LiDAR for object modeling and cost-effective sensor cameras for object tracking through image-point-cloud registration. Accurate correspondence between the model and observation can be estimated in real-time. The experiments are conducted in a typical scenario that requires adaptive modeling: Autonomous Valet Parking (AVP). Results demonstrate that the proposed system achieves comparable performance with significantly reduced system costs, highlighting its potential for large-scale deployment.**

I. Introduction

Localization and navigation are crucial functions for mobile robots, such as intelligent vehicles. Many indoor localization scenarios, such as Autonomous Valet Parking (AVP), require high-precision localization without relying on the Global Navigation Satellite System (GNSS). Agent-end Simultaneous Localization and Mapping (SLAM) and the map-matching-based pose estimation methods achieve high precision for single-agent indoor localization. However, they often require the installation of complex and expensive sensors on mobile robots, and the cost increases linearly with the number of agents. Infrastructure-based localization methods have the advantage of not requiring extensive modifications and reducing costs as the number of localization targets increases, making them an ideal indoor localization method.

The traditional infrastructure-based localization method involves installing Ultra-WideBand (UWB) receivers on the robot or utilizing perceptual markers such as color-labeled plates[16] or infrared-red reflective targets[19]. These methods can only localize specific objects that have these markers installed. Thus, the installation and maintenance of a large number of markers can be costly, and the issue of color-labeled plates potentially getting occluded as the robot moves is a significant concern. Infrared reflective markers are less prone to being occluded, but infrared cameras can be quite expensive. Therefore, these methods are unsuitable for scenarios with many agents, such as AVP.



(a) vehicle model

(b) cross-modal registration result

(c) vehicle localization result

Fig. 1. (a) The vehicle point cloud model built by LiDAR. (b) Cross-Modal registration result of vehicle model and camera image. (c) The vehicle model and its trajectory (green curve) in the parking lot.

The methods using infrastructure-based cameras[13], [10], [11], [1], [14], [20] can realize vehicle localization using ground geometry or multi-view geometry. However, due to the fact that the vehicle in images is often incomplete and the challenge of directly estimating vehicle poses from images, the accuracy of these methods is not sufficient to meet the requirements of AVP.

Our previous work implemented a infrastructure-based two-stage vehicle localization system using solid-state LiDARs [5]. In the first stage, two LiDARs set at the modeling area scan the entering vehicle to provide the object model. In the second stage, multiple LiDARs installed around the lot keep tracking the target within the lot area. That approach achieved high localization accuracy through adaptive modeling, but the high cost due to the large LiDAR numbers for full site coverage significantly restricted further large-scale deployment.

In this study, we proposed a low-cost solution to combine LiDAR's advantage of rich information and cameras' low cost in scaled deployment. Two LiDARs are kept for adaptive modeling, while all the other LiDARs are replaced with cameras to keep the full site coverage at a significantly lower cost. In such a system hardware setting, how to align modeled point clouds with real-time image observation presents a challenge. To address this issue, we proposed

This work was supported by National Natural Science Foundation of China (62203294 / U22A20100)

Fei Wang, Yuesheng He, Chenxi Yang, Ming Yang are with the Department of Automation, Shanghai Jiao Tong University, Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, 200240, China (email: heyuesh@sjtu.edu.cn).

Hanyang Zhuang is with the University of Michigan - Shanghai Jiao Tong University Joint Institute. Shanghai Jiao Tong University, Shanghai, 200240, China.

a cross-modal registration network for achieving real-time point cloud and image cross-modal registration localization. And we validated its localization accuracy in our indoor AVP parking lot. Fig.1 illustrates the vehicle point cloud model, cross-modal registration result, and vehicle localization result. The contribution of this paper is summarized as follows:

- An infrastructure-based localization system has been proposed, which uses a small number of LiDARs and a large number of cameras to significantly reduce costs while maintaining accuracy.
- A novel cross-modal registration method for point clouds and images is proposed, which can estimate the correspondence between 3D and 2D points in real-time.

II. Related Works

Infrastructure-based localization methods are widely used for indoor localization. Based on the different sensors they use, we divide them into two classes. And we also review the methods of cross-modal registration.

A. Camera-Infrastructure-based Method

Infrastructure-based localization methods using cameras have received attention in recent years, due to the low cost of cameras. The main idea of them is to first detect the localization object and then proceed with its position.

[13] and [14] presented segmenting the vehicle by subtracting the exponential time-smoothed background image from the original image and estimating the object's position using multi-view geometry, which achieves decimetre-level accuracy. [11] and [1] used motion history images (MHI)[4] to segment the vehicle and estimate the location of the vehicles by ground geometry. To overcome the problem of lacking a stable localization center, [10] uses motion history images to detect the vehicle, and then GrabCut[22] algorithm is used for vehicle object segmentation and finally the localization center is defined as the ground position below the center of the front license plate. After the emergence of deep learning, [20] introduces a deep learning algorithm (YOLOv4 [3]) to detect vehicles in images, and the localization center for this method is defined as the middle point of the bottom boundary of the extracted bounding box.

The key to these methods is to detect the vehicle from the camera's field of view, but is limited by the fact that the camera does not always capture the complete vehicle, and due to the camera angle, the camera may not be able to capture the center of localization as defined in [10]. Therefore, further exploration is needed on how to improve localization accuracy and robustness.

B. LiDAR-Infrastructure-based Method

Infrastructure-based localization methods using LiDAR can capture rich geometric information. Therefore, after calibration, a corresponding point cloud map can be obtained directly. The main idea of them is to identify vehicles from the point cloud map.

[15] chose 2D LiDAR to form the sensor network. It uses the Hough Transform and the Random Sample Consensus (RANSAC) [12] algorithm to detect the four wheels of the vehicle and uses the geometric center of the vehicle as the localization center. This actually defines the vehicle model in advance. The solid-state LiDAR is the sensor for [5]. The method sets up a fixed modeling area and modeling sensors so that when a vehicle drives through the area, a point cloud model of the vehicle is obtained, and the localization center of the vehicle is automatically annotated. Then the vehicle pose is obtained by using the Iterative Closest Point (ICP) [2] algorithm to register the vehicle model and the sensor point cloud. Both methods achieve highly accurate localization through model and observation matching. However, the cost of sensors limits further expansion.

C. Cross-Modal Registration Methods

Due to the advantages of combining the robustness of LiDAR point clouds with the low cost of cameras, cross-modal registration methods have also garnered significant attention. Cross-modal localization estimates the camera pose in prior LiDAR maps by matching the 3D point cloud and the 2D camera image. [24] achieves localization in point cloud maps using a monocular camera, which firstly extracts ORB feature points from images to generate a point cloud, and then uses semantically-constrained ICP registration orb feature point clouds and point cloud maps. [18] converts the match between point clouds and images into a classification problem for which it proposes a network that estimates the camera pose by predicting whether a point is in the camera's field of view or not.

Another branch of cross-modal localization is similar to the direct method in SLAM, with its direct finding of the correspondence between the point clouds and the images. [6] and [7] are proposed to first generate a synthetic depth image by projecting a LiDAR point cloud with a coarse initial pose into 2D space, and then regressing the camera pose using the optical flow network PWC-Net. [9] uses the optical flow network RAFT to improve localization accuracy, but it comes at the cost of poorer real-time performance.

III. Method

As shown in Fig.2, this system initially receives LiDAR point cloud to create a vehicle model. Once modeling is complete, for each frame in localization, the system selects appropriate camera images capturing the vehicle based on camera extrinsic and the vehicle pose predicted by the Extended Kalman Filter. Next, it inputs the vehicle model and the vehicle image into a cross-modal registration network to obtain the vehicle's pose in camera view. Following a pose transformation, the vehicle's pose is converted into world coordinate system. Finally, an Extended Kalman Filter is employed to optimize the vehicle's pose.

The system includes five modules:

- Vehicle Modeling Module: as the vehicle travels through the modeling area, the module models the vehicle.
- Camera Switching Module: the module selects an appropriate vehicle image based on the vehicle's pose predicted by EKF.

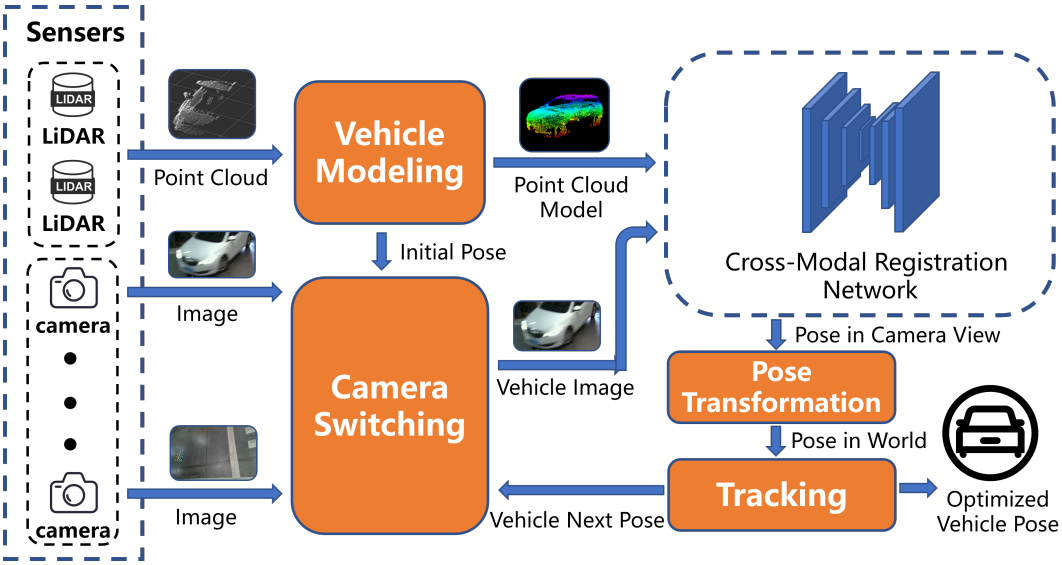


Fig. 2. system overview. This system obtains vehicle point cloud models through LiDAR, and then uses a cross-modal registration network to register the vehicle model with vehicle images, resulting in the current vehicle pose. Finally, an extended Kalman Filter is employed to estimate and optimize the pose.

- Cross-Modal Registration Module: for the vehicle model and vehicle images, the module utilizes a neural network to align them.
- Pose Transformation Module: as the pose obtained after registration is in the camera coordinate system, the module converts it to the world coordinate system.
- Tracking Module: the vehicle poses are optimized by the tracking module.

A. Vehicle Modeling Module

The vehicle point cloud model serves as a crucial reference for subsequent cross-modal registration. An area where the LiDARs are installed is marked out specifically for modeling purposes. When the vehicle travels through the modeling area, a ground segmentation is used to obtain a point cloud belonging to the vehicle. Subsequently, by using the ICP algorithm to align point clouds from different frames, the complete vehicle point cloud model is obtained.

The localization center also need to be annotated on the vehicle model, and for convenience, the geometric center is labeled as localization center.

B. Camera Switching Module

The module selects an appropriate camera for localization based on the camera extrinsic and the vehicle pose predicted by the tracking module. It does so through 3 metrics, including the intersection area of the vehicle's 2D bounding box and the camera's coverage polygon, the distance from the vehicle's localization center to the camera, and the proportion of the image occupied by the vehicle, as shown in Fig.3. And due to the distance between the vehicle and camera is secondary switching metrics, we map it to $[0, 1]$. Consequently, the criterion for camera switching can be formulated as:

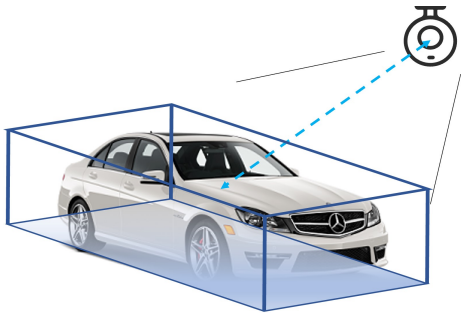


Fig. 3. Camera switching metrics. It includes three metrics, intersection area, distance, and image proportion.

$$S = S_{area} - \alpha * (\frac{2}{1 + e^{-\|t_{cam} - t_{veh}\|}} - 1) + \beta f_{image} \quad (1)$$

where S_{area} is the intersection area, t_{cam} and t_{veh} are the camera position and the vehicle position respectively, f_{image} is the proportion of the image occupied by the vehicle. For efficient computation, the proportion f_{image} is estimated by projecting the vehicle's 3D bounding box onto the image and calculating the proportion it occupies in the image. α is 4 and β is 1 in our experiment.

C. Cross-Modal Registration Module

The structure of the proposed Cross-Modal Registration Network is shown in Fig.4. The basic network structure is modified from the GMFlow[23] network and I2D-Loc[9]. The network regards depth-RGB correspondence estimation as a global matching problem. Considering the difference between depth and RGB image, the encoders use the same architecture but do not share weight parameters.

Since the camera captures the vehicle only from one view, the vehicle point cloud model needs to be converted to a depth image of the camera's view. The process of projecting

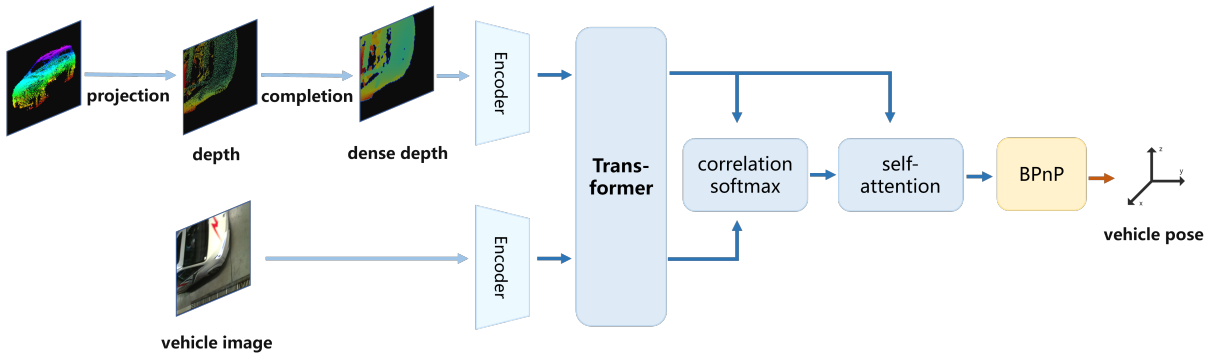


Fig. 4. Network structure of cross-modal registration. This network first converts the point cloud into a depth, then performs depth completion, estimates correspondences between the depth and RGB image, and finally calculates the pose using Backpropagating Perspective-n-Point (BPnP).

a 3D vehicle model into 2D space can be described by a perspective projection model. The process is implemented based on the pinhole camera model. Its definition is

$$[u, v, 1]^T = \boldsymbol{K} \cdot \boldsymbol{T} \cdot [x_{veh}, y_{veh}, z_{veh}, 1]^T \tag{2}$$

where $[x_{veh}, y_{veh}, z_{veh}, 1]^T$ is the point of the vehicle model, $[u, v, 1]^T$ is the point on 2D image plane. $\boldsymbol{K}$ is camera intrinsic and $\boldsymbol{T}$ is camera extrinsic, both of which are obtained in advance.

Due to the sparsity of the vehicle point cloud model, occluded 3D points might be visible on the projected depth. To remove occluded points, [21] is employed. It calculated the angle θ between the camera center and all other valid points in the $N \times N$ neighborhood. If the minimum angle θ_{min} is greater than a fixed threshold, the point is marked as visible. After the depth image is generated, method[17] is used to densify the sparse depth.

At the end of the structure, a BPnP[8] module is used to estimate the pose from the correspondences and it can compute the gradient of the pose estimation and optimize the parameters of the network by back-propagation.

For the network supervision, the average Endpoint Error (EPE) is employed to evaluate the error in the estimated correspondences:

$$L_{epe} = \frac{\sum \|F_{gt}(u, v) - F_{pre}(u, v)\|_2 \times mask_{uv}}{\sum mask_{uv}} \tag{3}$$

The reprojection error is applied to evaluate the accuracy of the estimated pose:

$$L_{reproj} = \sum_{i}^{N} \|p_i - KTP_i\|_2 \tag{4}$$

where K is intrinsic and T is estimated pose.

The zero error is used to eliminate the correspondence between the zero areas of the depth and RGB image:

$$L_{zero} = \frac{\sum \|F_{pre}(u, v)\|_2 \times zeromask_{uv}}{\sum zeromask_{uv}} \tag{5}$$

The final loss is composed of the weighted sum of the three losses, as

$$L = L_{epe} + \gamma L_{reproj} + \delta L_{zero} \tag{6}$$

where γ is 10 and δ is 1 in our experiments.

D. Pose Transformation Module

By default, the Perspective-n-Points (PnP) algorithm is employed to estimate the rotation and translation of the camera. But, in our system, the camera is fixed on the infrastructure and the vehicle is movable. Thus, we need to translate the rotation and translation of the camera estimated by the PnP algorithm to the vehicle motion.

The motion of the camera and the motion of the vehicle can be described by the following equations, respectively:

$$P'_{cam} = T_{cam} P_{cam} \tag{7}$$

$$P'_{veh} = T_{veh} P_{veh} \tag{8}$$

where T_{cam} is the translation matrix estimated by PnP algorithm and T_{veh} describe the motion of vehicle.

In addition, whether considering camera motion or vehicle motion, the relative pose between the camera and the vehicle remains unchanged:

$$P'_{cam} P_{veh} = P_{cam} P'_{veh} \tag{9}$$

Thus, bring eq. 9 into eq. 7 and eq. 8 leads to the transformation equation for transforming camera motion to vehicle motion:

$$T_{veh} = P_{cam}^{-1} T_{cam} P_{cam} \tag{10}$$

E. Tracking Module

An Extended Kalman Filter (EKF) is adopted for tracking, which reduces the localization noise, optimizes pose and estimates the vehicle speed. The vehicle state aims to estimate is

$$[x, y, \theta, v, w, a] \tag{11}$$

representing vehicle position and its yaw angle, vehicle speed, angular velocity, and its acceleration. Then, the standard procedure of EKF propagation procedure is applied to update the vehicle state.

TABLE I

SYSTEM PERFORMANCE COMPARISON AT FOUR DIFFERENT SCENARIOS

Method	01		02		03		04	
	Rot.[°]	Trans.[m]	Rot.[°]	Trans.[m]	Rot.[°]	Trans.[m]	Rot.[°]	Trans.[m]
I2D-Loc(vehicle01)	1.322	0.058	1.262	0.074	1.262	0.049	1.541	**0.047**
ours(vehicle01)	**0.507**	**0.047**	**0.764**	**0.069**	**0.918**	**0.038**	**1.273**	0.049
I2D-Loc(vehicle02)	fail	fail	2.713	0.0833	fail	fail	fail	fail
ours(vehicle02)	**1.265**	**0.041**	**1.912**	**0.048**	**2.492**	**0.0849**	**2.376**	**0.089**

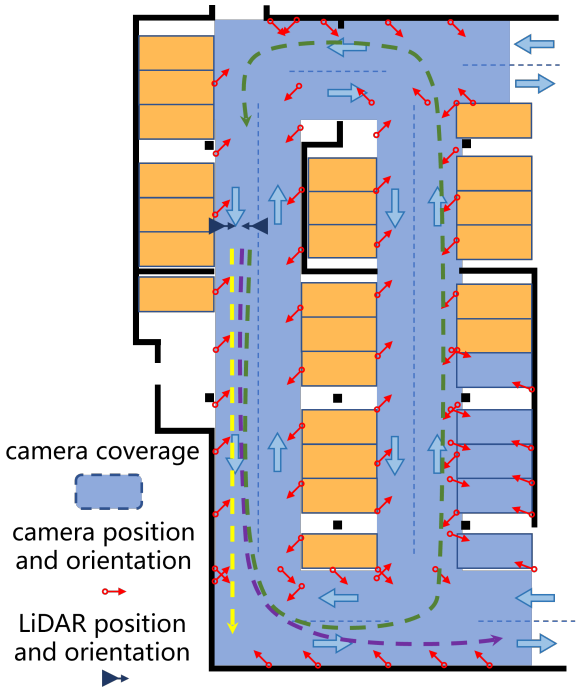


Fig. 5. The experiment site and experiment scenarios. The figure shows the positions, orientations of sensors, and the experiment scenarios, including straight (yellow curve), straight and then turn (purple curve), and circle (green curve).

(a) Vehicle01 (b) Vehicle02

Fig. 6. Experiment vehicle. (a) is a vehicle with SICK nav350. (b) is a regular vehicle without a localization system.

IV. EXPERIMENT

A. Experiment Setup

For our proposed system, we equip an indoor parking lot with both LiDARs and cameras on infrastructure and evaluate the system in the parking lot. As shown in Fig.5, the parking lot is equipped with 2 solid-state LiDARs for vehicle modeling and 65 cameras along the driveway for vehicle localization. The total area of the experimental site in our parking lot is about 900 m^2 and the total length of the lanes is about 130 m.

We use two different vehicles to collect data for evaluating the proposed system. $vehicle01$ (as shown in Fig.6(a)) is a test vehicle equipped with a single-line LiDAR (SICK nav350), and $vehicle02$ (as shown in Fig.6(b)) is a regular vehicle without a self-vehicle localization system. The ground truth localization of $vehicle01$ is provided by a LiDAR localization system based on retroreflective markers, while the ground truth localization of $vehicle02$ is provided by method [5]. For each vehicle, we collected data for four different traveling distances to evaluate the performance of the system in simple and complex scenarios.

For the training of the neural network, the training data is generated by combining [5] with a camera network. Due to time and experimental constraints, we conduct training on a limited number of objects.

Furthermore, the Average Pose Error (APE) is used as a quantitative evaluation metric for the experiment. APE is measured in meters and degrees, and a smaller value indicates better accuracy. All sensors are connected to a computer via Ethernet cables and all experiments are performed on a computer with an AMD R9-7950HX and RTX 3090 on Ubuntu 18.04 LTS(64-bit).

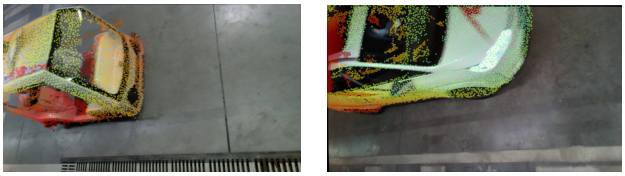
B. Quantitative Experiments of Localization System

As depicted in Fig.5, we evaluated the system's performance using four distinct scenarios collected from two separate vehicles, resulting in a total of eight sequences. These scenarios correspond to different trajectories, spanning from simple to complex. In scenario 01, the vehicles exclusively engage in straight-line motion. Scenario 02 involves the vehicles proceeding in a straight path followed by a turn. In Scenario 03, the vehicles initiate their journey from modeling area, traverse a circular path, and subsequently return to the vicinity of the starting point. Scenario 04 represents a more extensive trajectory, with the vehicle departing from the modeling area and covering a distance approximately twice that of Scenario 03. It is not illustrated in Fig.5 as it essentially entails the repetition of Scenario 03 twice, serving as a test of the system's long-distance performance.

There are some previously proposed cross-modal registration algorithms such as CMRNet[6] and I2D-Loc[9], but CMRNet is camera intrinsic dependent and cannot be applied to multi-camera systems because different cameras' intrinsics are different. Thus, to compare the system's performance,

TABLE II
CROSS-MODAL REGISTRATION

Seq.	I2D-Loc[9]			Ours		
	Rot.[°]	Trans.[m]	Fail[%]	Rot.[°]	Trans.[m]	Fail[%]
Vehicle01	1.753	0.031	2.88	**1.581**	**0.028**	**1.39**
Vehicle02	2.281	0.040	8.70	**2.087**	**0.036**	**3.47**



(a) vehicle01 (b) vehicle02

Fig. 7. Cross-modal registration results for two vehicles. The images show the aligned depth and RGB image after registration, with the colored points on the images representing depth points.

we adapt the SOTA cross-modal algorithm I2D-Loc[9] for infrastructure-based localization scenarios. Specifically, we have added a modeling module, a pose transformation module, and a simple camera switching module based on the intersection area between the vehicle and the camera's fov.

Table I summarizes the localization accuracy using different methods. As expected, an increase in scenario complexity may result in a slight decrease in localization accuracy. However, our system relies on single-frame matching, without accumulative errors, so the decrease in localization accuracy does not increase with the distance traveled. Compared to the I2D-Loc method, our approach achieved higher accuracy in both simple and complex scenarios, with a more significant improvement in accuracy observed in complex scenarios. The results also indicate that $Vehicle02$ has a larger rotation error. This could be due to the larger size of $Vehicle02$, which makes it more challenging to estimate the vehicle's pose when the camera captures only a portion of the vehicle.

C. Quantitative Experiments of Cross-Modal Registration

We also compared the accuracy of I2D-Loc[9] and our algorithm in cross-modal registration within an indoor parking lot environment. To ensure the diversity of experimental data, we chose scenario 03 (containing vehicle 01-03, vehicle 02-03) as the test data, and extracted images from them and disrupted the order. For $vehicle01$, a total of 1,941 pairs of point cloud models and images to be aligned are included, and for $vehicle02$, a total of 1,643 experimental data to be aligned. We assume the translation errors larger than 0.2 meters or rotation errors larger than 5 degrees as a failure.

The cross-modal registration results for two vehicles are shown in Fig.7, the results show that the network can effectively estimate correspondences, and the depth and RGB image can be aligned well. Table II shows the quantitative results. It can be observed that our registration method achieves better results compared to I2D-Loc in both translation and rotation errors. Moreover, our method achieves a lower failure rate and better robustness. The inference

TABLE III
TIME CONSUMING

	I2D-Loc[9]	ours
Inference time* [ms]	73	**42(∼23.8Hz)**

∗The image size is 640×360

TABLE IV
ABLATION STUDY

Method	Vehicle01-03		Vheicle02-03	
	Rot.[°]	Trans.[m]	Rot.[°]	Trans.[m]
Reg	0.975	0.054	fail	fail
Reg+Tracking	0.889	0.037	3.400	0.108
Reg+Cam	**0.729**	**0.025**	3.098	0.086
Reg+Cam+Tracking	0.797	0.030	**2.492**	**0.085**

efficiency results shown in Table III indicate that our method has a lower inference time than I2D-Loc and can meet the system's real-time requirements (15Hz).

D. Ablation Study

We conducted ablation experiments in sequences $vehicle01$-03 and $vehicle02$-03 to examine the impact of each module on the system's performance. Both sequences are long enough to allow a good evaluation of the system's performance for object localization. The quantitative results of the experiment are shown in Table IV. The vehicle modeling module is essential, so it is not explicitly mentioned in the method's name. Reg represents the cross-modal registration module. Cam represents the camera switching module, and if there is no Cam in name, it means using a simple camera switching module based on the intersection area.

The localization for $Vehicle01$ has already achieved a high level of accuracy, so adding other modules can only reach a similar level of accuracy. Although the inclusion of the tracking module may slightly reduce localization accuracy, it can achieve smoother vehicle motion. For $Vehicle02$, the localization accuracy is relatively lower, and each module demonstrates its importance, ultimately resulting in higher accuracy.

V. CONCLUSIONS

This paper proposes an infrastructure-based vehicle localization system using cross-modal registration, which utilizes LiDAR for vehicle modeling and locates vehicles by registering the vehicle model with the camera images. This system allows high accurate localization of vehicles without any modifications. We verified in the experiment section that the localization accuracy of the proposed system is able to serve Autonomous Valet Parking, and we believe that its localization accuracy is also sufficient for most other robotic applications, such as Automated Guided Vehicles. At the same time, our previous work [5] achieves a localization accuracy of 5 cm using solid-state LiDAR, and this proposed system achieves comparable accuracy at a much lower cost, with a wide range of prospects for dissemination.

REFERENCES

[1] D. Becker, J. Einsiedler, B. Schäufele, A. Binder, and I. Radusch, "Identification of vehicle tracks and association to wireless endpoints by multiple sensor modalities," in *International Conference on Indoor Positioning and Indoor Navigation*, 2013, pp. 1–10.

[2] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. Spie, 1992, pp. 586–606.

[3] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *ArXiv*, vol. abs/2004.10934, 2020.

[4] G. Bradski and J. Davis, "Motion segmentation and pose recognition with motion history gradients," in *Proceedings Fifth IEEE Workshop on Applications of Computer Vision*, 2000, pp. 238–244.

[5] B. Cao, Y. He, H. Zhuang, and M. Yang, "Infrastructure-based vehicle localization system for indoor parking lots using rgb-d cameras," *Journal of Shanghai Jiaotong University (Science)*, vol. 28, no. 1, pp. 61–69, 2023.

[6] D. Cattaneo, M. Vaghi, A. L. Ballardini, S. Fontana, D. G. Sorrenti, and W. Burgard, "Cmrnet: Camera to lidar-map registration," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 1283–1289.

[7] D. Cattaneo, D. G. Sorrenti, and A. Valada, "Cmrnet++: Map and camera agnostic monocular visual localization in lidar maps," *arXiv e-prints*, pp. arXiv–2004, 2020.

[8] B. Chen, A. Parra, J. Cao, N. Li, and T.-J. Chin, "End-to-end learnable geometric vision by backpropagating pnp optimization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8100–8109.

[9] K. Chen, H. Yu, W. Yang, L. Yu, S. Scherer, and G.-S. Xia, "I2d-loc: Camera localization via image to lidar depth flow," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 194, pp. 209–221, 2022.

[10] J. Einsiedler, D. Becker, and I. Radusch, "External visual positioning system for enclosed carparks," in *2014 11th Workshop on Positioning, Navigation and Communication (WPNC)*, 2014, pp. 1–6.

[11] J. Einsiedler, O. Sawade, B. Schäufele, M. Witzke, and I. Radusch, "Indoor micro navigation utilizing local infrastructure-based positioning," in *2012 IEEE Intelligent Vehicles Symposium*, 2012, pp. 993–998.

[12] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[13] A. Ibisch, S. Houben, M. Michael, R. Kesten, and F. Schuller, "Arbitrary object localization and tracking via multiple-camera surveillance system embedded in a parking garage," in *Video Surveillance and Transportation Imaging Applications 2015*, vol. 9407, 2015, p. 94070G.

[14] A. Ibisch, S. Houben, M. Schlipsing, R. Kesten, P. Reimche, F. Schuller, and H. Altinger, "Towards highly automated driving in a parking garage: General object localization and tracking using an environment-embedded camera system," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, 2014, pp. 426–431.

[15] A. Ibisch, S. Stümper, H. Altinger, M. Neuhausen, M. Tschentscher, M. Schlipsing, J. Salinen, and A. Knoll, "Towards autonomous driving in a parking garage: Vehicle localization and tracking using environment-embedded lidar sensors," in *2013 IEEE Intelligent Vehicles Symposium (IV)*, 2013, pp. 829–834.

[16] T. Krajník, M. Nitsche, J. Faigl, T. Duckett, M. Mejail, and L. Přeučil, "External localization system for mobile robotics," in *2013 16th International Conference on Advanced Robotics (ICAR)*, 2013, pp. 1–6.

[17] J. Ku, A. Harakeh, and S. L. Waslander, "In defense of classical image processing: Fast depth completion on the cpu," in *2018 15th Conference on Computer and Robot Vision (CRV)*. IEEE, 2018, pp. 16–22.

[18] J. Li and G. Hee Lee, "Deepi2p: Image-to-point cloud registration via deep classification," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 15 955–15 964.

[19] D. Mellinger, N. Michael, and V. Kumar, "Trajectory generation and control for precise aggressive maneuvers with quadrotors," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 664–674, 2012.

[20] T. Partanen, P. Müller, J. Collin, and J. Björklund, "Implementation and accuracy evaluation of fixed camera-based object positioning system employing cnn-detector," in *2021 9th European Workshop on Visual Information Processing (EUVIP)*, 2021, pp. 1–6.

[21] R. Pintus, E. Gobbetti, and M. Agus, "Real-time rendering of massive unstructured raw point clouds using screen-space operators," in *Proceedings of the 12th International conference on Virtual Reality, Archaeology and Cultural Heritage*, 2011, pp. 105–112.

[22] C. Rother, V. Kolmogorov, and A. Blake, "" grabcut" interactive foreground extraction using iterated graph cuts," *ACM transactions on graphics (TOG)*, vol. 23, no. 3, pp. 309–314, 2004.

[23] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, and D. Tao, "Gmflow: Learning optical flow via global matching," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 8121–8130.

[24] C. Zhang, H. Zhao, C. Wang, X. Tang, and M. Yang, "Cross-modal monocular localization in prior lidar maps utilizing semantic consistency," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 4004–4010.