Adaptive Cost Volume Fusion Network for Multi-Modal Depth Estimation in Changing Environments

Jinsun Park, Yongseop Jeong, Kyungdon Joo, Donghyeon Cho, and In So Kweon

***Abstract*—In this letter, we propose an adaptive cost volume fusion algorithm for multi-modal depth estimation in changing environments. Our method takes measurements from multi-modal sensors to exploit their complementary characteristics and generates depth cues from each modality in the form of adaptive cost volumes using deep neural networks. The proposed adaptive cost volume considers sensor configurations and computational costs to resolve an imbalanced and redundant depth bases problem of conventional cost volumes. We further extend its role to a generalized depth representation and propose a geometry-aware cost fusion algorithm. Our unified and geometrically consistent depth representation leads to an accurate and efficient multi-modal sensor fusion, which is crucial for robustness to changing environments. To validate the proposed framework, we introduce a new multi-modal depth in changing environments (MMDCE) dataset. The dataset was collected by our own vehicular system with RGB, NIR, and LiDAR sensors in changing environments. Experimental results demonstrate that our method is robust, accurate, and reliable in changing environments. Our codes and dataset are available at our project page.[1]**

***Index Terms*—AI-Based methods, data sets for robotic vision, deep learning for visual perception.**

I. Introduction

RECENT advances in computer vision and deep learning research have led to various real-world applications, including autonomous driving and unmanned robots for disasters. In these systems, an accurate depth perception capability is one of the most important factors to ensure safety and reliability. Depth information can be obtained using various sensors such as RGB cameras [1] and LiDARs [2]. However, the depth information from individual sensors is usually incomplete and noisy. Consequently, recent studies [3]–[6] have utilized multi-modal sensors and fuse depth cues to estimate accurate depth information. These methods tend to use existing familiar representations such as cost volumes [1], [6], [7], 2D depth maps [2]–[4], and 3D point clouds [8].

However, several issues pose as challenges to the multi-modal depth estimation in a practical setting. Firstly, there is no universal depth representation that can express the depth cues of various sensors in a unified manner. For example, conventional cost volumes are constructed with stereo images and are not directly applicable to represent depth cues from point clouds. Secondly, the existing algorithms prioritize specific configurations, such as a stereo RGB [1], [7] and RGB-LiDAR [9], [10] systems. Consequently, their frameworks are not directly scalable to different configurations. Moreover, the geometric configurations between sensors are not thoroughly considered during sensor fusion [9]. Lastly, real-world applications often operate under different weather conditions, times, and locations (*i.e.*, *changing environments*). However, public depth estimation datasets [5], [11]–[13] do not provide large-scale data captured in changing environments.

To overcome these problems, we propose an adaptive cost volume and extend its role to a generalized and unified depth representation for various sensors. It resolves redundant [14] and imbalanced depth bases problems and makes it possible to represent depth cues from passive and active depth sensors in a unified manner. Moreover, an efficient geometry-aware multi-modal cost volume fusion algorithm is proposed. Our method handles an arbitrary number of sensors for accurate

Manuscript received September 9, 2021; accepted January 25, 2022. Date of publication February 14, 2022; date of current version March 10, 2022. This letter was recommended for publication by Associate Editor X. Huang and Editor C. Cadena Lerma upon evaluation of the reviewers' comments. This work was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant NRF-2021R1I1A1A01060267, in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea Government (MSIT) under Grant 2020-0-01450, Artificial Intelligence Convergence Research Center, Pusan National University, and in part by the Ministry of Science and ICT and NIPA through HPC Support Project. The work of Kyungdon Joo was supported in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea Government (MSIT) under Grant 2020-0-01336 through Artificial Intelligence Graduate School Program (UNIST) and in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) under Grant NRF-2021R1C1C1005723. The work of Donghyeon Cho was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea Government (MSIT) under Grant 2021R1A4A1032580. *(Jinsun Park and Yongseop Jeong contributed equally to this work.) (Corresponding author: In So Kweon.)*

Jinsun Park is with the School of Computer Science and Engineering, Pusan National University (PNU), Busan 46241, Republic of Korea (e-mail: jspark@pusan.ac.kr).

Yongseop Jeong is with the Robotics Program, KAIST, Daejeon 34141, Republic of Korea (e-mail: yongseop@kaist.ac.kr).

Kyungdon Joo is with the Artificial Intelligence Graduate School and the Department of Computer Science and Engineering, UNIST, Ulsan 44919, Republic of Korea (e-mail: kyungdon@unist.ac.kr).

Donghyeon Cho is with the Department of Electronics Engineering, Chungnam National University (CNU), Daejeon 34134, Republic of Korea (e-mail: cdh12242@gmail.com).

In So Kweon is with the School of Electrical Engineering, KAIST, Daejeon 34141, Republic of Korea (e-mail: iskweon@kaist.ac.kr).

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2022.3150868, provided by the authors.

Digital Object Identifier 10.1109/LRA.2022.3150868

[1]https://github.com/zzangjinsun/MMDCE_RAL22

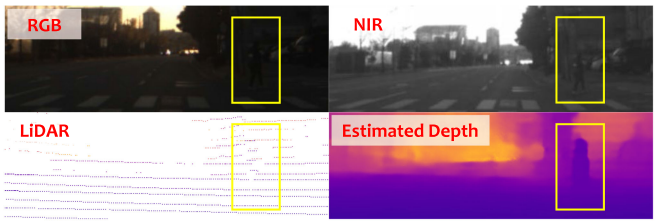

Fig. 1. **Example result of the proposed depth estimation method in a challenging environment.** Note that the depth of a person is correctly estimated owing to sparse LiDAR points although the person is hardly visible in RGB and NIR images (yellow boxes).

sensor fusion. We also introduce a new *multi-modal depth in changing environments* (*MMDCE*) dataset that includes various environmental changes. Our dataset was captured using stereo RGB, stereo NIR, and LiDAR sensors installed in a vehicular system to obtain complementary sensor data simultaneously. Our algorithm was evaluated on public outdoor [12] and our datasets to demonstrate its robustness, accuracy, and reliability in changing environments. A real-world depth estimation example in extreme light conditions is shown in Fig. 1.

II. Related Work

In this section, we introduce recent studies on deep learning-based depth estimation methods and multi-modal sensor systems and datasets.

Depth Estimation: Conventional depth estimation methods have utilized diverse input modalities for dense depth estimation. First of all, depth estimation can use either a single image [11] or stereo images [1], [7]. Chang and Chen [1] have constructed a cost volume that is iteratively refined by stacked hourglass networks using stereo images. MVSNet [15] and R-MVSNet [16] estimated a depth map from multi-view images by warping features from multi-view images to the reference frame. Furthermore, multi-modal information is proven to be effective for depth estimation. Liang *et al.* [17] estimated the depth from RGB and NIR images via spectral translation. Pseudo-LiDAR [18], [19] and PLUMENet [20] utilized cost volumes for the depth estimation and generated pseudo-LiDAR features for 3D object detection. Cheng *et al.* [4] proposed a noise-aware LiDAR and stereo RGB fusion network. However, these studies constructed cost volumes using a fixed disparity (or depth) range. Therefore, matching cost values should be calculated for each disparity basis regardless of the actual depth of a pixel. In other words, the *redundant depth bases problem* exists. DeepPruner [14] proposed a confidence range prediction to estimate the lower and upper bounds of the disparity range. However, this method still utilizes a fixed disparity interval between adjacent cost slices and does not consider sensor configurations (*e.g.*, intrinsic and extrinsic parameters).

Unlike previous methods, we utilize an adaptive cost volume as a generalized and unified depth representation for various types of sensors. The algorithm considers the overall geometric configuration of the system (*i.e.*, intrinsic and extrinsic parameters) together with the trade-off between accuracy and computational efficiency. To be specific, it adopts adaptive disparity and depth intervals between adjacent cost slices. This strategy increases the efficiency and maintains accuracy simultaneously by removing tiny disparity and depth spacing between adjacent cost slices while preserving distinctive depth bases. Furthermore, the cost volumes from multiple sensors are constructed to be geometrically consistent, ensuring the accuracy and efficiency in our sensor fusion process.

Multi-Sensor System and Dataset: Varieties of multi-sensor systems have recently been developed and used not only for academic purposes but also practical applications [21]–[24]. Geiger *et al.* [12] developed a vehicular system equipped with multiple cameras, a GNSS sensor, and a high-definition LiDAR for autonomous driving. Choi *et al.* [13] introduced a multi-spectral dataset containing RGB and thermal images. This dataset was captured at various times throughout the 24-hour cycle to ensure robust perception algorithm development. Bijelic *et al.* [25] proposed a multi-sensor system comprising a gated camera, RGB cameras, a LiDAR, and an FIR camera. The authors provide a large-scale dataset, called *DENSE*, captured in the real-world driving scenes over a wide range of areas for robust object detection in adverse weather conditions.

Unlike existing datasets, our multi-modal depth dataset provides depth information under various environmental changes with the help of our sensor system, which can operate under changing environments reliably.

III. Multi-Modal Depth Estimation

In this section, we first formally define the multi-modal depth estimation problem, and then we describe the manner in which adaptive depth bases are determined and adaptive cost volumes unify depth representations of passive (*e.g.*, stereo system) and active (*e.g.*, LiDAR) sensors. We then describe our network architecture and loss functions for training.

A. Problem Definition

We define the multi-modal depth estimation problem as a task to estimate dense depth information of a scene given data from diverse sensors as inputs. Here, the input data may include images (*e.g.*, RGB, grayscale, and NIR), sparse depth measurements (*e.g.*, LiDAR and Radar), and other forms of data containing depth cues (*e.g.*, defocus blur [26]). Let $\mathcal{D} = \{I_i\}_{i=1}^{|\mathcal{D}|}$ be a set of input domains where I_i is the i-th domain input. Then, the multi-modal depth estimation problem can be defined as follows:

$$\mathbf{D} = Z\left(F\left(\{G_i(I_i|\,\theta_{G_i})\}_{i=1}^{|\mathcal{D}|}\,\middle|\,\theta_F\right)\middle|\,\theta_Z\right), \tag{1}$$

where $\mathbf{D}$ denotes a dense depth; $Z(\cdot|\theta_Z)$ is a depth regression function; $F(\cdot|\theta_F)$ is a depth cue fusion function; $G_i(\cdot|\theta_{G_i})$ is the depth cue generation function of the i-th domain (*e.g.*, a stereo matcher or an optical flow estimator); and θ_Z, θ_F, and θ_{G_i} denote parameters of the corresponding functions (network weights and camera parameters). This definition implies that intermediate depth cues (adaptive cost volumes) are generated from domain-specific functions to benefit maximally from the multi-modal data. The overall pipeline of the proposed algorithm is shown in Fig. 2.

B. Adaptive Cost Volume Generation

The cost volume that we selected as our unified depth representation differs from the conventional cost volume because each depth basis is determined by considering the spacing between adjacent depth bases. In a stereo RGB setup, for example, suppose that the features from left and right images and a

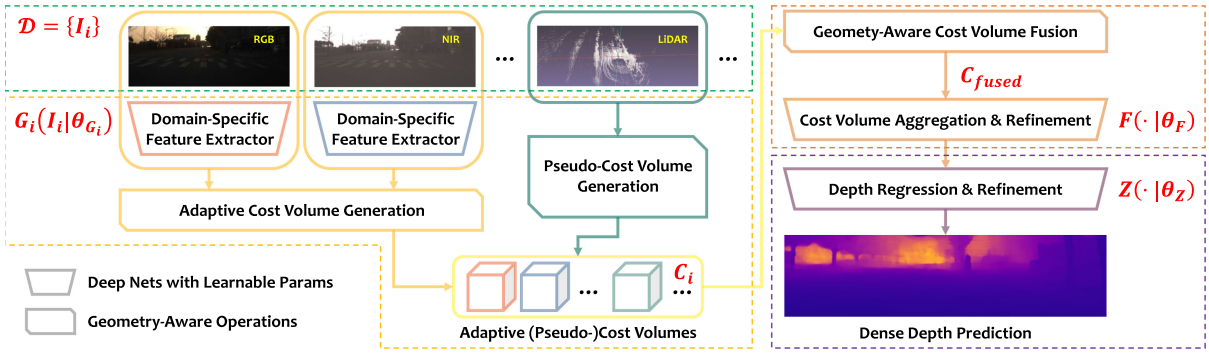

Fig. 2. **Overall pipeline of the proposed algorithm.** The proposed framework is an end-to-end trainable deep neural network. Depth cues from multiple sensors are represented by adaptive and pseudo-cost volumes and are geometrically fused for depth estimation. Note that we adopt individual feature extractors for passive sensors to effectively utilize domain-specific features.

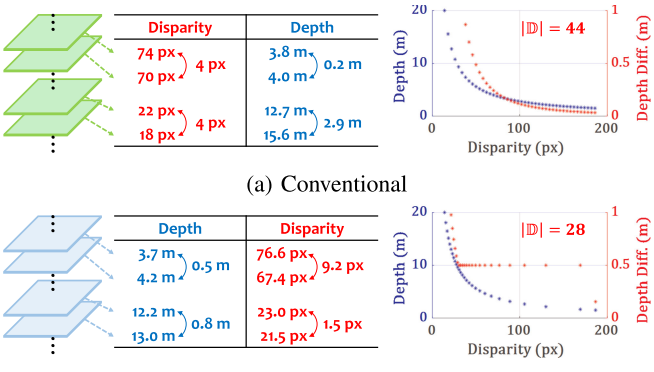

Fig. 3. **Comparison of conventional and proposed adaptive cost volumes and depth bases analyses.** Cost volumes are calculated with an example stereo system with focal length 700 px, baseline 0.4 m, and disparity interval 4 px. (a) Conventional disparity-based cost volume is purely based on a fixed disparity interval. (b) Proposed adaptive cost volume minimizes the redundancy and imbalance of depth and disparity bases. In depth bases plots on the right, the blue and red points denote depth values of each cost slice and depth differences between adjacent cost slices, respectively.

Algorithm 1: Adaptive Depth Bases Generation.

Input : Target depth range $[d_{min}, d_{max}]$, unit depth τ, unit disparity $\bar{\tau}$, and depth-disparity conversion function $z(\cdot)$
Output : Adaptive depth bases $\mathbb{D}$ for cost volume generation

$\mathbb{D} = \{d_{min}\}$, $j = 1$;
while $d_j < d_{max}$ **do**
  $\bar{d}_j = z(d_j)$;
  $d_\alpha = d_j + \tau$; // Ensure unit depth τ
  $\bar{d}_\alpha = z(d_\alpha)$;
  if $|\bar{d}_j - \bar{d}_\alpha| < \bar{\tau}$ **then**
    $\bar{d}_\alpha = \bar{d}_j - \bar{\tau}$; // Ensure unit disparity $\bar{\tau}$
    $d_\alpha = z(\bar{d}_\alpha)$;
  $\mathbb{D} = \mathbb{D} \cup \{d_\alpha\}$; // Add a new depth basis
  $j = j + 1$;
end

target depth range (*i.e.*, depth range of interest) are given. In conventional methods [1], [4], [7], the target depth range is further discretized to a set of depth bases $\mathbb{D} = \{d_j\}_{j=1}^{|\mathbb{D}|}$ where j is the depth basis index. Then, a cost volume C is constructed by calculating matching costs (*e.g.*, ℓ_1 or ℓ_2 feature distance) for each depth basis, d_j, as follows:

$$C(x, y, j) = \phi\left(f_l(x, y), f_r(x - \bar{d}_j, y)\right), \tag{2}$$

where x and y are the pixel coordinates; $\bar{d}_j$ is the disparity value that corresponds to depth d_j; f_l and f_r are features from left and right images, respectively; and $\phi(\cdot, \cdot)$ is a matching cost function. After several optional cost aggregation and refinement processes [7], [27] on C, the disparity value with the minimum matching cost is assigned to the pixel and converted to the corresponding depth value. An example cost volume for the depth range [1.5 m, 20 m] is shown in Fig. 3(a). Because the regularly discretized disparity bases $\bar{\mathbb{D}} = \{\cdots, 18, 22, \cdots\}$ are ignorant of the system configuration, there exist imbalances and tiny depth spacing between adjacent cost slices (*i.e.*, *imbalanced and redundant depth bases problem*). These subtle depth differences can be often ignored to minimize redundancy in $\mathbb{D}$ and reduce computational costs in real-world applications [14], [28].

To resolve the problem, we propose to construct a cost volume based on a set of adaptive depth bases $\mathbb{D}$. We determine $\mathbb{D}$ to minimize redundancy and imbalance by considering the system configuration (*e.g.*, focal length and baseline for stereo) and depth and disparity intervals between adjacent cost slices. Algorithm 1 describes the proposed adaptive depth bases generation algorithm. In Algorithm 1, $\mathbb{D}$ is determined by two rules: two adjacent cost slices should either have i) greater depth difference than the unit depth τ, or ii) greater disparity difference than the unit disparity $\bar{\tau}$. Our algorithm ensures that the generated $\mathbb{D}$ is with less redundancy by only preserving the meaningful depth and disparity differences between adjacent cost slices.

Fig. 3 provides analyses of the conventional and proposed adaptive depth bases of cost volumes. The conventional method results in many tiny depth and disparity gaps between adjacent cost slices. In contrast, our method has eliminated redundant depth bases effectively, yielding a drastically reduced computational cost ($28/44 = 63.6\%$). These properties are particularly important because a low computational cost is essential for real-world applications.

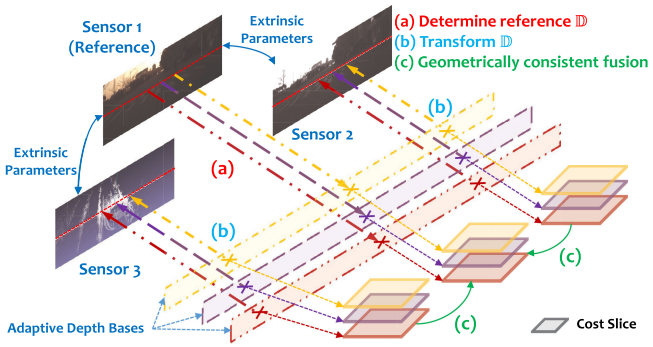

Fig. 4. **Geometry-aware cost volume fusion algorithm.** We first (a) determine the set of adaptive depth bases $\mathbb{D}$ of the reference sensor, and then (b) $\mathbb{D}$ is transformed to the coordinates of other sensors. This leads to (c) geometrically consistent cost volumes across sensors because the cost slices of each sensor at the same index represent the same 3D plane in the world.

C. Pseudo-Cost Volume Generation

Although we have adopted the adaptive cost volume as a unified depth representation for multi-modal fusion, the construction of cost volumes from active sensors that directly provide depth measurements (*e.g.*, LiDARs) is not straightforward. Unlike existing algorithms that directly infer dense depth from a sparse depth [2] or assign fixed coefficients [29], we propose to construct a pseudo-cost volume from depth measurements.

Specifically, once $\mathbb{D}$ and $Z(\cdot)$ are given (*cf.*, (1)), a depth value can be directly converted to a cost-like representation. In this work, the softmax-based depth regression function [1], [7] is adopted as $Z(\cdot)$, defined as follows:

$$\mathbf{D} = Z(C \,|\, \theta_Z) = \sum_{j=1}^{|\mathbb{D}|} d_j \cdot \sigma\left(C(j)\right), \tag{3}$$

where $\sigma(\cdot)$ denotes the softmax function. Note that θ_Z includes the set of adaptive depth bases $\mathbb{D}$.

With this setup, a depth value d can be represented as a linear combination of the elements in $\mathbb{D}$ as follows:

$$d = \alpha d_l + (1-\alpha) d_u, \quad d \in [d_l, d_u), \tag{4}$$

where d_l and d_u denote the lower and upper depth boundaries in $\mathbb{D}$. With a slight abuse of notation, the proposed pseudo-cost representation is defined as follows:

$$C_{pseudo}(j) = \begin{cases} \alpha & \text{if } d_j = d_l \\ 1-\alpha & \text{if } d_j = d_u, \\ 0 & \text{otherwise} \end{cases} \quad j \in [1, |\mathbb{D}|], \tag{5}$$

where $|\cdot|$ denotes the cardinality. Note that this representation can be understood as a special case of Eq. (3), where all the coefficients, except for α and $1-\alpha$, are zeros. With the proposed pseudo-cost volume representation, the depth information from both passive (*e.g.*, stereo cameras) and active (*e.g.*, LiDARs) sensors can be manipulated seamlessly.

D. Geometry-Aware Cost Volume Fusion

In order to accurately fuse depth cues from multi-modal sensors, we must consider the geometric configuration between sensors, as shown in Fig. 4. Without loss of generality, we first set one sensor as a reference and assume that the coordinates of the other sensors are aligned with the reference sensor. This assumption is reasonable because all sensors are calibrated and sensor data can be rectified (*e.g.*, images) or reprojected (*e.g.*, LiDARs). Under this assumption, the set of adaptive depth bases $\mathbb{D}$ of the reference sensor is determined [see Fig. 4(a)] and transformed to the coordinates of other sensors with extrinsic parameters between sensors [see Fig. 4(b)]. As a result, the cost slices of each sensor at the same index represent the identical depth plane in the 3D space regardless of sensor types or locations. This ensures a geometric consistency in cost volumes across sensors.

Note that our geometrically consistent representation simplifies the reprojection in 3D space to the warping in 2D space [30] because the cost slices to be fused between different modalities are on the same plane, as shown in Fig. 4. Therefore, the proposed method further reduces computational costs because warping requires 2D interpolation, whereas reprojection requires 3D interpolation.

The proposed geometry-aware cost volume fusion is defined as follows:

$$C_{fused} = F\left(\{C_i\}_{i=1}^{|\mathcal{D}|} \,\middle|\, \theta_F\right) = \frac{\sum_{I_i \in \mathcal{D}} M_i \hat{C}_i}{\sum_{I_i \in \mathcal{D}} M_i}, \tag{6}$$

where C_{fused} is the fused cost volume, $\hat{C}_i$ is the warped cost volume from the i-th domain I_i, and M_i is its valid pixel mask. Note that θ_F includes intrinsic and extrinsic parameters of sensors. In (6), the valid pixel mask M_i plays an important role in fusing depth cues from only the valid pixels of each sensor. For example, the depth value of a pixel without LiDAR depth information should be determined by information from stereo images from RGB, grayscale, and NIR cameras. In the valid pixel mask M of LiDAR pseudo-cost volumes, the regions without LiDAR values are set to zero. Thus, only the depth cues from stereo images contribute to estimating the accurate depth values regardless of missing LiDAR values.

Note that although M can be replaced with confidence or reliability predictions, we presume that it would be more challenging to estimate them in changing environments. Instead, we utilize intra- and cross-scale cost aggregation processes [7] on C_{fused} to suppress any unreliable and inaccurate depth cues. In fact, the expected roles of the aggregation and confidence predictions are similar; therefore, we adopt the valid pixel mask-based fusion strategy. After fusion and cost aggregation, the final depth is estimated by (3) through the fused cost volume C_{fused}.

E. Depth Estimation Framework

An overview of our network is shown in Fig. 2. For image feature extractors, we adopted feature pyramid networks [31] to utilize multi-scale features. We also utilized image-based depth refinement [27] to further enhance the depth accuracy.

We trained our network with ℓ_1 or ℓ_2 loss as a reconstruction loss with the ground truth depth as follows:

$$L(\mathbf{D}^{gt}, \mathbf{D}^{pred}) = \sum_{x,y} \left|\mathbf{D}^{gt}(x,y) - \mathbf{D}^{pred}(x,y)\right|^{\rho}, \tag{7}$$

where $\mathbf{D}^{gt}$ and $\mathbf{D}^{pred}$ denote the ground truth and predicted depth values, respectively. Here, ρ is set to 1 for ℓ_1 loss and to 2 for ℓ_2 loss. Note that modules with trainable parameters, such as feature extractors, cost aggregation and refinement, and depth refinement modules, are trained with (7).

IV. Multi-Modal Depth Dataset

Over the last decade, various large-scale multi-sensor depth datasets [5], [12], [25], [32]–[35] have been made publicly available. Unfortunately, they do not consider dynamic environmental changes or do not provide the accurate (semi-)dense ground truth depth information.

Therefore, we introduce a new multi-modal depth dataset with changing environments. For this purpose, we propose a new multi-modal vehicular sensor system. In addition, we introduce a new KITTI multi-modal depth dataset by rearranging existing KITTI datasets [2], [12]. Due to space limitations, we will briefly describe our system and multi-modal datasets. Please refer to the supplementary material for further details.

A. Vehicular Multi-Modal Sensor System

To benefit from the complementary characteristics of multi-modal sensors and ensure robustness toward changing environments, we adopted stereo RGB, stereo NIR, and two LiDARs for our sensor system. In addition, we utilized a GNSS/IMU sensor to capture synchronized vehicle poses in GNSS coordinates. The intrinsic and extrinsic parameters of the sensors are pre-calibrated by setting the left RGB camera as a reference [36]–[38]. The stereo RGB provides 1224×360 images with 0.3 m baseline, the stereo NIR provides 1280×360 images with 0.05 m baseline after the rectification, and LiDARs provide roughly 8 K depth points that are visible in the reference frame. All sensors, except LiDARs that are waterproof, were installed inside the vehicle with a stable power supply to ensure reliable operations under extreme conditions including rain, snow, and fog.

B. Multi-Modal Depth in Changing Environments Dataset

For our multi-modal depth in changing environments (MMDCE) dataset generation, we collected synchronized sensor data in campus, residential, and downtown areas in various environmental conditions, including weather, time, and seasonal changes. We followed the semi-dense ground truth depth generation method of the KITTI Depth Completion (KITTI DC) dataset [2]. Specifically, 10–15 successive frames of point clouds were accumulated using vehicle poses from the GNSS/IMU sensor. Then their alignment was further refined [39] and filtered with stereo depth estimation results [7].

We collected 20 driving sequences in day and night with various environments and extracted more than 100 K synchronized frames, and semi-dense ground truth depth maps are generated. In total, 6,628 images were generated, including 5,876 daytime (Train: 4,344, Validation: 656, Test: 876) and 752 nighttime (Train: 601, Test: 151) images. Note that there is no overlap between train, validation, and test sets in time, weather, and locations. Fig. 5 shows example data captured under various environmental conditions.

C. KITTI Multi-Modal Depth Dataset

One commonly used depth dataset, the KITTI DC dataset [2], only provides a single RGB image and LiDAR measurements. Fortunately, the KITTI raw dataset [12] provides stereo RGB and grayscale images (1242×375 and 0.54 m baseline), and we can trace synchronized stereo grayscale and RGB images of the KITTI DC dataset. Therefore, we rearranged the KITTI DC dataset to generate the KITTI multi-modal depth (KITTI MMD) dataset in order to compare the proposed approach with other methods. In this way, we collected 32,917 train, 3,426 validation, and 1,000 test data. Each sample consists of stereo RGB, stereo grayscale, and LiDAR point clouds with roughly 20 K points, as well as a semi-dense ground truth depth image. Fig. 6(a) and (b) show example images of the KITTI MMD dataset.

Fig. 5. **Example images of the proposed MMDCE dataset.** Our dataset was captured with various environmental changes. Note that only left RGB images are shown.

V. Experimental Results

In this section, we describe the implementation details and evaluate the depth estimation performance of the proposed algorithm on the MMDCE and KITTI MMD datasets. Furthermore, the robustness and stability of our algorithm in dynamic environments were verified using the proposed MMDCE dataset. Additionally, ablation studies on input modalities, adaptive depth bases, and computational costs were conducted.

Our method was implemented using PyTorch [44] and trained using four NVIDIA TITAN RTX GPUs. For all experiments, we used an ADAM optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.99$, and an initial learning rate set to 0.001. Other training details will be individually described for each dataset. The evaluation metrics adopted for the quantitative evaluations are RMSE, MAE, iRMSE, and iMAE [2].

Note that adaptive cost volumes are constructed from stereo RGB, stereo grayscale (or NIR), and pseudo-cost volumes are constructed from LiDARs in our algorithm.

A. KITTI MMD Dataset

Our algorithm was trained for 30 epochs with the ℓ_1 loss and a batch size of 16. The learning rate decayed by 0.2 at 10, 20, and 25 epochs.

Table I shows the quantitative evaluation results on the KITTI MMD dataset. Note that the proposed, LS [4], and CCVN [6] methods were trained with approximately 30 K training data, whereas the other methods were trained with approximately 90 K training data. The depth estimation accuracy of the LiDAR only algorithm [2] is quite low because of the sparsity of the input information. The RGB and LiDAR-based depth completion algorithms [10], [40]–[43] handled this problem effectively by utilizing guidance from the additional RGB image to propagate input sparse depth values into neighboring pixels. Stereo RGB and LiDAR-based algorithms [4], [6] show comparable performance or even further improvement despite the relatively small amount of training data because the depth cues from stereo images are helpful for the regions without LiDAR measurements. Furthermore, the proposed algorithm using stereo RGB, stereo grayscale, and LiDAR shows the best performance in RMSE and iMAE, as well as comparable performance in MAE and iRMSE.

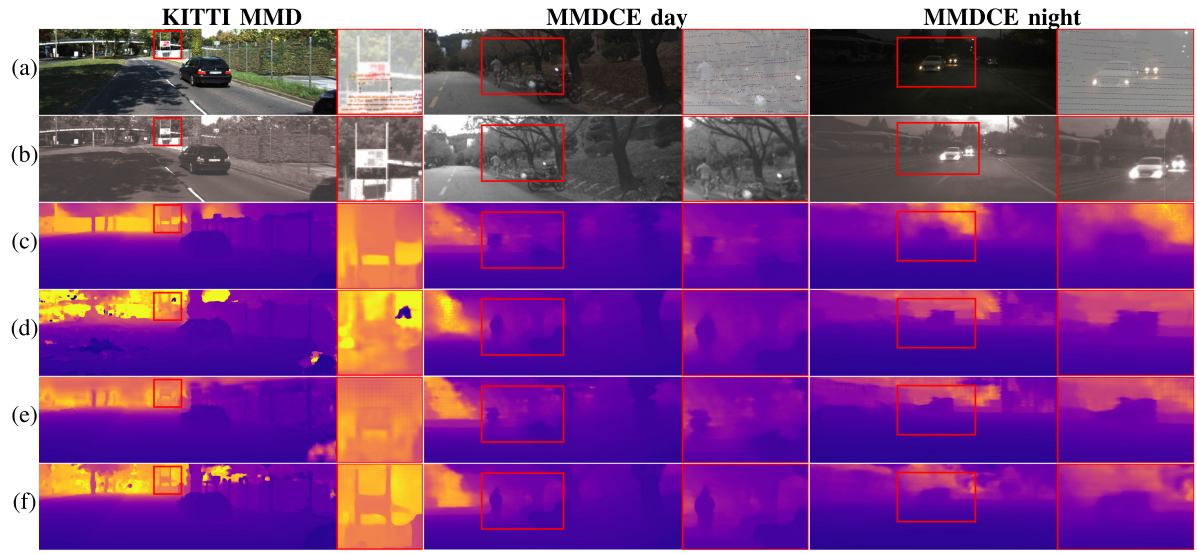

Fig. 6. **Depth estimation results on the KITTI MMD, MMDCE day, and MMDCE night datasets.** (a) RGB and LiDAR, (b) Grayscale or NIR, (c) NLSPN [10], (d) LS [4], (e) CCVN [6], and (f) Ours. Our method estimates accurate depth values especially on the regions without LiDAR measurements. Note that the amount of the input sparse depth of our MMDCE dataset is approximately 40% compared to that of the KITTI MMD dataset.

TABLE I
QUANTITATIVE EVALUATION ON THE KITTI MMD DATASET (R: RGB, L: LIDAR, G: GRAYSCALE, M: MONO, S: STEREO)

Method	Input			RMSE	MAE	iRMSE	iMAE
	R	L	G	(mm)	(mm)	(1/km)	(1/km)
SparseConv [2]	-	✓	-	2010.00	680.00	-	-
DDP [40]	M	✓	-	1310.03	347.17	-	-
NConv [41]				908.76	209.56	2.50	0.90
S2D [42]				878.56	260.90	3.25	1.34
DN [43]				811.07	236.67	2.45	1.11
NLSPN [10]				771.80	**197.30**	2.00	**0.80**
LS [4]	S	✓	-	832.16	283.91	2.19	1.10
CCVN [6]				749.30	252.50	**1.40**	0.81
Proposed			S	**673.34**	202.56	1.69	**0.80**

Bold entities are the best result for each metric.
Underlined entries are the second-best result for each metric.

TABLE II
QUANTITATIVE EVALUATION ON THE MMDCE DATASET (R: RGB, L: LIDAR, N: NIR, M: MONO, S: STEREO)

	Method	Input			RMSE	MAE	iRMSE	iMAE
		R	L	N	(mm)	(mm)	(1/km)	(1/km)
(a) Day	NLSPN [10]	M	✓	-	1750.6	709.7	9.3	4.6
		-		M	1791.4	831.8	13.5	7.3
	LS [4]	S	✓	-	1759.6	939.8	7.7	4.7
		-		S	13009.5	8353.5	63.9	55.3
	CCVN [6]	S	✓	-	2141.4	1046.2	10.6	5.7
		-		S	5884.9	2379.0	17.8	8.9
	Proposed	S	✓	S	**1226.2**	**610.4**	**6.9**	**3.8**
(b) Night	NLSPN [10]	M	✓	-	1755.2	716.8	9.7	5.3
		-		M	2126.5	1031.8	15.6	9.4
	LS [4]	S	✓	-	3589.8	1431.8	16.3	9.0
		-		S	9289.3	6162.2	67.7	56.8
	CCVN [6]	S	✓	-	1722.4	727.0	10.3	5.5
		-		S	3884.4	1569.2	17.6	9.4
	Proposed	S	✓	S	**1371.3**	**663.6**	**8.2**	**4.8**

Bold entities are the best result for each metric.

Owing to the additional depth cues from grayscale stereo images, the ambiguities arising from stereo RGB or LiDAR sensors are effectively resolved.

Fig. 6 shows depth estimation results in challenging areas. NLSPN [10] shows reliable performance on the regions with LiDAR measurements; however, its depth prediction accuracy degrades on the regions without any depth values, as shown in Fig. 6(c). This problem can be easily handled by stereo-based algorithms, as the depth cues for those regions are still available from stereo images. For example, LS [4] and CCVN [6] utilize stereo RGB and LiDAR; however, they simply concatenate a sparse depth image to the corresponding RGB image. Because they do not consider geometric information, the depth prediction accuracy is very low [see Fig. 6(d) and (e)]. This problem is effectively handled by our algorithm because depth cues from multiple domains are carefully fused in a geometry-aware manner [see Fig. 6(f)]. Furthermore, possible noisy depth cues from one domain can be implicitly handled during multi-modal cost volume fusion owing to the accurate and complementary depth cues from the other domains.

B. *Proposed MMDCE Dataset*

Our network was trained for 30 epochs with the $\ell_1 + \ell_2$ loss and a batch size of 8. For the nighttime split, we fine-tuned the network trained on the daytime split because the number of nighttime images was less. The other training settings were the same as those in Sec. V-A.

Table II(a) shows the quantitative evaluation results on the daytime split of our MMD dataset. NLSPN [10], LS [4], and CCVN [6] were trained with the RGB-LiDAR and NIR-LiDAR configurations for detailed analyses. The proposed algorithm yields the best performance in all metrics. NLSPN shows similar performance with RGB-LiDAR and NIR-LiDAR configurations because it only utilizes a single image from the RGB or NIR domains. However, LS and CCVN do not show reliable performance with the NIR-LiDAR setup. In the proposed system, the stereo NIR has a smaller baseline (0.05 m) compared to that of the stereo RGB (0.3 m) (*cf.*, Sec. IV-A). In this case, the viable

TABLE III
PERFORMANCE EVALUATION WITH VARIOUS INPUT COMBINATIONS
(R: RGB, L: LIDAR, G: GRAYSCALE, N: NIR, S: STEREO)

Dataset	Input			RMSE	MAE	iRMSE	iMAE
	R	L	G/N	(mm)	(mm)	(1/km)	(1/km)
KITTI MMD	S	-	-	904.18	339.74	2.67	1.24
		✓		722.63	244.25	2.22	1.00
			S	**688.18**	**235.82**	**2.08**	**0.97**
MMDCE Day	S	-	-	2186.2	1245.5	15.2	8.5
	-		S	2516.7	1501.2	19.0	11.1
	S	✓	-	1679.6	865.0	10.7	5.5
	-		S	1929.8	1031.3	12.6	6.8
	S			**1540.8**	**778.4**	**9.0**	**4.9**

Bold entities are the best result for each metric.

disparity range of the stereo NIR is quite limited. Because LS and CCVN utilize the conventional cost volume construction method, they fail to estimate far-depth ranges. In contrast, the proposed adaptive cost volume construction method can handle various system configurations. Therefore, our algorithm yields reliable performance even with a small stereo baseline setup.

Similarly, our method shows reliable performance in nighttime scenes, as shown in Table II(b). Because nighttime scenes often suffer from low lighting conditions or blur, the performance is slightly lower compared to that of the daytime split, with the exception of CCVN. However, our algorithm successfully exploits various depth cues from multiple sensors and outperforms the other methods by a large margin.

Fig. 6 shows depth estimation results with challenging environments in daytime and nighttime. Because the amount of input sparse depth points is about 40% compared to that of the KITTI MMD, NLSPN [10] experiences performance degradation. Specifically, as there are fewer depth points on objects, boundaries are often not well preserved and unwanted artifacts frequently appear [see Fig. 6(c)]. Although LS [4] and CCVN [6] show better performance, they suffer from mixed depth values on under-exposed areas [see Fig. 6(d) and (e)]. In our algorithm, although RGB images suffer from under-exposure, NIR images exhibit slightly better exposure; therefore, as shown in Fig. 6(f), better depth cues can be extracted from them, compared to those obtained using RGB images.

In the nighttime, the advantages of the proposed method are easily observable, as shown in Fig. 6. Because of the low-light conditions, RGB images do not provide detailed information about a given scene. However, LiDARs and NIR images still provide sufficient information, although NIR images exhibit slight blurring. Therefore, the proposed algorithm shows better dense depth prediction results compared to those of the other algorithms, as shown in Fig. 6.

C. Ablation Studies

Input Modalities: To analyze the effect of input modalities, our network was trained for 10 epochs with various input combinations. Table III shows performance comparisons with various input modalities on the KITTI MMD and the proposed MMDCE daytime datasets. The stereo RGB alone shows moderate depth estimation performance. While LiDARs provide accurate depth values from close to far distances, the stereo images cannot provide accurate depth estimation at far distances due to the limited disparity resolution. Therefore, adding LiDAR information effectively guides depth estimation in distant regions. Moreover,

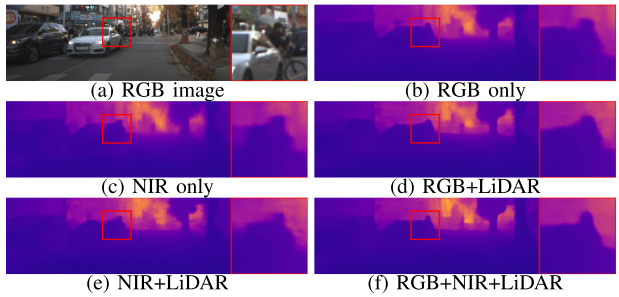

Fig. 7. **Depth estimation results with various input combinations.** Depth accuracy is getting improved with more input modalities (*cf.*, Tab. III). (a) RGB image. (b) RGB only. (c) NIR only. (d) RGB+LiDAR. (e) NIR+LiDAR. (f) RGB+NIR+LiDAR.

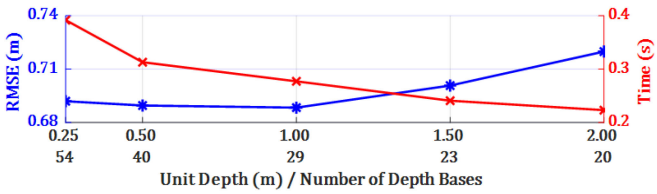

Fig. 8. **Depth estimation performance and processing time analysis with various depth bases on the KITTI MMD dataset.** Processing time decreases while RMSE increases as adaptive depth bases get coarser (*i.e.*, as unit depth gets larger) as expected.

in the KITTI MMD dataset, the additional fusion of depth cues from grayscale images further boosts the performance significantly. Because grayscale images provide depth cues from a different perspective, some occluded regions in RGB images may be observed. Furthermore, in the proposed MMDCE dataset, NIR images often provide higher image quality in extreme light conditions than that of RGB images (*cf.*, Fig. 6). Therefore, the fusion of depth cues from grayscale or NIR images leads to better prediction results. Fig. 7 shows the prediction results on our MMDCE dataset with various input modalities. As more input modalities are utilized, the depth estimation accuracy improves.

Number of Adaptive Depth Bases: We further analyzed the proposed algorithm by comparing the depth estimation performance and processing time with various unit depth values on the KITTI MMD dataset, as shown in Fig. 8. Note that a small unit depth τ leads to a finer $\mathbb{D}$ with increasing number of elements. As expected, a coarser $\mathbb{D}$ leads to a lower processing time and larger RMSE. Therefore, $\mathbb{D}$ can be determined from the available computational resources of the given environment.

VI. CONCLUSION

In this letter, we proposed a multi-modal sensor fusion algorithm for depth estimation in changing environments. The role of the cost volume is extended to a generalized depth representation, and the adaptive cost volume is proposed to minimize redundancy and imbalance of depth bases in conventional cost volumes. In addition, our geometry-aware cost volume fusion algorithm accurately fuses geometrically consistent cost volume across multiple sensors. As a result, complementary characteristics of multi-modal sensors are effectively merged, and the robustness to changing environments is increased considerably. We also proposed a vehicular multi-modal sensor system that comprises stereo RGB, stereo NIR, a GNSS/IMU,

and two LiDARs. Using the system, we collected a multi-modal depth dataset captured through various environmental changes in weather, location, and time. The proposed algorithm was evaluated on the KITTI MMD and proposed MMDCE datasets both quantitatively and qualitatively and was shown to outperform state-of-the-art algorithms substantially. In future studies, we will investigate domain-specific confidence-based cost fusion algorithms.

REFERENCES

[1] J.-R. Chang and Y.-S. Chen, "Pyramid stereo matching network," in *Proc. Comput. Vis. Pattern Recognit.*, 2018, pp. 5410–5418.

[2] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger, "Sparsity invariant CNNs," in *Proc. Int. Conf. 3D Vis.*, 2017, pp. 11–20.

[3] F. Ma and S. Karaman, "Sparse-to-dense: Depth prediction from sparse depth samples and a single image," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 4796–4803.

[4] X. Cheng, Y. Zhong, Y. Dai, P. Ji, and H. Li, "Noise-aware unsupervised deep lidar-stereo fusion," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 6332–6341.

[5] T. Zhi, B. R. Pires, M. Hebert, and S. G. Narasimhan, "Deep material-aware cross-spectral stereo matching," in *Proc. Comput. Vis. Pattern Recognit.*, 2018, pp. 1916–1925.

[6] T.-H. Wang, H.-N. Hu, C. H. Lin, Y.-H. Tsai, W.-C. Chiu, and M. Sun, "3D LiDAR and stereo fusion using stereo matching network with conditional cost volume normalization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 5895–5902.

[7] H. Xu and J. Zhang, "AANet: Adaptive aggregation network for efficient stereo matching," in *Proc. Comput. Vis. Pattern Recognit.*, 2020, pp. 1959–1968.

[8] Y. Chen, B. Yang, M. Liang, and R. Urtasun, "Learning joint 2D-3D representations for depth completion," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 10022–10031.

[9] X. Cheng, P. Wang, and R. Yang, "Depth estimation via affinity learned with convolutional spatial propagation network," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 103–119.

[10] J. Park, K. Joo, Z. Hu, C.-K. Liu, and I. S. Kweon, "Non-local spatial propagation network for depth completion," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 913–917.

[11] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2366–2374.

[12] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Proc. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.

[13] Y. Choi *et al.*, "Kaist multi-spectral day/night data set for autonomous and assisted driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 934–948, Mar. 2018.

[14] S. Duggal, S. Wang, W.-C. Ma, R. Hu, and R. Urtasun, "DeepPruner: Learning efficient stereo matching via differentiable patchmatch," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 4383–4392.

[15] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan, "MVSNet: Depth inference for unstructured multi-view stereo," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 767–783.

[16] Y. Yao, Z. Luo, S. Li, T. Shen, T. Fang, and L. Quan, "Recurrent mvsnet for high-resolution multi-view stereo depth inference," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 5520–5529.

[17] M. Liang, X. Guo, H. Li, X. Wang, and Y. Song, "Unsupervised cross-spectral stereo matching by learning to synthesize," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 8706–8713.

[18] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Q. Weinberger, "Pseudo-lidar from visual depth estimation: Bridging the gap in 3D object detection for autonomous driving," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 8437–8445.

[19] Y. You *et al.*, "Pseudo-LiDAR: Accurate depth for 3D object detection in autonomous driving," in *Proc. Int. Conf. Learn. Representations*, 2019, pp. 8437–8445.

[20] Y. Wang, B. Yang, R. Hu, M. Liang, and R. Urtasun, "PLUMENet: Efficient 3D object detection from stereo images," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 3383–3390.

[21] M.-F. Chang *et al.*, "Argoverse: 3D tracking and forecasting with rich maps," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 8740–8749.

[22] X. Huang *et al.*, "The apolloscape dataset for autonomous driving," in *Proc. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 1067–10676.

[23] A. Ligocki, A. Jelinek, and L. Zalud, "Brno urban dataset-the new data for self-driving agents and mapping tasks," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 3284–3290.

[24] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The oxford robotcar dataset," *J. Robot. Res.*, vol. 36, no. 1, pp. 3–15, 2017.

[25] M. Bijelic *et al.*, "Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather," in *Proc. Comput. Vis. Pattern Recognit.*, 2020, pp. 11682–11692.

[26] S. Suwajanakorn, C. Hernandez, and S. M. Seitz, "Depth from focus with your mobile phone," in *Proc. Comput. Vis. Pattern Recognit.*, 2015, pp. 3497–3506.

[27] R. Chabra, J. Straub, C. Sweeney, R. Newcombe, and H. Fuchs, "StereoDRNet: Dilated residual stereonet," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 11778–11787.

[28] A. Badki, A. Troccoli, K. Kim, J. Kautz, P. Sen, and O. Gallo, "Bi3D: Stereo depth estimation via binary classifications," in *Proc. Comput. Vis. Pattern Recognit.*, 2020, pp. 1597–1605.

[29] S. Imran, Y. Long, X. Liu, and D. Morris, "Depth coefficients for depth completion," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 12438–12447.

[30] M. Jaderberg *et al.*, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 239–242.

[31] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, pp. 936–944.

[32] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 746–760.

[33] A. Chang *et al.*, "Matterport3D: Learning from RGB-D data in indoor environments," in *Proc. Int. Conf. 3D Vis.*, 2018, pp. 667–676.

[34] V. Guizilini, R. Ambrus, S. Pillai, A. Raventos, and A. Gaidon, "3D packing for self-supervised monocular depth estimation," in *Proc. Comput. Vis. Pattern Recognit.*, 2020, pp. 2482–2491.

[35] H. Caesar *et al.*, "nuScenes: A multimodal dataset for autonomous driving," in *Proc. Comput. Vis. Pattern Recognit.*, 2020, pp. 11618–11628.

[36] J. Wang and E. Olson, "AprilTag 2: Efficient and robust fiducial detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 4193–4198.

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

[38] J. Rehder, J. Nikolic, T. Schneider, T. Hinzmann, and R. Siegwart, "Extending kalibr: Calibrating the extrinsics of multiple IMUs and of individual axes," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2016, pp. 4304–4311.

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

[40] Y. Yang, A. Wong, and S. Soatto, "Dense depth posterior (ddp) from single image and sparse range," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 3348–3357.

[41] A. Eldesokey, M. Felsberg, and F. S. Khan, "Confidence propagation through CNNs for guided sparse depth regression," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2423–2436, Oct. 2019.

[42] F. Ma, G. V. Cavalheiro, and S. Karaman, "Self-supervised sparse-to-dense: Self-supervised depth completion from lidar and monocular camera," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 3288–3295.

[43] Y. Xu, X. Zhu, J. Shi, G. Zhang, H. Bao, and H. Li, "Depth completion from sparse lidar data with depth-normal constraints," in *Proc. Int. Conf. Comput. Vis.*, 2019. pp. 2811–2820.

[44] A. Paszke *et al.*, "Automatic differentiation in PyTorch," in *Proc. Conf. Neural Inf. Process. Syst. Autodiff Workshop*, 2017.