IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 16, 2021, accepted September 26, 2021, date of publication October 4, 2021, date of current version October 12, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3117710

Non-Learning Stereo-Aided Depth Completion Under Mis-Projection via Selective Stereo Matching

YASUHIRO YAO[1,2], RYOICHI ISHIKAWA[1], SHINGO ANDO[2], KANA KURATA[2], NAOKI ITO[2], JUN SHIMAMURA[2], AND TAKESHI OISHI[1], (Member, IEEE)

[1]Institute of Industrial Science, The University of Tokyo, Tokyo 153-0041, Japan
[2]NTT Human Informatics Laboratories, Yokosuka 239-0847, Japan

Corresponding author: Yasuhiro Yao (yao@cvl.iis.u-tokyo.ac.jp)

ABSTRACT We propose a non-learning depth completion method for a sparse depth map captured using a light detection and ranging (LiDAR) sensor guided by a pair of stereo images. Generally, conventional stereo-aided depth completion methods have two limiations. (i) they assume the given sparse depth map is accurately aligned to the input image, whereas the alignment is difficult to achieve in practice; (ii) they have limited accuracy in the long range because the depth is estimated by pixel disparity. To solve the abovementioned limitations, we propose selective stereo matching (SSM) that searches the most appropriate depth value for each image pixel from its neighborly projected LiDAR points based on an energy minimization framework. This depth selection approach can handle any type of mis-projection. Moreover, SSM has an advantage in terms of long-range depth accuracy because it directly uses the LiDAR measurement rather than the depth acquired from the stereo. SSM is a discrete process; thus, we apply variational smoothing with binary anisotropic diffusion tensor (B-ADT) to generate a continuous depth map while preserving depth discontinuity across object boundaries. Experimentally, compared with the previous state-of-the-art stereo-aided depth completion, the proposed method reduced the mean absolute error (MAE) of the depth estimation to 0.65 times and demonstrated approximately twice more accurate estimation in the long range. Moreover, under various LiDAR-camera calibration errors, the proposed method reduced the depth estimation MAE to 0.34-0.93 times from previous depth completion methods.

INDEX TERMS Computer vision, depth completion, LiDAR, sensor fusion, stereo matching.

I. INTRODUCTION

Depth measurement is conducted in several ways such as time-of-flight (ToF), stereo cameras, and structured light projection [1]. Stereo cameras and structured light projection estimate depth by pixel disparity. Hence their precision dramatically reduces as the distance increases since a small disparity change indicates a large depth change in the long range. In comparison, ToF sensors have a higher precision in the long range. Among ToF sensors, light detection and ranging (LiDAR) is used in various systems that require adaptability to dynamic environments, e.g., automated driving and robots, because of its active sensing capability and robustness to environmental changes. However, in terms of measurement density, LiDAR has a limitation because of the number of lasers in its array and the narrow beam measurement.

An alternative to address the sparsity of LiDAR is the depth completion, and the most common approach uses a single synchronized image as a guide [2]–[22]. These methods generate a sparse depth map by projecting LiDAR points onto the image, and then the depth map is completed using pixel intensities. Such image-aided depth completion methods are extensively studied and range from non-learning to supervised methods.

An important issue in the depth completion is mis-projection where LiDAR points are projected onto different objects in the image. Mis-projection often occurs because of spatial and temporal displacement of sensors, dynamic objects, decalibration, and calibration errors. Because camera and LiDAR are typically placed at different position, occlusion is unavoidable for near or dynamic objects. The temporal

The associate editor coordinating the review of this manuscript and approving it for publication was Sudipta Roy.

displacement of each LiDAR beam also causes errors without precise synchronization. Decalibration appears at run time because of oscillations of the vehicle or other mechanical reasons. Furthermore, the extrinsic calibration error between sensors results in mis-projection over the entire image.

Generally, extrinsic LiDAR and camera calibration is difficult because of their different modalities. To build the KITTI dataset [23], Geiger *et al.* calibrated LiDAR and cameras using multiple calibration boards in a controlled garage environment [24] followed by the manual selection of corresponding points. Although marker-less calibration methods have been examined [25]–[27], it remains difficult to stably realize accurate extrinsic calibration in uncontrolled environments.

In recent studies, stereo images have been used for depth completion to solve the mis-projection issue rather than a single image. This is because stereo camera systems are widely available and such systems can perceive 3D information with the help of stereo matching algorithms. For example, a self-supervised method is applicable under mis-projection caused by displacements of sensors [28]. However, this method still requires a dataset measured with a well-calibrated LiDAR-stereo system to train the neural network (NN). Furthermore, this method estimates depth by pixel disparity and suffers from low depth precision in the long range.

Therefore, in this study, we propose a non-learning depth completion method for a stereo-LiDAR system that is effective for the long range and robust to mis-projection. The proposed method comprises two techniques, i.e., selective stereo matching (SSM) and binary anisotropic diffusion tensor (B-ADT) [8]-aided smoothing. An important proposal is SSM, which searches for an optimal depth value for each pixel from its neighborly projected LiDAR points using an energy minimization framework (Fig. 1). This energy minimization approach can handle any type of mis-projection. Furthermore, SSM directly uses LiDAR depths and is advantageous in long-range accuracy. SSM is discrete optimization; thus, we apply B-ADT-aided smoothing for continuous depth estimation while preserving discontinuity between different objects.

Our contributions are summarized as follows.

- We propose SSM, which performs stereo matching in a selective manner to upsample LiDAR depths while maintaining the depth precision of LiDAR in the long range and considering mis-projection of LiDAR points.
- We propose a non-learning depth completion framework that combines SSM and B-ADT-aided smoothing. The framework achieves boundary-aware continuous depth estimation in addition to the SSM effects (long-range depth accuracy and robustness to mis-projection).

The rest of the paper is organized as follows. Section II reviews the related works and limitations of existing depth completion methods. Section III explains the proposed method. Section IV shows the evaluations. Section V gives the conclusion and summarizes limitations and future works.

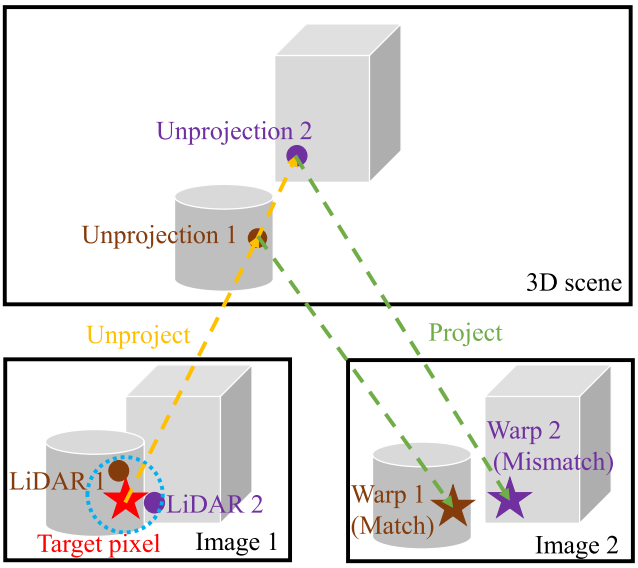


FIGURE 1. Conceptual diagram of SSM. SSM selects the most appropriate depth for each pixel from its nearby LiDAR projections within a search radius based on stereo correspondence and smoothness. The diagram ignores the smoothness for simplicity. Here, two depths (LiDAR 1 and 2) exist within the search radius from the target pixel in image 1. SSM warps the target pixel to image 2 using the depths of LiDAR 1 and 2. Then, SSM evaluates the correspondence of image 1 at the target pixel and image 2 at the warps (Warp 1 and 2). Here, the correspondence is higher for Warp 1, in other words, when LiDAR 1 is selected.

II. RELATED WORK

A. STEREO MATCHING

Stereo matching is extensively studied for 3D scanning because of the availability of stereo camera systems. The methods span from non-learning [29]–[34] to NN-based self-supervised [35] and supervised [36] methods.

In terms of accuracy, the supervised methods perform the best among them. The supervised methods also have the potential to perform in the long range because the precision of the ground truth disparity, which is usually made by other sensors such as LiDAR, is high and sub-pixel-level accurate. However, it is challenging to prepare a large dataset with ground truth disparity to train NNs.

Non-learning and self-supervised methods have difficulties achieving high precision in the long range because their depth estimation is limited by pixel-level stereo matching. As mentioned, a small disparity change indicates a large depth change in the long range. Therefore, there is uncertainty in the depth estimation even if the matching is accurate at the pixel level. Moreover, stereo matching methods still suffer from challenging scenarios such as repetitive pattern, low texture, discontinuity to cause occlusion, and specular reflection conditions.

B. SINGLE-IMAGE-AIDED DEPTH COMPLETION

Depth completion methods generate high-resolution and dense depth maps from sparse or low-resolution depth maps captured using LiDAR or depth cameras. The most common approach uses a single image as guidance. Kopf *et al.* [2]

proposed a method to interpolate low-resolution depth values based on the joint distance of color and space in a high-resolution image. Diebel and Thrun performed upsampling using a Markov random field (MRF) formulation [4]. In this method, the smoothness term is weighted as per texture derivatives; however, the results suffer from surface over-flattening. To address this issue, Ferstl *et al.* formalized depth completion into ADT-aided and TGV-regularized energy minimization [3]; and their method has been successfully used to smooth and optimize depth maps in more recent methods [37], [38]. Recently, Yao *et al.* proposed B-ADT to achieve depth completion to preserve discontinuity between different objects [8].

In addition, NNs have been applied to depth completion tasks. The most common approach is to train networks with ground truth dense depth maps [9]–[22]. Recently, self-supervised and semi-supervised methods have been examined because it is difficult to acquire the dense ground truth. Ma *et al.* [6] and Wong *et al.* [7] proposed methods with NNs that can be self-supervised using monocular camera frames and sparse depth maps from LiDAR with motion. Yang *et al.* proposed a method that can train a NN by the likelihood of the observed sparse point cloud under a hypothesized depth map [39].

A major limitation of the single-image-aided depth completion is that mis-projection of LiDAR points is not considered.

C. STEREO-AIDED DEPTH COMPLETION

Stereo images have been used as guides to complete the sparse measurements of LiDAR. These methods have been developed based on stereo matching, and they perform dense stereo matching using the accurate sparse depth value by LiDAR as a clue.

Badino *et al.* used LiDAR measurements to reduce the search space for stereo matching and provided predefined paths for dynamic programming [40]. Maddern *et al.* proposed a probabilistic model to fuse LiDAR and disparities by combining prior from each sensor [41], and Park *et al.* used NNs to learn such a model, which takes two disparities as input, i.e., one from the interpolated LiDAR and the other from semi-global matching [42]. Choe *et al.* recently proposed a geometry-aware stereo-LiDAR fusion network for long-range depth estimation [43]. As the same as single-image-aided methods, these methods do not consider mis-projection in given sparse depth maps.

Several recent methods have attempted to infer dense disparity maps from inaccurately projected LiDAR points with the help of stereo images. For example, Cheng *et al.* proposed a self-supervised method to train a NN to remove occluded background projection of LiDAR points to infer dense disparity maps [28]; however, this method does not handle incorrect projection caused by extrinsic calibration errors between the LiDAR and camera. Park *et al.* proposed a supervised method to train a NN to infer dense disparity maps from LiDAR inputs with extrinsic calibration errors between LiDAR and the camera [44]; however, this method requires accurately calibrated LiDAR and cameras to acquire effective training data.

Furthermore, the previous non-learning and self-supervised methods [28], [40], [41] estimate the depth by pixel disparity; thus, their depth precision is limited in the long range.

In summary, there are two major limitations in existing stereo-aided depth completion methods.

- These methods require accurate LiDAR-camera extrinsic calibration at some point in their process, which is often difficult to realize.
- The precision in the depth estimation is dramatically reduced as the distance increases because of the nature of disparity estimation using images.

The cost of the proposed method is similar to the previous studies [28], [29], whereas the approach to minimize the cost is different. The proposed method searches the minimizer by the selection from projected LiDAR depth values. The approach can handle any type of mis-projection without requiring accurate LiDAR-camera extrinsic calibration in any part of the process. Moreover, this selective approach has an advantage in the long-range precision because it directly uses LiDAR depth values.

III. PROPOSED METHOD

As shown in Fig. 2, the proposed method applies SSM followed by B-ADT-aided smoothing [8]. SSM is a discrete optimization, and its output is discrete; thus, smoothing improves the quality of the result. B-ADT allows us to incorporate boundary-direction-aware discontinuity in a variational approach. Both SSM and B-ADT-aided smoothing; thus, the proposed method as a whole, preserve depth discontinuity between different objects.

We give the problem statement in Section III-A, introduce SSM in Section III-B, explain B-ADT-aided smoothing in Section III-C, and describe a practical parameter tuning approach for SSM in Section III-D.

A. PROBLEM STATEMENT

Our problem settings assume a stereo camera and LiDAR are used to capture the scene; however, the camera and LiDAR calibration contains errors. Such conditions are possibly occur because of the difficulty associated with calibration, particularly when a calibration target is not available. Our aim is to estimate a dense depth map that is aligned with the image. Using mathematical notations, the target problem is defined as follows.

We are given a pair of stereo images ($I_1 : \Omega_1 \to \mathbb{R}$, $I_2 : \Omega_2 \to \mathbb{R}$) and a sparse inverse depth map captured by LiDAR ($D_S : \Omega_1 \to \mathbb{R} \bigcup \{\phi\}$) with $\Omega_1 \subset \mathbb{R}^2$ and $\Omega_2 \subset \mathbb{R}^2$ being the image domains and ϕ indicating an empty depth. Here, D_S is determined by projecting the LiDAR points to the image I_1; however, D_S is not accurately aligned with I_1 because we expect LiDAR-camera miscalibration and occlusions. Our aim is to derive a dense inverse depth map

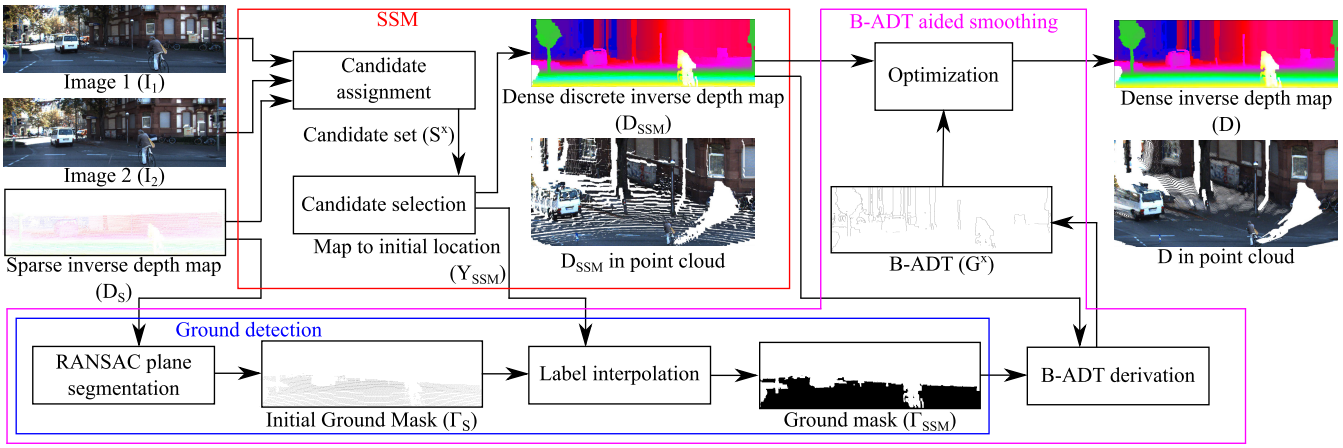


FIGURE 2. Proposed framework. SSM is composed of candidate assignment and candidate selection via optimization. The candidate assignment takes stereo images (I_1 and I_2) and a sparse inverse depth map (D_S) to derive a candidate set (S^x). The candidate selection via optimization derives a dense discrete inverse depth map (D_{SSM}) and a map to the initial inverse depth location (Y_{SSM}). B-ADT-aided smoothing first performs the ground detection to derive the ground mask (Γ_{SSM}), then derives B-ADT(G^x), and finally performs optimization to derive a dense inverse depth map (D).

$D : \Omega_1 \to \mathbb{R}$ aligned with the input image I_1. Throughout this paper, we normalize I_1 and I_2 to the range [0, 1], and the unit of the depth is meter.

Note that we derive an inverse depth map D rather than directly deriving the depth map. This is performed to balance the contribution of both near and distant depths [8], [45]. Deriving a dense inverse depth map D is equivalent to deriving a dense depth map D^{-1} or dense disparity map D'. The conversions at $\mathbf{x} \in \Omega_1$ are given as Eq. (1), (2) using camera focal length f and stereo baseline b.

$$D^{-1}(\mathbf{x}) = D(\mathbf{x})^{-1} \tag{1}$$

$$D'(\mathbf{x}) = fbD(\mathbf{x}) \tag{2}$$

The proposed method is applicable to motion stereo images as input as in the evaluation using the Komaba dataset (Section IV-B).

In this paper, we use $|\cdot|$ to denote the vector norm. In particular, given a vector $\mathbf{p} \in \mathbf{R}^K$ with K being the arbitrary number of dimensions, the norm is given as follows:

$$|\mathbf{p}| = \sqrt{\sum_i^K p_i{}^2} \tag{3}$$

B. SELECTIVE STEREO MATCHING

SSM searches the most appropriate inverse depth value for every $\mathbf{x} \in \Omega_1$ from its neighborly projected LiDAR points. SSM comprises the candidate assignment and candidate selection steps. In the candidate assignment step, each pixel in the image is assigned a set of LiDAR inverse depth values. In the candidate selection step, SSM selects the most appropriate value from the candidate set using an energy minimization framework. Here, the energy is defined as the sum of the stereo matching cost and the smoothness regularization term.

Implementation-wise, both the candidate assignment and the candidate selection are composed of pixel-wise calculations which are parallelized on GPU.

FIGURE 3. Candidate assignment of SSM. The circles indicate the areas to create the candidate sets for the centered pixels. Although there is mis-projection, the appropriate LiDAR values are located in the areas, e.g., purple for the pole at the left, and yellow for the person at the right.

1) CANDIDATE ASSIGNMENT

a: INITIAL MAP OF CANDIDATE SETS

First, SSM constructs a candidate set $S^{\mathbf{x}}$ ($\mathbf{x} \in \Omega_1$), which is a set of inverse depth values in the surrounding pixels within a pre-defined radius r from $\mathbf{x}$ (Eq. (4)), as shown in Fig. 3.

$$S^{\mathbf{x}} = \{d \mid d = D_S(\mathbf{y}) \wedge d \neq \phi \text{ where } |\mathbf{y} - \mathbf{x}| < r\} \tag{4}$$

Note that we introduce an empirical approach to set r in Section III-D. If the cardinality of $S^{\mathbf{x}}$ is less than predefined threshold m, the set is assumed to be empty ($S^{\mathbf{x}} = \emptyset$) to avoid selecting a value from a small number of candidates (we used $m = 4$ in our evaluations).

b: CANDIDATE SET INTERPOLATION

To fill the pixels with non-empty candidate sets, we interpolate the candidate sets using image-guided nearest neighbor search (IGNNS) [8]. IGNNS searches the nearest neighbor by the cumulative distance of image gradients. Here, let Ψ be the set of pixels where the candidate set is empty ($\Psi = \{\mathbf{x} \mid S^{\mathbf{x}} = \emptyset\}$), and let $\bar{\Psi}$ be the complement of Ψ ($\bar{\Psi} = \Omega_1 \backslash \Psi$). We search an image-guided nearest neighbor of every $\mathbf{x} \in \Psi$ from $\bar{\Psi}$ (denoted $\mathbf{z}$) and update the candidate set by $S^{\mathbf{x}} = S^{\mathbf{z}}$.

We derive $\mathbf{z}$ as Eq. (5) by denoting $\pi(\bar{\mathbf{x}}, \mathbf{x})$ being the set of pixels on the grid path from a surrounding non-empty pixel $\bar{\mathbf{x}} \in \bar{\Psi}$ to $\mathbf{x}$.

$$\mathbf{z} = \arg\min_{\bar{\mathbf{x}} \in \bar{\Psi}} \min_{\pi(\bar{\mathbf{x}}, \mathbf{x})} \sum_{\mathbf{y} \in \pi(\bar{\mathbf{x}}, \mathbf{x})} \left\{ |\nabla I_1(\mathbf{y})|^2 + c \right\}, \tag{5}$$

where c is the constant cost of the unit path length, and we set $c = 0.04$ in our evaluations following the parameter study in the literature [8].

c: CORRESPONDENCE SEARCH

Following candidate set interpolation, we identify the correspondence $\mathbf{x}'(d) \in \Omega_2$ of $\mathbf{x} \in \Omega_1$. $\mathbf{x}'(d)$ is the location on I_2 where $\mathbf{x}$ on I_1 is warped with inverse depth value d. We calculate $\mathbf{x}'(d)$ for all combinations of $\mathbf{x}$ and $d \in S^{\mathbf{x}}$. Below we denote $\mathbf{x} = (x_0\, x_1)^T$.

If I_1 and I_2 are a pair of rectified binocular stereo images, using the floor function denoted as $\lfloor \cdot \rfloor$, $\mathbf{x}'(d)$ is derived with camera focal length f and baseline b as follows:

$$\mathbf{x}'(d) = \begin{pmatrix} \lfloor x_0 - fbd \rfloor \\ x_1 \end{pmatrix}. \tag{6}$$

For motion stereo images, $\mathbf{x}'(d)$ is calculated using the camera intrinsic parameter ($K \in \mathbb{R}^{3\times3}$), rotation ($R_C \in SO(3)$), and the translation ($t_C \in \mathbb{R}^3$) of the camera motion as follows:

$$\mathbf{x}'(d) = \begin{pmatrix} \lfloor \tilde{x}'_0(d)/\tilde{x}'_2(d) \rfloor \\ \lfloor \tilde{x}'_1(d)/\tilde{x}'_2(d) \rfloor \end{pmatrix}, \quad \text{where} \begin{pmatrix} \tilde{x}'_0(d) \\ \tilde{x}'_1(d) \\ \tilde{x}'_2(d) \end{pmatrix} = K \left(R_C K^{-1} \begin{pmatrix} x_0 \\ x_1 \\ d^{-1} \end{pmatrix} + \mathbf{t}_C \right). \tag{7}$$

If there are two or more $d \in S^{\mathbf{x}}$ to derive the same $\mathbf{x}'(d)$, we only maintain the nearest from $\mathbf{x}$ among those in $S^{\mathbf{x}}$. Note that this pruning process is performed to realize computational efficiency of the following optimization process.

2) CANDIDATE SELECTION VIA OPTIMIZATION

a: STEREO MATCHING COST

The stereo cost $L^{\mathbf{x}}(d)$ evaluates the consistency of the inverse depth d and the pair of input images at location $\mathbf{x} \in \Omega_1$.

Similar to the literature [28], we compose the stereo cost $L^{\mathbf{x}}(d)$ using the sum of the photometric loss ${L_P}^{\mathbf{x}}(d)$, the census loss ${L_C}^{\mathbf{x}}(d)$, and the image gradient loss ${L_G}^{\mathbf{x}}(d)$ with weights α_C and α_G as follows:

$$L^{\mathbf{x}}(d) = {L_P}^{\mathbf{x}}(d) + \alpha_C {L_C}^{\mathbf{x}}(d) + \alpha_G {L_G}^{\mathbf{x}}(d). \tag{8}$$

Here, ${L_P}^{\mathbf{x}}(d)$, ${L_C}^{\mathbf{x}}(d)$, and ${L_G}^{\mathbf{x}}(d)$ are calculated using the warped coordinates $\mathbf{x}'(d)$ in Eq. (6) or (7) with the predefined window W as follows:

$${L_P}^{\mathbf{x}}(d) = \sum_{\delta \in W} \min\left(\left| I_1(\mathbf{x} + \delta) - I_2\left(\mathbf{x}'(d) + \delta\right) \right|, l_P \right) \tag{9}$$

$${L_C}^{\mathbf{x}}(d) = \min\left(\left\| C_1(\mathbf{x}) - C_2\left(\mathbf{x}'(d)\right) \right\|_H, l_C \right) \tag{10}$$

$${L_G}^{\mathbf{x}}(d) = \sum_{\delta \in W} \min\left(\left| \nabla I_1(\mathbf{x} + \delta) - \nabla I_2\left(\mathbf{x}'(d) + \delta\right) \right|, l_G \right), \tag{11}$$

where C_1 and C_2 respectively represent the census transformation of I_1 and I_2 with window W, $\| \cdot \|_H$ denotes the Hamming distance, and l_P, l_C, and l_G are the maximum cost values. We set the window W to be an 11×11 square centered at $\mathbf{x}$. In our evaluations, we set $\alpha_C = \alpha_G = 1$ and $l_P = l_C = l_G = 0.5$.

b: ENERGY DEFINITION

SSM searches the optimal depth value from $S^{\mathbf{x}}$ for every $\mathbf{x} \in \Omega_1$, which is performed using an energy minimization. Here, the energy follows the conventional stereo disparity estimation [29]. We construct an MRF whose nodes are $\mathbf{x} \in \Omega_1$, and the edges $\mathcal{E}$ comprise all the pairs of adjacent pixels. The energy E_{SSM} is defined by the addition of the stereo matching cost $L^{\mathbf{x}}(d)$ defined in Eq. (8) and a smoothness regularization term for the inverse depth as follows:

$$E_{\mathrm{SSM}} = \sum_{\mathbf{x} \in \Omega_1} L^{\mathbf{x}}(d) + \lambda_{\mathrm{SSM}} \sum_{e \in \mathcal{E}} \min\left(|\Delta_e d|, l_d \right), \quad \text{with } d \in S^{\mathbf{x}}. \tag{12}$$

Here, Δ_e represents taking the difference across the edge e, λ_{SSM} is the regularization weight, and l_d is the maximum smoothness term. We empirically set $\lambda_{SSM} = 10^2$.

c: OPTIMIZATION

SSM derives a discrete inverse dense depth map (D_{SSM}) by minimizing the energy (E_{SSM}) in Eq. (12).

The minimization of E_{SSM} is an optimization of MRF, which we solve by Loopy Belief Propagation (LBP) [46]. In particular, by setting $X \subset \Omega_1$ as the set of four adjacent pixels of $\mathbf{x} \in \Omega_1$, we iteratively update the message from $\mathbf{x}$ to one of its adjacent pixels $\mathbf{y} \in X$ by the min-sum algorithm as shown in Eq. (13) and (14). Here, we denote the iteration index as n, the normalized message from $\mathbf{x}$ to $\mathbf{y}$ as $\mathrm{msg}^n_{\mathbf{x}\to\mathbf{y}}(d)$, and the message prior to normalization as $\overline{\mathrm{msg}}^n_{\mathbf{x}\to\mathbf{y}}(d)$.

$$\overline{\mathrm{msg}}^n_{\mathbf{x}\to\mathbf{y}}(d) = \min_{d' \in S^{\mathbf{x}}} L^{\mathbf{x}}\left(d'\right) + \lambda_{\mathrm{SSM}} \min\left(\left| d - d' \right|, l_d \right) + \sum_{\mathbf{z} \in X \setminus \mathbf{y}} \mathrm{msg}^{n-1}_{\mathbf{z}\to\mathbf{x}}(d') \tag{13}$$

$$\mathrm{msg}^n_{\mathbf{x}\to\mathbf{y}}(d) = \overline{\mathrm{msg}}^n_{\mathbf{x}\to\mathbf{y}}(d) - \log \sum_{d \in S^{\mathbf{x}}} \exp\left(\overline{\mathrm{msg}}_{\mathbf{x}\to\mathbf{y}}(d) \right). \tag{14}$$

Denoting the message after convergence as msg^∞, the optimal inverse depth value $d^{\mathbf{x}}_{\mathrm{SSM}}$ at $\mathbf{x}$ is expressed as follows:

$$d^{\mathbf{x}}_{\mathrm{SSM}} = \arg\min_{d \in S^{\mathbf{x}}} \left\{ L^{\mathbf{x}}(d) + \sum_{\mathbf{y} \in X} \mathrm{msg}^\infty_{\mathbf{y}\to\mathbf{x}}(d) \right\}. \tag{15}$$

The output inverse depth map D_{SSM} is assigned based on the optimal values as Eq. (16).

$$D_{\mathrm{SSM}}(\mathbf{x}) = d^{\mathbf{x}}_{\mathrm{SSM}} \tag{16}$$

D_{SSM} is visually shown in Fig. 2.

In addition, for the ground mask creation in later process (Section III-C1), we construct a map $Y_{\mathrm{SSM}} : \Omega_1 \to \Omega_1$ to indicate the original location of the inverse depth map. Because D_{SSM} is created by the selection, we know where each value of D_{SSM} initially located in D_S. In particular, if the value of D_{SSM} at $\mathbf{x} \in \Omega_1$ is originally at $\mathbf{y} \in \Omega_1$ in D_S, we set $Y_{\mathrm{SSM}} = \mathbf{y}$. By using equations, this assignment is expressed as Eq. (17).

$$\begin{aligned} Y_{\mathrm{SSM}}(\mathbf{x}) &= \mathbf{y} \\ \text{such that } D_S(\mathbf{y}) \in S^{\mathbf{x}} &\wedge D_S(\mathbf{y}) = D_{\mathrm{SSM}}(\mathbf{x}). \end{aligned} \tag{17}$$

C. B-ADT AIDED SMOOTHING

D_{SSM} is discrete because it is generated by the selection from a finite number of candidates. Here, we apply B-ADT weighted TGV smoothing from [8] to derive smooth depth with discontinuity preservation at boundaries. Below, we explain the ground detection to create the filter for B-ADT derivation, the B-ADT derivation, and the optimization. Implementation-wise, the ground detection is the RANSAC plane segmentation, B-ADT derivation is a single-step calculation, and the optimization is iterative pixel-wise and parallelized on the GPU.

1) GROUND DETECTION

We create a ground mask to filter out occlusion boundaries that are faultily detected on the ground.

First, we detect the ground points for the input LiDAR depth map. We convert the LiDAR depth map to the point cloud and apply the RANSAC plane segmentation [48]. The RANSAC plane segmentation iteratively searches the coefficients of a plane having the maximum number of inlier points within the given threshold d_{rand}, by randomly sampling three points from the point cloud to derive the plane coefficients in every iteration. For RANSAC parameters, we set $d_{\mathrm{rand}} = 0.2$ [m] and the number of iterations as 100.

Then, we project the inlier points of the derived plane to the image domain Ω_1 and acquire the ground mask $\Gamma_S : \Omega_1 \to \{0, 1\}$, where $\Gamma_S(\mathbf{x}) = 1$ if $\mathbf{x}$ is the ground.

Finally, we create a dense ground mask $\Gamma_{\mathrm{SSM}} : \Omega_1 \to \{0, 1\}$ which is aligned with D_{SSM}. Γ_{SSM} is derived by interpolating Γ_S based on the selection performed in SSM. In particular, using Y_{SSM} in Eq. (17), Γ_{SSM} is assigned as follows:

$$\Gamma_{\mathrm{SSM}}(\mathbf{x}) = \Gamma_S(Y_{\mathrm{SSM}}(\mathbf{x})) \tag{18}$$

In Fig 2, Γ_S and Γ_{SSM} are visualized by indicating the ground pixels in black.

2) B-ADT DERIVATION

B-ADT is pixel-wise weighting for the variational regularization term. Here, B-ADT is derived based on the occlusion boundary conditions in D_{SSM} and the ground mask Γ_{SSM}. Occlusion boundaries are boundaries where objects are not in contact; thus, the depth values at the occlusion boundaries immediately change.

The B-ADT for each pixel is assigned based on the following two conditions: A, i.e., the pixel is on a vertical occlusion boundary, and B, i.e., the pixel is on a horizontal occlusion boundary. Here, a vertical occlusion boundary is a vertical line segment across which the depth is horizontally discontinuous, and a horizontal occlusion boundary is a horizontal line segment across which the depth is vertically discontinuous.

In particular, with predefined threshold t, we determine a pixel $\mathbf{x} \in \Omega_1$ is in A if $\left|\partial_x D^{-1}_{\mathrm{SSM}}(\mathbf{x})\right| > t$, and in B if $\left|\partial_y D^{-1}_{\mathrm{SSM}}(\mathbf{x})\right| > t$. To make occlusion boundaries where the adjacent depths change more than 2 [m], we used $t = 2$ in our evaluations. Because images are defined on 2D grids, every pixel belongs to one of four sets, i.e., neither A nor B ($\bar{A} \cap \bar{B}$), A but not B ($A \cap \bar{B}$), not A but B ($\bar{A} \cap B$), and A and B ($A \cap B$).

Boundary detection by a single threshold can be faulty, particularly in the ground region because the ground is often a large plane parallel to the view direction with a wide depth range. Thus, we filter out occlusion boundaries that are detected on the ground by Γ_{SSM} in Eq. (18).

Finally, by denoting B-ADT at pixel $\mathbf{x} \in \Omega_1$ as $G^{\mathbf{x}}$, we set $G^{\mathbf{x}}$ based on the boundary conditions and the ground mask as follows:.

$$G^{\mathbf{x}} = \begin{cases} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & \text{if } \mathbf{x} \in \bar{A} \cap \bar{B} \text{ or } \Gamma_{\mathrm{SSM}}(\mathbf{x}) = 1 \\ \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} & \text{if } \mathbf{x} \in A \cap \bar{B} \text{ and } \Gamma_{\mathrm{SSM}}(\mathbf{x}) = 0 \\ \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} & \text{if } \mathbf{x} \in \bar{A} \cap B \text{ and } \Gamma_{\mathrm{SSM}}(\mathbf{x}) = 0 \\ \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} & \text{if } \mathbf{x} \in A \cap B \text{ and } \Gamma_{\mathrm{SSM}}(\mathbf{x}) = 0 \end{cases} \tag{19}$$

3) OPTIMIZATION

We minimize the energy with B-ADT weighted TGV regularization to acquire the output of the proposed framework. By denoting the inverse depth map during optimization as $u : \Omega_1 \to \mathbb{R}$ and the relaxation variable as $\mathbf{v} : \Omega_1 \to \mathbb{R}^2$, we define the energy E_{TGV} as the sum of the data term $C[u]$ and the smoothness term $R[u, \mathbf{v}]$ as follows:

$$E_{\mathrm{TGV}} = C[u] + R[u, \mathbf{v}] \tag{20}$$

$$C[u] = \int_{\Omega_1} w\,|u - D_{\mathrm{SSM}}|^2 \, d\mathbf{x} \tag{21}$$

$$R[u, \mathbf{v}] = \int_{\Omega_1} \lambda_A \left|G^{\mathbf{x}}(\nabla u - \mathbf{v})\right| + \lambda_B\,|\nabla \mathbf{v}|\, d\mathbf{x}, \tag{22}$$

where w is the pixel-wise weight for the data term, and λ_A and λ_B are weights for the energy terms. We set $w = D^{-2.5}_{\mathrm{SSM}}$, $\lambda_A = 1.0$, and $\lambda_B = 8.0$ based on the literature [8].

Note that E_{TGV} is convex, and we can derive the optimums of u and $\mathbf{v}$ using the primal dual algorithm [49]. In the end, the output of the proposed framework (a dense inverse depth

TABLE 1. Depth completion results on KITTI dataset with the accurate calibration.

Method	Input	*Supervised	Processing time [s]	Error rate [%]	MAE [m]
Hernandez et al. [30]	Stereo	No	0.003	6.59	0.977
Yamaguchi et al. [31]	Stereo	No	1.947	4.31	1.021
Kopf et al. [2]	Monocular + LiDAR	No	1.650	9.70	0.819
Ferstl et al. [3]	Monocular + LiDAR	No	0.064	7.45	0.578
Yao et al. [8]	Monocular + LiDAR	No	0.065	4.47	0.413
Maddern et al. [41]	Stereo + LiDAR	No	n/a	5.91	n/a
Ours (SSM only)	Stereo + LiDAR	No	0.968	3.43	0.399
Ours	Stereo + LiDAR	No	0.999	3.32	**0.356**
Park et al. [42]	Stereo + LiDAR	Yes	n/a	4.84	n/a
Cheng et al. [28]	Stereo + LiDAR	Yes	1.721	**2.17**	0.548

*'Yes" if the method requires accurate LiDAR camera extrinsic calibration parameters during training.

map D) is the optimum of u as Eq. (23).

$$D = \arg\min_{u} E_{\mathrm{TGV}}. \tag{23}$$

D. PARAMETER SETTING FOR SSM

SSM introduces a parameter r as the radius for candidate set search. Here, we present a practical approach to select the r value. r should be as small as possible to cover the nearest appropriate depth because the number of candidates increases as r increases, which generally leads to inappropriate selections. Furthermore, we consider two primary causes for mis-projection, i.e., LiDAR-camera calibration error and occlusion.

The mis-projection caused by calibration errors is primarily attributed to rotational errors. At the center of the image, projection error σ_{calib} caused by rotation error θ_{calib} can be calculated as follows:

$$\sigma_{\mathrm{calib}} = f \tan\theta_{\mathrm{calib}} \text{ [pixel]}, \tag{24}$$

where f is the camera focal length. Although the exact value of θ_{calib} cannot be known, it can be practically given in several ways, e.g., the error range presented in the reference of the original calibration method or by visually observing the LiDAR points projected onto the image.

To handle mis-projection by occlusion, all pixels typically should have several candidates in the range. Empirically, we found that this can be achieved when the radius is set to cover two scanlines. The pixel distance between two scanlines σ_{scan} is estimated using Eq. (24) with angle θ_{scan} between the scanlines as follows:

$$\sigma_{\mathrm{scan}} = f \tan\theta_{\mathrm{scan}} \text{[pixel]}. \tag{25}$$

We set the optimal radius r^* to be maximum of σ_{calib} and σ_{scan} to cover mis-projection caused by calibration errors and occlusion as follows:

$$r^* = \max\left(\sigma_{\mathrm{calib}}, \sigma_{\mathrm{scan}}\right) \text{[pixel]}. \tag{26}$$

IV. EVALUATION

We performed an evaluation that used the accurate LiDAR-camera extrinsic calibration (Section IV-A), another that used erroneous LiDAR-camera extrinsic calibration (Section IV-B), and the other for the parameter study (Section IV-C). In the first evaluation, we compared the accuracy of the proposed method to that of existing state-of-the-art methods under common experimental conditions [28], [41], [42]. Moreover, we analyzed the accuracy distribution over the depth range to demonstrate the advantage of the proposed method in the long range. In the second evaluation, we examined the robustness of the proposed method against LiDAR-camera extrinsic calibration errors. In this experiment, we used the KITTI [24] and Komaba datasets [38] with added calibration errors.

In all evaluations, we implemented SSM and B-ADT aided smoothing on GPU by CUDA, and used RANSAC plane segmentation from PCL library [50] for the ground detection.

A. EVALUATION WITH ACCURATE CALIBRATION

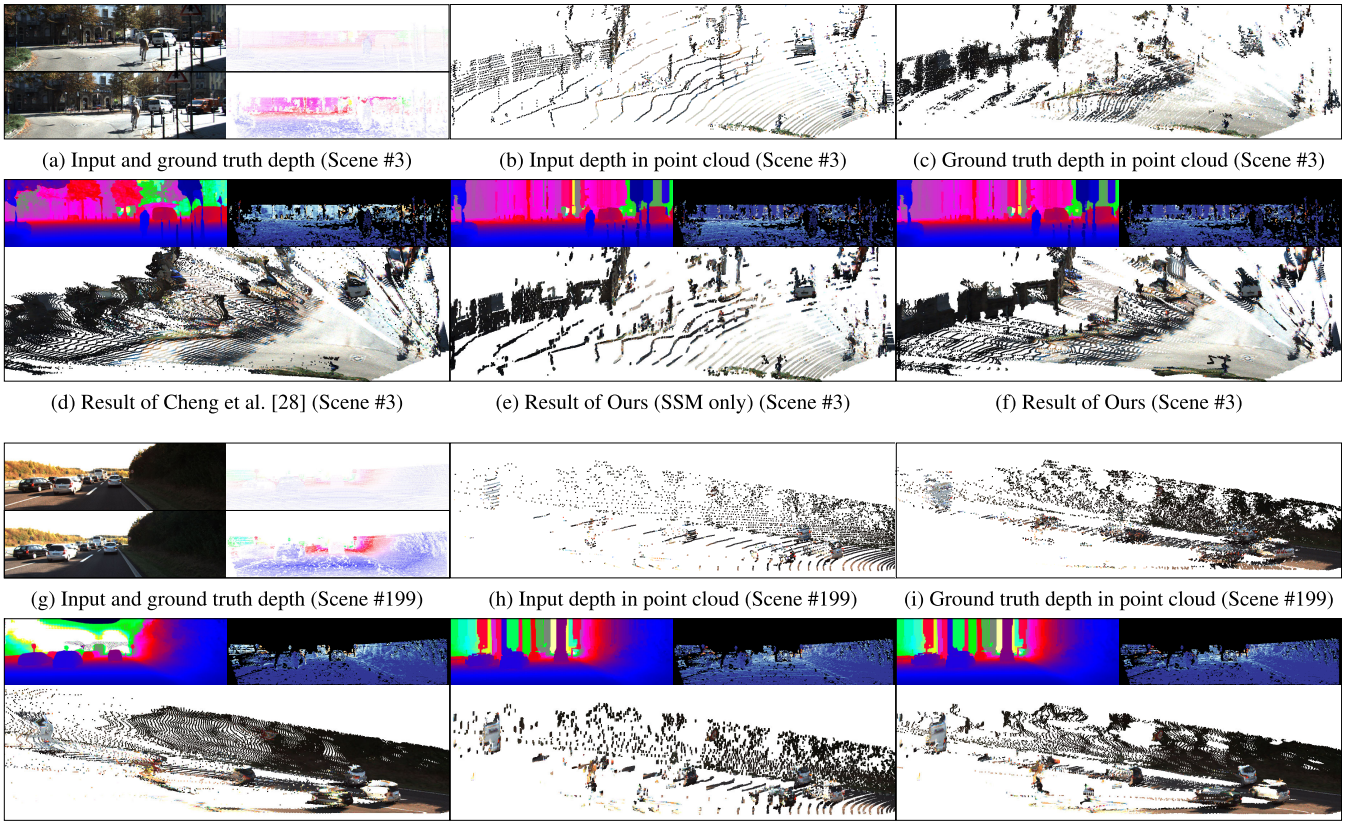
We evaluated the proposed method on a subset of the KITTI dataset, which is commonly used to evaluate stereo-LiDAR fusion [28], [41], [42]. These data comprise 141 sets of left and right images, sparse LiDAR depth maps, dense disparity maps, and dense depth maps. The figure of an example frame of the KITTI dataset is in the supplementary material. Here, we used the ground truths of the dense disparity map [47] and dense depth map [51] for the evaluation. Note that the input sparse depth maps still have mis-projection caused by occlusions, although the extrinsic calibration is accurate, as shown in Fig. 7 (a). In this evaluation, we set the radius for the candidate search to $r = 5$ [pixel] for our method.

We compared the proposed method to non-learning stereo methods [30], [31], non-learning single-image-aided depth completion methods [3], [8], non-learning stereo-aided depth completion methods [41], and supervised stereo-aided depth completion methods [28], [42]. Note that we assume Cheng's method [28] is a supervised because it uses an accurately calibrated dataset during training.

Implementation conditions are as follows. We used our own CUDA implementations for several methods [2], [3], [8]. We used the authors' implementation for the non-learning stereo methods [30], [31]. We used the authors' implementation and their trained model for Cheng's method [28]. We referred to the results in the original papers with the same experimental conditions for methods [41], [42].

1) OVERALL ACCURACY

Table 1 compares the accuracy of each method, and Fig. 4 shows the visualized results. This evaluation was based on



(a) Input and ground truth depth (Scene #3) (b) Input depth in point cloud (Scene #3) (c) Ground truth depth in point cloud (Scene #3)

(d) Result of Cheng et al. [28] (Scene #3) (e) Result of Ours (SSM only) (Scene #3) (f) Result of Ours (Scene #3)

(g) Input and ground truth depth (Scene #199) (h) Input depth in point cloud (Scene #199) (i) Ground truth depth in point cloud (Scene #199)

(j) Result of Cheng et al. [28] (Scene #199) (k) Result of Ours (SSM only) (Scene #199) (l) Result of Ours (Scene #199)

FIGURE 4. Inputs and results of depth completion on KITTI with the accurate calibration. The scene numbers are from the KITTI stereo training set [47]. (a), (g) Top left: the input left images, bottom left: input right images, top right: input sparse depth maps, bottom right: ground truth dense depth maps. (b), (h) The point clouds created from the input sparse depth maps. (c), (i) The point clouds created from the ground truth dense depth maps. (d), (e), (f), (j), (k), (l) Top left: the depth completion results, top right: the error maps, bottom: the point clouds generated from the depth completion results. As expected, SSM gives discrete point clouds, and our whole framework gives continuous point clouds. The surface shapes of the poles and walls in the background are preserved by the proposed method, while they are missing in the results obtained using Cheng's method [28]. In addition, the error maps demonstrate smaller errors of our method in the long range compared to Cheng's method [28].

the error rate measured with the dense disparity maps and the Mean Absolute Error (MAE) measured with the dense depth maps. Here, the error rate is defined as per the literature [47] and is the percentage of stereo disparity outliers that have errors greater than or equal to three pixels. The proposed method outperformed the compared methods in terms of MAE. Moreover, although the proposed method is a non-learning method, it demonstrated a competitive error rate compared to the supervised stereo-aided depth completion [28], [42].

In addition, Table 1 demonstrates the general advantage of LiDAR-aided methods, including our method, in relation to stereo-only methods [30], [31] in terms of depth accuracy. Furthermore, Fig. 5 shows the advantage of LiDAR-aided methods in challenging conditions as a repetitive pattern, low texture, discontinuity, and specular reflection.

2) LONG-RANGE ACCURACY

Figure 6 shows a breakdown of MAE against the depth range compared to Cheng's method [28]. As shown, the difference in MAE between the proposed method and Cheng's method [28] increased as the distance increased. Cheng's method estimate depth by pixel disparity, and its depth precision decreases as the distance increases. In contrast, the proposed method is based on the selection of LiDAR depths and does not lose the depth precision in the long range. The accuracy in the long range is also visible by the point clouds in Fig. 4. The method of Cheng *et al.* [28] lost the shapes of background objects, e.g., the poles, walls, and cars, whereas the shapes of these objects were retained in the results of the proposed method.

3) PROCESSING TIME

Table 1 also shows the processing time in our environment, which is a laptop computer running Intel Core i9 and GeForce RTX 2080. Although the processing time of our method is not in real time as methods [3], [8], [30], our method performed faster than the state-of-the-art non-learning stereo matching [31] and stereo-aided depth completion [28].

B. EVALUATIONS WITH CALIBRATION ERRORS

We evaluated the proposed method with LiDAR camera extrinsic calibration errors. Here, we applied random errors to the KITTI and Komaba datasets. This comparison was

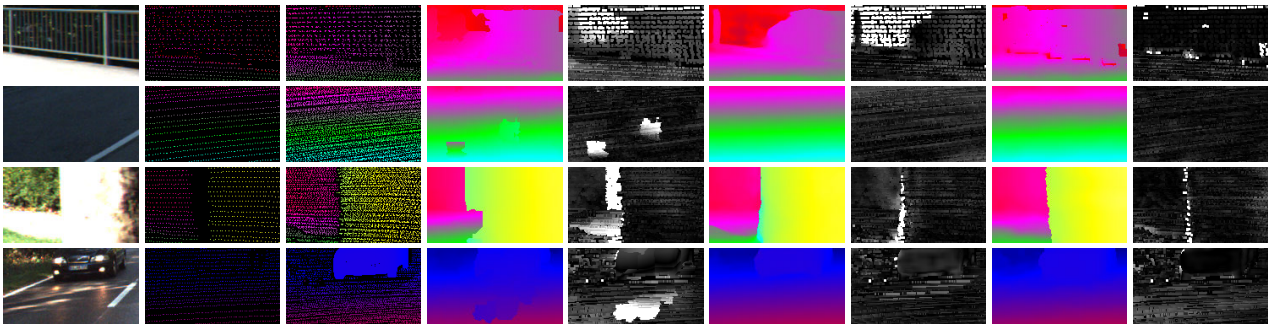


FIGURE 5. Disparity estimation in challenging conditions: from top to bottom, repetitive pattern, low texture, discontinuity, and specular reflection. (a) Input left images. (b) Input LiDAR disparity maps. (c) The ground truth dense disparity maps. (d), (e), (f) left: disparity estimations, right: error maps with white indicating large error. The stereo-only method [31] has larger disparity error than LiDAR-aided methods. Moreover, our method has less error in the repetitive pattern and discontinuity conditions than the state-of-the-art stereo-aided depth completion [28].

TABLE 2. Input calibration error details.

Dataset	error type	Rot. axis (x,y,z)	Rot. error [deg.]	Trans. direction (x,y,z)	Trans. error [m]
KITTI	*blueprint*	(0.04,-0.89,0.45)	0.952	(0.03,-0.05,-0.99)	0.076
	error-1 (avg.)*	(-0.79,0.44,0.43)	0.675	(0.51,0.54,0.67)	0.155
	error-2 (avg.)*	(-0.94,0.33,-0.03)	0.667	(0.36,0.83,0.42)	0.207
Komaba	-	(0.51,-0.11,0.85)	1.096	(-0.26, -0.73, 0.63)	0.207

* Averages are shown since every frame has different errors.

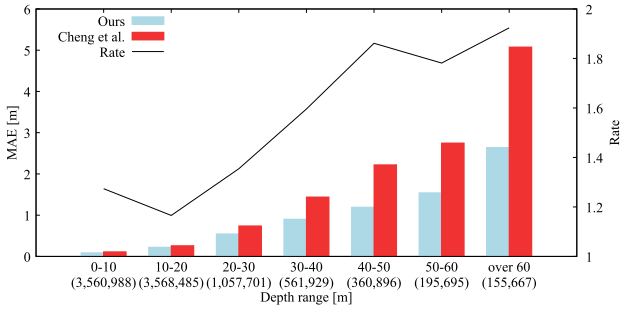


FIGURE 6. Plot of MAEs against depth ranges on KITTI with the accurate calibration. The numbers in parentheses indicate the number of occurrences. The proposed method obtained smaller MAE than Cheng's method [28] in all the ranges. Moreover, the proposed method approximately achieved half MAE in longer ranges than 40 [m].

FIGURE 7. Mis-projection in the KITTI dataset. (a) Mis-projection is caused by occlusion although the calibration is accurate. (b), (c), (d) Calibration errors cause mis-projection in addition to the occlusion (see the poles).

performed against unsupervised methods [2], [3], [8]. Supervised stereo-LiDAR fusion methods were not applied because accurately calibrated scans for training are not available. In this evaluation, the parameter settings were the same as those discussed in Section IV-A, except for the candidate search radius, which was set to $r = 15$ [pixel].

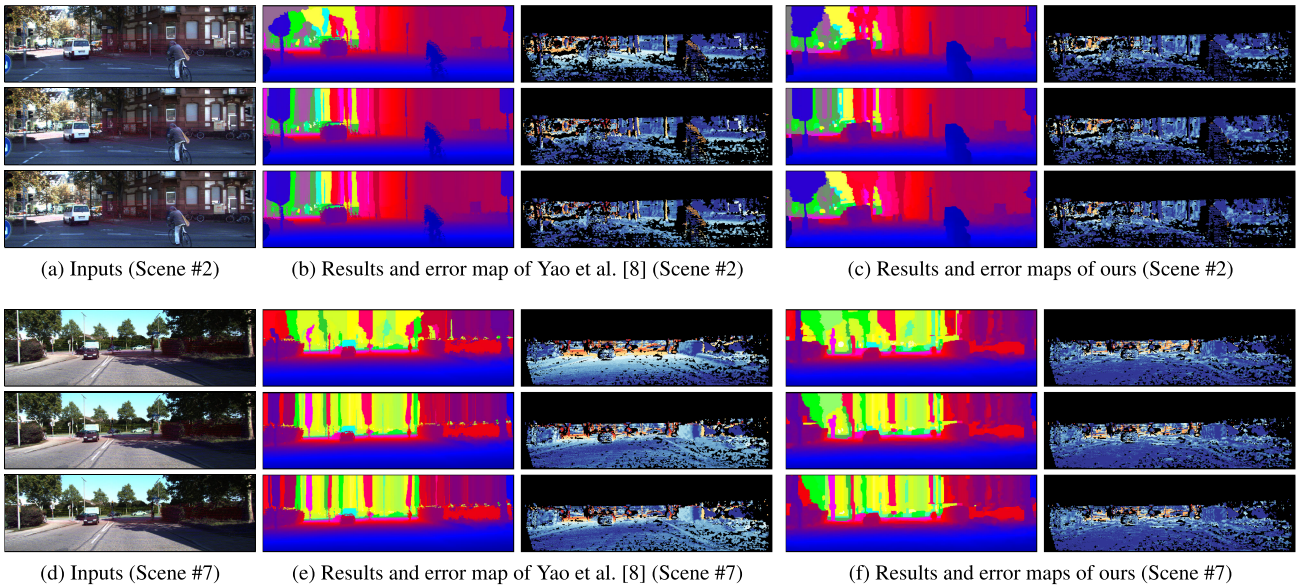
1) KITTI DATASET

We applied the following three error types to the KITTI dataset used in Section IV-A.

- *blueprint* represents the extrinsic parameters before calibration, derived by the sensor setup blueprint of the KITTI dataset.
- *error-1* represents parameters calibrated by a single-frame-marker-less method from the initial 2 [deg.] of rotation and 0.2 [m] of translation errors. After calibration, the average error was 0.675 [deg.] and 0.155 [m].
- *error-2* represents parameters calibrated by a single-frame-marker-less method from the initial 4 [deg.] of rotation and 0.4 [m] of translation errors. After calibration, the average error was 0.667 [deg.] and 0.207 [m].

The single-frame-marker-less calibration method to derive *error-1* and *error-2* is explained in the supplementary material. The intention of *error-1* and *error-2* is to emulate the worst-case calibration error expected in practical cases.

Table 2 shows the details of the errors, and Fig 7 shows the visual of the calibration error in the input data. Note that



(a) Inputs (Scene #2) (b) Results and error map of Yao et al. [8] (Scene #2) (c) Results and error maps of ours (Scene #2)

(d) Inputs (Scene #7) (e) Results and error map of Yao et al. [8] (Scene #7) (f) Results and error maps of ours (Scene #7)

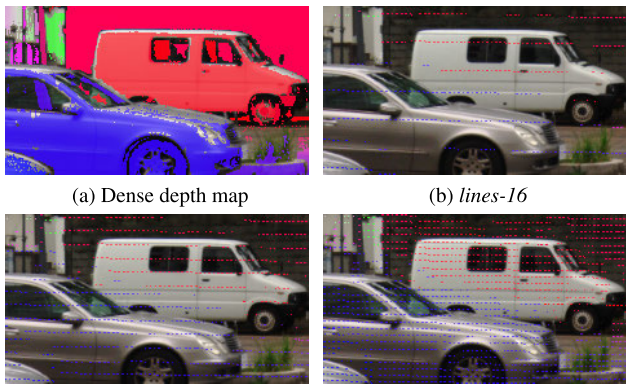
FIGURE 8. Inputs and results on KITTI dataset with calibration errors (top: *blueprint*; middle: *error-1*; and bottom: *error-2*). The scene numbers are from the KITTI stereo dataset [47]. Refer to Fig. 11 for the ground truths. (a), (d) The input sparse depth maps projected onto the input left image. (b), (c), (e), (f) The depth completion results and error maps. The areas of poles have smaller errors in (c), (f) compared to (b), (e).

TABLE 3. Depth completion results obtained on KITTI dataset with calibration errors.

Method	*blueprint*		*error-1*		*error-2*	
	Error rate [%]	MAE [m]	Error rate [%]	MAE [m]	Error rate [%]	MAE [m]
Kopf et al. [2]	9.93	1.584	15.65	1.094	14.97	1.083
Ferstl et al. [3]	9.72	1.600	15.17	1.025	14.34	1.022
Yao et al. [8]	7.66	1.576	12.91	1.023	12.08	0.986
Ours(SSM only)	4.53	0.577	4.90	0.577	5.00	0.584
Ours	**4.11**	**0.528**	**4.43**	**0.527**	**4.50**	**0.532**

KITTI with calibration errors also has mis-projection caused by temporal and spatial occlusions in the original KITTI dataset.

Table 3 shows the results obtained on the KITTI dataset. The proposed method outperformed the baselines under all experimental conditions. Moreover, by comparing with those in Table 1 (the same dataset with accurate calibration), the proposed method outperformed Park's method [42] in terms of error rate and Cheng's method [28] in terms of MAE, although the proposed method was applied to data with calibration errors. The results indicate that the proposed method is robust to LiDAR-camera extrinsic calibration errors. Figure 8 shows the results and error maps. Figure 8 indicates that the proposed method successfully densified the depth of thin objects, e.g., poles, although the LiDAR points were not projected onto thin objects in the image.



(a) Dense depth map (b) *lines-16*

(c) *lines-32* (d) *lines-64*

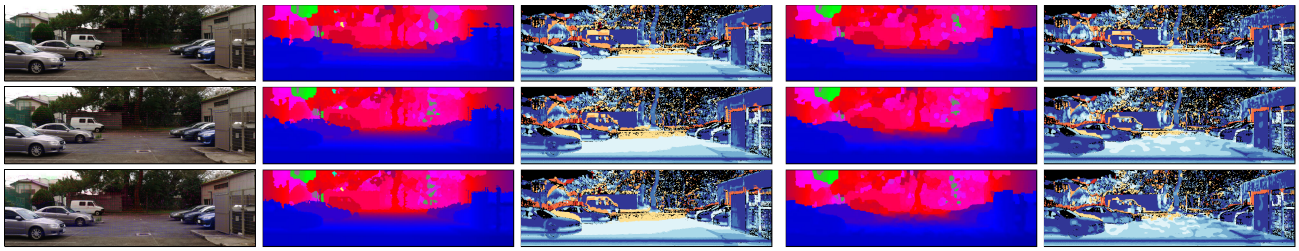
FIGURE 9. Mis-projection in the Komaba dataset. (a) There is no mis-projection in the ground truth. (b), (c), (d) Calibration error cause mis-projection (see the edges of vehicles).

2) KOMABA DATASET

The Komaba dataset was introduced in the literature [38] and has been used in a previous study [8]. The figure of an example frame of the Komaba dataset is in the supplementary material. This dataset includes five frames of data comprising motion stereo image pairs and dense depth maps captured by FARO FocusS 150. The motion between two scans is estimated by aligning the LiDAR point clouds. There is no spatial and temporal displacement between the camera and LiDAR, and occlusions are not expected in the Komaba dataset.

To create input sparse depth maps, we sampled the original dense depth maps and applied the randomly generated calibration errors (Table 2). Here, three sampling patterns were applied to simulate different LiDAR resolutions.

- *lines-16* sampled 16 scanlines.
- *lines-32* sampled 32 scanlines.



(a) Inputs (b) Results and error map of Yao et al. [8] (c) Results and error maps of ours

FIGURE 10. Inputs and results on Komaba dataset with calibration errors (top: *lines-16*, middle: *lines-32*, and bottom: *lines-64*). Refer Fig. 11 for the ground truth. (a) The input sparse depth maps projected onto the input image. (b), (c) Depth completion results and error maps. Around the boundary of the van in the rear left, the depth is not correctly estimated by Yao's method [8] and the proposed method with *lines-16*. With more scanlines (*lines-32* and *lines-64*), the proposed method estimated depth more accurately around the boundary.

TABLE 4. Depth completion results obtained on Komaba dataset with calibration errors.

Method	*lines-16*		*lines-32*		*lines-64*	
	iMAE [1/m]	MAE [m]	iMAE [1/m]	MAE [m]	iMAE [1/m]	MAE [m]
Kopf et al. [2]	6.80×10^{-3}	0.924	6.70×10^{-3}	0.921	6.63×10^{-3}	0.910
Ferstl et al. [3]	7.76×10^{-3}	1.001	7.68×10^{-3}	1.034	6.64×10^{-3}	1.013
Yao et al. [8]	7.16×10^{-3}	0.917	7.24×10^{-3}	0.938	7.23×10^{-3}	0.933
Ours (SSM only)	6.52×10^{-3}	0.866	6.17×10^{-3}	0.827	6.05×10^{-3}	0.816
Ours	$\mathbf{6.44 \times 10^{-3}}$	**0.857**	$\mathbf{6.09 \times 10^{-3}}$	**0.816**	$\mathbf{6.00 \times 10^{-3}}$	**0.809**

- *lines-64* sampled 64 scanlines. This condition is similar to the KITTI dataset, which has 64 scanlines.

The density and mis-projection in input data are visualized in Fig. 9.

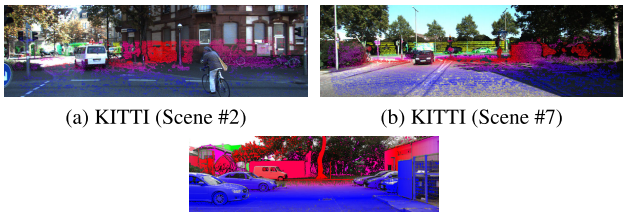
Table 4 shows the results obtained on the Komaba dataset. Here, rather than the error rate, we evaluated the inverse MAE (iMAE) because the Komaba dataset does not provide the ground truth of the disparity maps. The iMAE evaluates the accuracy of the inverse of the depth, which is proportional to the disparity. The proposed method outperformed the baselines under all experimental conditions. However, we observed a performance degradation with the proposed method as the number of scanlines decreased. This reduction in performance occurred because there was less possibility to find an appropriate value near the target pixel if scanlines are sparse. The relationship between the number of scanlines and performance is visually confirmed in Fig. 10.

C. PARAMETER STUDY

We evaluated the effect on MAE of the value of r using the KITTI dataset (Section IV-A), the KITTI dataset with the *blueprint* condition, and the Komaba dataset with the *lines-64* condition (Section IV-B). The results are shown in Table 5.

In Table 5, the value of r^* derived from Eq. (26) reside close to the r value to give the minimum MAE for every data. Here, we derived r^* for each data using Eq. (26) as follows.

- KITTI: $r^* = 6.68$ with $f = 959.7915$, $\theta_{\text{calib}} = 0$, and $\theta_{\text{scan}} = 0.4$.
- KITTI (*blueprint*): $r^* = 15.90$ with $f = 959.791$, $\theta_{\text{calib}} = 0.952$, and $\theta_{\text{scan}} = 0.4$.
- Komaba (*lines-64*): $r^* = 16.453$ with $f = 956.925$ and $\theta_{\text{calib}} = 1.096$. Note that, in this case, we ignored σ_{scan} because occlusion was not expected.



(a) KITTI (Scene #2) (b) KITTI (Scene #7)

(c) Komaba

FIGURE 11. Ground truth depth maps for Fig. 8 and 10.

TABLE 5. MAE variation by changing the radius in SSM.

Radius [pixels]	MAE [m]		
	KITTI	KTTI w/ error (*blueprint*)	Komaba w/ error (*lines-64*)
$r = 2$	0.374	1.575	0.949
$r = 5$	**0.356**	1.157	0.892
$r = 10$	0.442	0.565	0.814
$r = 15$	0.493	**0.528**	**0.809**
$r = 20$	0.530	0.573	0.860

Hence, the results supports our approach to set r in Section III-D.

V. CONCLUSION

We proposed a non-learning stereo-aided depth completion method that is robust to mis-projection and preserves LiDAR precision in the long range. Unlike previous methods, our method does not require accurate LiDAR-stereo extrinsic calibration parameters in any part of its process. Therefore, it is applicable in the conditions that the calibration is difficult to conduct. In the evaluations, our method demonstrated smaller

MAEs than previous state-of-the-art stereo-aided depth completion methods.

Our proposal is composed of SSM and the framework combining SSM and B-ADT aided smoothing. SSM searches for an optimal depth value for each pixel from its neighborly projected LiDAR points by an energy minimization approach, which can handle any type of mis-projection. In addition, we apply B-ADT-aided smoothing [8] to generate boundary-preseriving continuous depth maps since SSM is discrete optimization.

The current limitations of the proposed method include the accuracy dependency on the LiDAR scan density, as demonstrated by the evaluation discussed in Section IV-B.

We aim to extend our approach to run in real time for applying it to actual robotic systems. Since most of our processing time comes from LBP of SSM candidate selection (0.943 out of 0.999 [s]), we consider improving the selection process to be able to adapt faster optimizers.

REFERENCES

[1] J. Batlle, E. Mouaddib, and J. Salvi, "Recent progress in coded structured light as a technique to solve the correspondence problem: A survey," *Pattern Recognit.*, vol. 31, no. 7, pp. 963–982, 1998.

[2] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele, "Joint bilateral upsampling," *ACM Trans. Graph.*, vol. 26, no. 3, p. 96, Jul. 2007.

[3] D. Ferstl, C. Reinbacher, R. Ranftl, M. Rüether, and H. Bischof, "Image guided depth upsampling using anisotropic total generalized variation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 993–1000.

[4] J. Diebel and S. Thrun, "An application of Markov random fields to range sensing," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 291–298.

[5] N. Schneider, L. Schneider, P. Pinggera, U. Franke, M. Pollefeys, and C. Stiller, "Semantically guided depth upsampling," in *Proc. German Conf. Pattern Recognit.* Cham, Switzerland: Springer, 2016, pp. 37–48.

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

[7] A. Wong, X. Fei, S. Tsuei, and S. Soatto, "Unsupervised depth completion from visual inertial odometry," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1899–1906, Apr. 2020.

[8] Y. Yao, M. Roxas, R. Ishikawa, S. Ando, J. Shimamura, and T. Oishi, "Discontinuous and smooth depth completion with binary anisotropic diffusion tensor," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 5128–5135, Oct. 2020.

[9] L. Liu, X. Song, X. Lyu, J. Diao, M. Wang, Y. Liu, and L. Zhang, "FCFR-Net: Feature fusion based coarse-to-fine residual learning for depth completion," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 3, pp. 2136–2144.

[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. (ECCV)*, 2020, pp. 120–136.

[11] X. Cheng, P. Wang, C. Guan, and R. Yang, "CSPN++: Learning context and resource aware convolutional spatial propagation networks for depth completion," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 10615–10622.

[12] S. Zhao, M. Gong, H. Fu, and D. Tao, "Adaptive context-aware multi-modal network for depth completion," *IEEE Trans. Image Process.*, vol. 30, pp. 5264–5276, 2021.

[13] Y. Chen, B. Yang, M. Liang, and R. Urtasun, "Learning joint 2D-3D representations for depth completion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 10023–10032.

[14] J. Qiu, Z. Cui, Y. Zhang, X. Zhang, S. Liu, B. Zeng, and M. Pollefeys, "DeepLiDAR: Deep surface normal guided depth prediction for outdoor scene from sparse LiDAR data and single color image," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3313–3322.

[15] A. Li, Z. Yuan, Y. Ling, W. Chi, S. Zhang, and C. Zhang, "A multi-scale guided cascade hourglass network for depth completion," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 32–40.

[16] W. Van Gansbeke, D. Neven, B. De Brabandere, and L. Van Gool, "Sparse and noisy LiDAR completion with RGB guidance and uncertainty," in *Proc. 16th Int. Conf. Mach. Vis. Appl. (MVA)*, May 2019, pp. 1–6.

[17] 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. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2811–2820.

[18] L. Yan, K. Liu, and E. Belyaev, "Revisiting sparsity invariant convolution: A network for image guided depth completion," *IEEE Access*, vol. 8, pp. 126323–126332, 2020.

[19] 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. 2020.

[20] R. Schuster, O. Wasenmuller, C. Unger, and D. Stricker, "SSGP: Sparse spatial guided propagation for robust and generic interpolation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 197–206.

[21] L. Bai, Y. Zhao, M. Elhousni, and X. Huang, "DepthNet: Real-time LiDAR point cloud depth completion for autonomous vehicles," *IEEE Access*, vol. 8, pp. 227825–227833, 2020.

[22] S. S. Shivakumar, T. Nguyen, I. D. Miller, S. W. Chen, V. Kumar, and C. J. Taylor, "DFuseNet: Deep fusion of RGB and sparse depth information for image guided dense depth completion," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 13–20.

[23] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, 2013.

[24] A. Geiger, F. Moosmann, O. Car, and B. Schuster, "Automatic camera and range sensor calibration using a single shot," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2012, pp. 3936–3943.

[25] G. Pandey, J. McBride, S. Savarese, and R. Eustice, "Automatic targetless extrinsic calibration of a 3D LiDAR and camera by maximizing mutual information," in *Proc. AAAI Conf. Artif. Intell.*, 2012, vol. 26, no. 1, pp. 1–7.

[26] R. Ishikawa, T. Oishi, and K. Ikeuchi, "LiDAR and camera calibration using motions estimated by sensor fusion odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 7342–7349.

[27] V. John, Q. Long, Z. Liu, and S. Mita, "Automatic calibration and registration of LiDAR and stereo camera without calibration objects," in *Proc. IEEE Int. Conf. Veh. Electron. Saf. (ICVES)*, Nov. 2015, pp. 231–237.

[28] X. Cheng, Y. Zhong, Y. Dai, P. Ji, and H. Li, "Noise-aware unsupervised deep LiDAR-stereo fusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6339–6348.

[29] L. Zhang and S. M. Seitz, "Parameter estimation for MRF stereo," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2005, pp. 288–295.

[30] D. Hernandez-Juarez, A. Chacón, A. Espinosa, D. Vázquez, J. C. Moure, and A. M. López, "Embedded real-time stereo estimation via semi-global matching on the GPU," *Proc. Comput. Sci.*, vol. 80, pp. 143–153, Dec. 2016.

[31] K. Yamaguchi, D. McAllester, and R. Urtasun, "Efficient joint segmentation, occlusion labeling, stereo and flow estimation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 756–771.

[32] M. P. Muresan, M. Negru, and S. Nedevschi, "Improving local stereo algorithms using binary shifted windows, fusion and smoothness constraint," in *Proc. IEEE Int. Conf. Intell. Comput. Commun. Process. (ICCP)*, Sep. 2015, pp. 179–185.

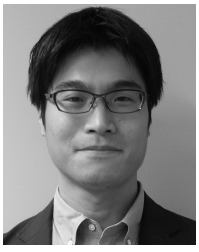
[33] R. Spangenberg, T. Langner, S. Adfeldt, and R. Rojas, "Large scale semi-global matching on the CPU," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2014, pp. 195–201.

[34] A. Zureiki, M. Devy, and R. Chatila, "Stereo matching using reduced-graph cuts," in *Proc. IEEE Int. Conf. Image Process.*, vol. 1, Sep. 2007, p. I-237.

[35] H. Wang, R. Fan, P. Cai, and M. Liu, "PVStereo: Pyramid voting module for end-to-end self-supervised stereo matching," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4353–4360, Jul. 2021.

[36] X. Cheng, Y. Zhong, M. Harandi, Y. Dai, X. Chang, H. Li, T. Drummond, and Z. Ge, "Hierarchical neural architecture search for deep stereo matching," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 22158–22169.

[37] L. Chen, Y. He, J. Chen, Q. Li, and Q. Zou, "Transforming a 3-D LiDAR point cloud into a 2-D dense depth map through a parameter self-adaptive framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 1, pp. 165–176, Jan. 2017.
[38] A. Hirata, R. Ishikawa, M. Roxas, and T. Oishi, "Real-time dense depth estimation using semantically-guided LiDAR data propagation and motion stereo," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3806–3811, Oct. 2019.
[39] Y. Yang, A. Wong, and S. Soatto, "Dense depth posterior (DDP) from single image and sparse range," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3353–3362.
[40] H. Badino, D. Huber, and T. Kanade, "Integrating LiDAR into stereo for fast and improved disparity computation," in *Proc. Int. Conf. 3D Imag., Modeling, Process., Vis. Transmiss.*, May 2011, pp. 405–412.
[41] W. Maddern and P. Newman, "Real-time probabilistic fusion of sparse 3D LiDAR and dense stereo," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 2181–2188.
[42] K. Park, S. Kim, and K. Sohn, "High-precision depth estimation with the 3D LiDAR and stereo fusion," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2018, pp. 2156–2163.
[43] J. Choe, K. Joo, T. Imtiaz, and I. S. Kweon, "Volumetric propagation network: Stereo-LiDAR fusion for long-range depth estimation," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4672–4679, Jul. 2021.
[44] K. Park, S. Kim, and K. Sohn, "High-precision depth estimation using uncalibrated LiDAR and stereo fusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 321–335, Jan. 2020.
[45] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2320–2327.
[46] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Generalized belief propagation," in *Proc. NIPS*, vol. 13, 2000, pp. 689–695.
[47] M. Menze, C. Heipke, and A. Geiger, "Object scene flow," *ISPRS J. Photogram. Remote Sens.*, vol. 140, pp. 60–76, Jun. 2018.
[48] M. A. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
[49] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *J. Math. Imag. Vis.*, vol. 40, no. 1, pp. 120–145, 2011.
[50] R. B. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Shanghai, China, May 2011, pp. 1–4.
[51] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger, "Sparsity invariant CNNs," in *Proc. Int. Conf. 3D Vis. (DV)*, Oct. 2017, pp. 11–20.



YASUHIRO YAO received the B.S. and M.E. degrees from The University of Tokyo, Japan, in 2007 and 2010, respectively, where he is currently pursuing the Ph.D. degree in information studies. In 2010, he joined NTT as a Researcher. From 2013 to 2016, he was a Cloud Solution Architect at Dimension Data APAC, Singapore. He is currently a Senior Research Engineer at NTT Human Informatics Laboratories. His research interests include computer vision and sensor fusion.

RYOICHI ISHIKAWA received the B.E. degree from the Department of Electrical Engineering, The University of Tokyo, Japan, in 2014, and the M.E. and Ph.D. degrees in electrical engineering and information systems from The University of Tokyo, in 2016 and 2019, respectively. Since 2019, he has been a Project Researcher at the Institute of Industrial Science, The University of Tokyo. His research interests include robot vision, sensor fusion, and calibration.

SHINGO ANDO received the B.E. degree in electrical engineering and the Ph.D. degree in engineering from Keio University, Kanagawa, in 1998 and 2003, respectively. In 2003, he joined NTT. He has been engaged in research and practical application development in the fields of image processing, pattern recognition, and digital watermarks. He is a member of IEICE and the Institute of Image Information and Television Engineers.

KANA KURATA received the B.S. degree in earth and planetary sciences and the M.E. degree in environmental studies from Nagoya University, in 2016 and 2018, respectively. In 2018, she joined NTT. Her research interests include computer vision and pattern recognition.

NAOKI ITO received the M.E. degree from the Toyohashi University of Technology, Aichi, in 2001. In 2001, he joined NTT and engaged in research on character recognition. In 2004, he moved to NTT EAST and engaged in the development of security systems. In 2008, he moved to NTT Cyber Space Laboratories. He has been working on the development of real-world digitalization technologies. He is currently a Senior Research Engineer at NTT Human Informatics Laboratories.

JUN SHIMAMURA received the B.E. degree in engineering science from Osaka University, in 1998, and the M.E. and Ph.D. degrees from the Nara Institute of Science and Technology, in 2000 and 2006, respectively. In 2000, he joined NTT Cyber Space Laboratories. He is currently a Senior Research Engineer and a Supervisor of scene analysis technology at NTT Human Informatics Laboratories, Japan. His research interests include computer vision and mixed reality.

TAKESHI OISHI (Member, IEEE) received the B.Eng. degree in electrical engineering from Keio University, in 1999, and the Ph.D. degree in interdisciplinary information studies from The University of Tokyo, in 2005. He is currently an Associate Professor at the Institute of Industrial Science, The University of Tokyo. His research interests include 3D modeling from reality, digital archiving of cultural heritage assets, and mixed/augmented reality.