LSMD-Net: LiDAR-Stereo Fusion with Mixture Density Network for Depth Sensing

Hanxi Yin[1], Lei Deng[2], Zhixiang Chen[3], Baohua Chen[4], Ting Sun[2], Yuseng Xie[2], Junwei Xiao[1], Yeyu Fu[2], Shuixin Deng[2], and Xiu Li[1]*

[1] Shenzhen International Graduate School, Tsinghua University, Shenzhen, China
{yhx20,xjw20}@mails.tsinghua.edu.cn,li.xiu@sz.tsinghua.edu.cn
[2] School of Instrument Science and Opto-Electronics Engineering, Beijing Information Science and Technology University, Beijing, China
[3] Department of Computer Science, The University of Sheffield, UK
[4] Department of Automation, Tsinghua University, Beijing, China

Abstract. Depth sensing is critical to many computer vision applications but remains challenge to generate accurate dense information with single type sensor. The stereo camera sensor can provide dense depth prediction but underperforms in texture-less, repetitive and occlusion areas while the LiDAR sensor can generate accurate measurements but results in sparse map. In this paper, we advocate to fuse LiDAR and stereo camera for accurate dense depth sensing. We consider the fusion of multiple sensors as a multimodal prediction problem. We propose a novel end-to-end learning framework, dubbed as LSMD-Net to faithfully generate dense depth. The proposed method has dual-branch disparity predictor and predicts a bimodal Laplacian distribution over disparity at each pixel. This distribution has two modes which captures the information from two branches. Predictions from the branch with higher confidence is selected as the final disparity result at each specific pixel. Our fusion method can be applied for different type of LiDARs. Besides the existing dataset captured by conventional spinning LiDAR, we build a multiple sensor system with a non-repeating scanning LiDAR and a stereo camera and construct a depth prediction dataset with this system. Evaluations on both KITTI datasets and our home-made dataset demonstrate the superiority of our proposed method in terms of accuracy and computation time.

1 Introduction

Real-time, dense and accurate depth sensing is crucial for many computer vision applications, including SLAM, autonomous driving and augmented realities. There are two kinds of sensors, active and passive sensors used to sense depth. However, either active sensors like LiDAR scanner or passive sensors like stereo camera have their limitations. On the one hand, stereo camera can provide dense

[*] Corresponding author

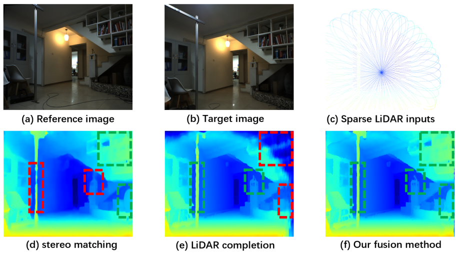


Fig. 1. Illustration of predicted disparity maps (d-f) based on inputs (a-c). Green/red dotted rectangles show the areas where the depth prediction is accurate/inaccurate. The proposed method (f) can take full advantages of different sensors.

depth estimation but underperforms in texture-less or repetitive areas, occlusion areas, thin structure and poor light conditions. On the other hand, LiDAR scanner often provides precise but relatively sparse depth measurements. These limitations hinder their usages in practical applications. One possible solution to remedy this issue is to combine them by multiple sensor fusion. In terms of fusing LiDAR with RGB camera, there are existing works proposing to fuse LiDAR and monocular camera [20, 36, 16]. However, the monocular camera setting makes it depend on strong scene priors and is vulnerable to overfitting as monocular depth estimation is inherently unreliable and ambiguous. On the contrary, in this paper, we consider LiDAR-stereo fusion. The stereo camera is more robust as it computes the geometric correspondence between an image pair. The fused depth also benefits from the robustness.

With a stereo camera and a LiDAR sensor, there are two possible ways to generate dense depth prediction: (1) stereo matching from a pair of stereo images, (2) LiDAR completion from sparse LiDAR measurements and a RGB image. The former estimates disparities between image pairs by matching pixels and recovers depth through triangulation, while the latter utilizes a corresponding RGB image to guide the depth interpolation. These two methods exploit information from different modalities with different priori hypotheses and characteristics. The performance of stereo camera depends on image matching, while that of LiDAR completion is limited by the density and quality of point clouds. As illustrated in Fig. 1, stereo matching works well in rich textured areas, but has difficulties in dealing with fine structure and texture-less areas. LiDAR completion performs depth interpolation accurately. However, it has poor extrapolation ability in areas where point clouds are too sparse or missing. Besides, the quality of LiDAR point clouds are poor in reflective surface and distant areas. Based on this analysis, these two methods are expected to complement each other from the perspective of multimodal fusion.

Existing LiDAR-stereo fusion works either use LiDAR information to assist stereo matching [37, 29] or simply combine them at the output stage [38, 28]. The former one simply injects LiDAR information into cost volume which is the core component of stereo matching. It is confined to the stereo matching architecture and therefore cannot avoid the inherent drawbacks of image matching. The latter one lacks deep feature fusion, which makes it not fully utilize the intrinsic information of different sources. To take full advantage of the unique characteristics of different sources, we propose a confidence based fusion method by combining stereo matching and LiDAR completion branches. The features of reference image is fed into both branches to fuse with features from different sensor. This breaks the symmetry of stereo image pair and thus gets rid of the limitation of stereo matching pipeline. Besides, the confidence-based fusion can better solve the redundancy and contradiction between heterogeneous sources. Furthermore, our model is built over disparity which is inversely proportional to depth. Inverse depth allows probability distribution to describe depth from nearby to infinity and is more stable to regress with a finite boundary. Specifically, we formulate the task of LiDAR-Stereo fusion as a multimodal prediction problem. Instead of regress disparity directly, we exploit a mixture density network to estimate a bimodal probability distribution over possible disparities for each pixel. Predictions from the branch with higher confidence is selected as the final disparity result at each specific pixel.

To further evaluate our method, we have constructed a dataset based on a solid state Livox LiDAR. Compared with conventional spinning LiDAR, Solid state LiDAR is more suitable for our LiDAR-stereo fusion task in various scenarios for large FOV overlap with RGB camera, advantages in terms of point cloud density and affordable cost.

In summary, the contributions of this paper are summarized as follows:

(I) We propose a novel end-to-end learning dual-branch framework called LSMD-Net (**L**iDAR-**S**tereo fusion with **M**ixture **D**ensity Network) to fuse LiDAR and stereo camera for accurate and dense depth estimation in real time.

(II) We treat multisensor fusion as a multimodal prediction problem. A bimodal distribution is utilized to capture information from different modes and provides a measure of confidence for them at each pixel, which can take full advantage of different sensors for better depth prediction.

(III) We build a data collecting system equipped with a solid state LiDAR and a stereo camera and present a depth prediction dataset.

2 Related Works

Stereo Matching. With the development of convolutional neural networks (CNNs), learning-based stereo matching methods have achieved great success. An end-to-end stereo matching architecture has four trainable components: (a) feature extraction, (b) cost volume, (c) aggregation and (d) regression. Most methods can be categorized into 2D architectures and 3D architectures according to the type of cost volume. The first class [24, 27] performs correlation layer to build 2D cost volume and uses 2D CNNs aggregation, which is less accurate than the second. The second class constructs 3D cost volume by concatenating image features [19, 4] or using group-wise correlation [13]. Although more accurate, the second class suffers from computational complexities. Stereo matching

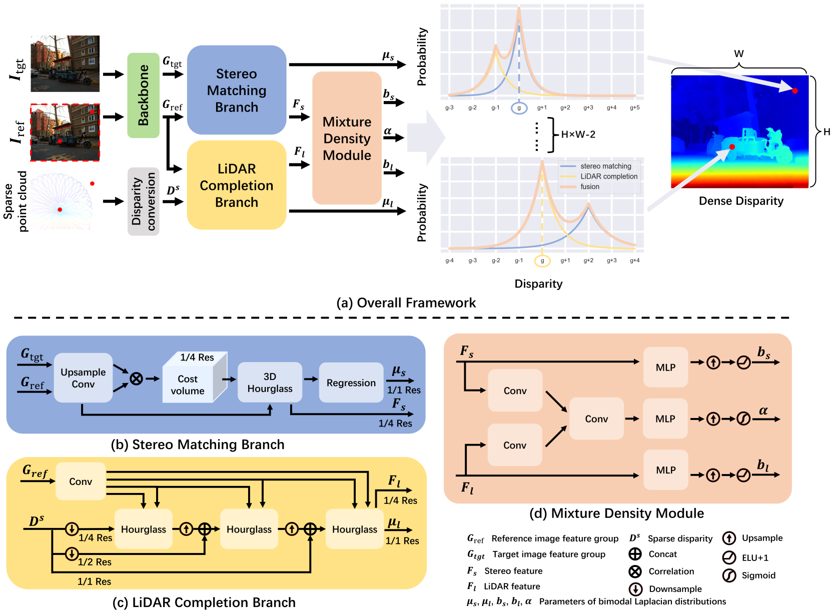


Fig. 2. An overview of our method. Our model has two branches (blue and yellow) which extract features from different sensors. Features from different branches are fused in mixture density module. For each pixel in reference image, a bimodal Laplacian distribution (light orange curve) is predicted. At inference time, we use the expectation of the more confident branch as the final disparity for each specific pixel (as shown by the dotted line).

performs bad in texture-less or repetitive areas. Fusion with LiDAR is therefore important for obtaining reliable depth estimation.

LiDAR-RGB Fusion. LiDAR-camera fusion is well known for its practicability in 3D perception. There are two types of fusion: LiDAR-monocular and LiDAR-stereo fusion. The former one, also known as depth completion, regresses dense depth from sparse depth cues with the help of monocular image information [20, 36, 16]. Relying on priors of a particular scene, LiDAR-monocular fusion is inherently unreliable and ambiguous [37]. LiDAR-stereo fusion is less ambiguous in terms of the absolute distance for stereo matching relies on the geometric correspondence across images. Several works [26, 1, 9, 22, 34] studied the application of LiDAR-stereo fusion in robotic for the past two decades. Park et al. [28] was the first to implement CNNs in context of LiDAR and stereo fusion. Learning-based methods can be roughly divided into feature level fusion and decision level fusion. Feature level fusion [37, 29, 5, 41] encodes LiDAR information at early stage in stereo matching while decision level fusion [38, 28] directly fuses hypotheses generated by different sensors. Our LiDAR-stereo fusion method takes advantage of both feature level fusion and confidence-based decision level fusion.

Multimodal Predictions with CNNs. Standard depth prediction works directly regress a scalar depth at every pixels. However, LiDAR-stereo fusion system is complex for there are multimodal inputs and we can obtain more than one candidate outputs from them. A lot of works have been done for multiple solutions from CNNs. Guzman-Rivera et al. [14] introduced the Winner-Takes-All (WTA) loss for classification tasks while another option is Mixture Density Networks (MDNs) by C. M. Bishop [3]. Instead of using a parametric distribution, MDNs learn parameterization as a part of the neural network. O. Makansi et al. [23] used MDNs for Multimodal Future Prediction. G. Hager [17] proposed a MDNs-based approach to estimate uncertainty in stereo disparity prediction networks. F. Tosi [35] uses a bimodal approach to solve the over-smoothing issue in stereo matching, which inspired us greatly. In contrast to them, we apply bimodal distributions to capture information from two sensors, which can solve the redundancy and contradiction between heterogeneous sources.

3 LSMD-Net

As shown in Fig. 2(a), our proposed model aims to generate an accurate dense disparity map $\boldsymbol{D}^d \in \mathbb{R}^{H \times W}$ given sparse LiDAR measurements $\boldsymbol{S} \in \mathbb{R}^{n \times 3}$ $(n < H \times W)$ and a pair of stereo images $\boldsymbol{I}_{ref}$, $\boldsymbol{I}_{tgt} \in \mathbb{R}^{H \times W \times 3}$. A dense depth map can be further obtained by $\boldsymbol{D}^d$. There are two stages in our model pipeline. At the first stage (Sec. 3.1), we separately estimate dense disparity maps and confidence related feature maps for images and LiDAR measurements. The image based disparity estimation is obtained by stereo matching. The LiDAR based disparity estimation is completed by LiDAR completion with reference image. At the second stage (Sec. 3.2), we employ a mixture density module to fuse these estimations into a final dense depth map. Specifically, we introduce a confidence-based fusion to effectively exploit the information from different sensors.

3.1 Dual-Branch Disparity Predictor

To estimate dense disparity and features from images and LiDAR measurements, we employ separated disparity prediction branches of stereo matching and LiDAR completion. Before passing through the individual branches, we firstly preprocess the images and LiDAR signals. The images are processed by a backbone network to extract meaningful features. Specifically, we adopt the MobileNetV2 model [31] pre-trained on ImageNet [8] to extract image features at scales of 1/2, 1/4, 1/8, 1/16 and 1/32 of the input image resolution. Both reference and target images are passed through the same backbone with shared weight to obtain the corresponding feature groups $\boldsymbol{G}_{ref}$, $\boldsymbol{G}_{tgt}$. Inspired by the parametrization for monocular SLAM [7], we project sparse LiDAR points $\boldsymbol{S}$ onto the image plane of $\boldsymbol{I}_{ref}$ and convert the projected depth map to a disparity map $\boldsymbol{D}^s$. Note that the disparity is inversely proportional to the depth. Our network directly predicts disparity rather than depth. This conversion from depth to disparity has two advantages. First, it allows us to consider a wider depth range in our model. Second, the model prediction is more stable as the regression target is with finite boundary.

Stereo Matching Branch This branch takes as input the feature groups of reference and target images, $\boldsymbol{G}_{ref}$ and $\boldsymbol{G}_{tgt}$ to generate a disparity map $\boldsymbol{\mu}_s \in \mathbb{R}^{H \times W}$ and a feature $\boldsymbol{F}_s \in \mathbb{R}^{H/4 \times W/4 \times D/4}$, where D is the maximum disparity value. D is set to 192 in our network. This feature contains matching probabilities along possible disparities and is used in the fusion stage. Denoting this stereo matching branch as ϕ with parameters θ, we can formally write it as,

$$\{\boldsymbol{F}_s, \boldsymbol{\mu}_s\} = \phi_\theta(\boldsymbol{G}_{ref}, \boldsymbol{G}_{tgt}). \tag{1}$$

Our design of this stereo matching branch follows the mainstream learning-based stereo matching framework. It consists of matching cost computation, cost aggregation, and disparity regression. With feature groups $\boldsymbol{G}_{ref}$, $\boldsymbol{G}_{tgt}$ as input, a U-Net [30] style upsampling module with long skip connections at each scale level is built to propagate context information to higher resolution layers. Image features with less than 1/4 of the input image resolution in $\boldsymbol{G}_{ref}$, $\boldsymbol{G}_{tgt}$ are upsampled by this module to a quarter of the input image resolution. The cost volume is then built by computing the correlation between the outputs of the upsample module. We keep the size of cost volume at $H/4 \times W/4 \times D/4$ to reduce cost aggregation computing costs. As for cost aggregation, instead of employing neighborhood aggregation, we first capture geometric features from cost volume by 3D convolutions, and then utilize the guidance weights generated from image features to redistribute this geometric information to local features as in CoEx [2]. The output feature $\boldsymbol{F}_s$ is from the aggregated cost volume. To reduce the computation time of regression, our model regresses disparity at 1/4 of the input image resolution from cost volume and finally upsamples it to the original input image resolution.

LiDAR Completion Branch This branch takes as input the feature group of reference image $\boldsymbol{G}_{ref}$ and a sparse disparity map $\boldsymbol{D}^s$ to generate a dense disparity map $\boldsymbol{\mu}_l \in \mathbb{R}^{H \times W}$ and a feature map $\boldsymbol{F}_l \in \mathbb{R}^{H/4 \times W/4 \times C}$, where C is set to 64. This feature contains information extracted from LiDAR measurement and reference image and is used in the fusion stage. Denoting this disparity completion branch as ψ with parameters ω, we can formally write it as,

$$\{\boldsymbol{F}_l, \boldsymbol{\mu}_l\} = \psi_\omega(\boldsymbol{G}_{ref}, \boldsymbol{D}^s). \tag{2}$$

This disparity completion can be considered as a disparity map interpolation guided by the reference image feature. Similar to MSG-CHN [20], we use coarse to fine cascade hourglass CNNs to interpolate the disparity features at three levels. The output of the coarse level is upsampled and concatenated with the sparse disparity map at corresponding scale as the input of the fine level. At each level, the hourglass CNNs refine the disparity features according to both the input disparity features and the corresponding scale image features from $\boldsymbol{G}_{ref}$. The disparity features and the image features are fused by concatenation. We expect that this design can exploit the clues from reference image to guide the interpolation of disparities for LiDAR. The output feature $\boldsymbol{F}_l$ is extracted from the last hourglass CNNs. It contains LiDAR and image information and is with the same resolution as $\boldsymbol{F}_s$.

3.2 Mixture Density Module

The mixture density module is used to fuse the disparity information from two branches. We view the estimated disparity at each pixel as a probability distribution over the possible range of disparities. And the fusion of disparity estimations leads to the final probability distribution, from which we can get the output disparity. To be specific, we utilize the Laplacian distribution to model the probability distribution for each branch as

$$\boldsymbol{P}(\boldsymbol{d}) = \frac{1}{2\boldsymbol{b}} e^{\left(-\frac{|\boldsymbol{\mu}-\boldsymbol{d}|}{\boldsymbol{b}}\right)}, \tag{3}$$

where $\boldsymbol{\mu}$ is the location parameter map and $\boldsymbol{b}$ is the scale parameter map. We opt to the Laplacian distribution rather than the widely used Gaussian distribution. This is because the Gaussian assumption is sensitive to outliers but the Laplacian distribution is more robust as it has a heavier tails than Gaussian.

We take the estimated disparities from each branch, $\boldsymbol{\mu}_s$ and $\boldsymbol{\mu}_l$ as the location parameters. Rather than setting a global scale parameter for each branch, we propose to learn pixel wise parameters $\boldsymbol{b}$ from the features of each branch $\boldsymbol{F}_s$ and $\boldsymbol{F}_l$ with two independent networks, as shown in Fig. 2(d).

$$\boldsymbol{b}_s = \sigma(MLP(\boldsymbol{F}_s)),\ \boldsymbol{b}_l = \sigma(MLP(\boldsymbol{F}_l)), \tag{4}$$

where $\sigma(\cdot)$ is the activation function. An exponential activation function is adopted in traditional mixture density network to predict parameters with positive value. However, the exponential increases to a very large value in case of high variance, which makes the training unstable. In this work, following [12], we choose ELU as the activation function. The ELU activation function shares the same exponential behavior for small activation value but is linear to the input for large activation value.

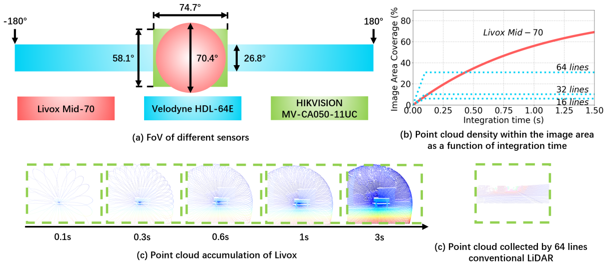
$$f(\beta, x) = ELU(\beta, x) + 1 = \begin{cases} \beta(e^x - 1) + 1, & x < 0 \\ x + 1, & x \geq 0 \end{cases}, \tag{5}$$

where β is a parameter to control the slope and is set to 1 in our work.

With the Laplacian distributions of two branches, $\boldsymbol{P}_s$ and $\boldsymbol{P}_l$ at hand, we compute the final probability distribution as a weighted sum of these two distributions.

$$\boldsymbol{P}_m(\boldsymbol{d}) = \boldsymbol{\alpha}\boldsymbol{P}_s(\boldsymbol{d}) + (1-\boldsymbol{\alpha})\boldsymbol{P}_l(\boldsymbol{d}) = \frac{\boldsymbol{\alpha}}{2\boldsymbol{b}_s} e^{-\frac{|\boldsymbol{\mu}_s-\boldsymbol{d}|}{\boldsymbol{b}_s}} + \frac{1-\boldsymbol{\alpha}}{2\boldsymbol{b}_l} e^{-\frac{|\boldsymbol{\mu}_l-\boldsymbol{d}|}{\boldsymbol{b}_l}} \tag{6}$$

where $\boldsymbol{\alpha}$ is a parameter map weighting the contributions of different branches. We also design a network to learn $\boldsymbol{\alpha}$ from the branch features $\boldsymbol{F}_s$ and $\boldsymbol{F}_l$ (Fig. 2(d)). Convolutional layers are applied to process and aggregate the branch features followed by MLP and a sigmoid activation function. We can use the fused distribution in Eq. 6 to compute the loss at training stage. However, at inference stage, we aims to predict a single disparity value for each pixel. One possible solution is to use the conditional expectation as the final output. In our case,



(a) FoV of different sensors

(b) Point cloud density within the image area as a function of integration time

(c) Point cloud accumulation of Livox

(c) Point cloud collected by 64 lines conventional LiDAR

Fig. 3. Illustration of characteristics of two LiDARs. (a) shows that solid state LiDAR (red) fits better with camera (green) than conventional LiDAR (blue) in terms of FoV. (b) and (c) illustrates point cloud accumulation of solid state Livox LiDAR quantitatively and qualitatively.

one branch may be more confident than the other branch for particular pixels. Simply calculating the conditional expectation will deteriorate the performance as the outliers from the less confident branch are considered. To this end, we propose to use the expectation of the more confident branch as the final disparity prediction $\hat{\boldsymbol{d}}$. And the branch confidence is determined by $\boldsymbol{\alpha}$.

$$\hat{\boldsymbol{d}} = \begin{cases} \boldsymbol{\mu}_s, \ \boldsymbol{\alpha} \geq 0.5 \\ \boldsymbol{\mu}_l, \ \boldsymbol{\alpha} < 0.5 \end{cases} . \tag{7}$$

3.3 Losses

Our model is trained with the supervision on the final output and the intermediate supervisions over two branches. The training objective is to minimize the overall loss

$$\mathcal{L} = \omega_m \mathcal{L}_m + \omega_s \mathcal{L}_s + \omega_l \mathcal{L}_l, \tag{8}$$

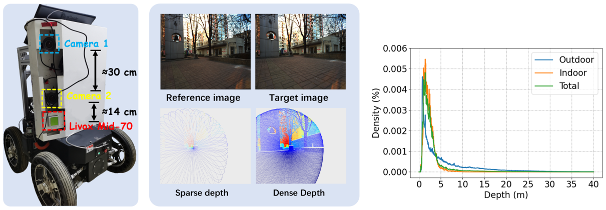
where ω_m, ω_s and ω_l are the weighting parameters for the three losses $\mathcal{L}_m$ $\mathcal{L}_s$ $\mathcal{L}_l$. $\mathcal{L}_m$ is the loss over the final output of the fusion method. $\mathcal{L}_s$ and $\mathcal{L}_l$ are the losses over the outputs of stereo matching branch and LiDAR completion branch, respecitvely. We compute the negative logarithm of the likelihood loss based on the PDFs (P_m, P_s, P_l) in Eq. 6 for each loss which can be expressed as:

$$\mathcal{L}_{NLL}(\theta) = -\mathbb{E}_{d,x,I} \log P(d|x, I_{ref}, I_{tgt}, D^s, \theta), \tag{9}$$

where d is the ground truth disparity at each pixel location x in reference image I_{ref} from the dataset. θ denotes parameters of our model.

4 Livox-Stereo Dataset

There are two widely used kinds of LiDAR sensors: mechanical spinning LiDAR and solid state LiDAR. The mechanical spinning LiDAR uses mechanical



(a) Data collecting system (b) A set of data (c) Depth distribution

Fig. 4. Illustration of (a) data collecting system hardware, (b) a set of data in dataset and (c) the depth distribution of dataset.

rotation to spin the sensor for 360 degree detection. The density of collected point cloud is determined by the number of scanner layer. Most existing depth sensing datasets [36, 40] use this kind of sensor to acquire LiDAR information. The solid state LiDAR such as Livox uses prism scanning to acquire depth information in a non-repeating scanning pattern and accumulates point clouds to generate relative dense depth maps [21, 42]. Livox LiDAR is promising for LiDAR-stereo fusion task for three reasons. Firstly, as shown in Fig. 3(a),it fits well with cameras because of their large overlapping FoV. Livox collects point cloud which covers a larger image area, rather than focusing on a limited number of scan lines with a narrow FoV like traditional LiDAR, which is illustrated in Fig. 3(c)(d). Secondly, compared to traditional LiDAR, Livox has advantages in terms of point cloud density within the image area when the accumulation time is sufficient as Fig. 3(b) shows. Thirdly, Livox is promising in various scenarios owing to its portability and advantages in terms of cost. To show the feasibility of our proposed depth fusion method on different LiDAR sensors, we further present a Livox-stereo dataset collected by our own system for evaluation.

4.1 Data Collecting System

As shown in Fig. 4(a), our system hardware includes two HIKVISION MV-CA050-11UC color cameras stacked vertically and a Livox Mid-70 LiDAR. The distance between the two cameras is 30 centimeters, which results in errors in four centimeters for depth within three meters (see Suppl. for the error estimation). The Livox is placed close to the reference camera (the camera below) to increase the overlap between their FoV.

The calibration process of our system can be divided into two steps. First, We follow the binocular calibration process in OpencV [18] to compute the intrinsic and extrinsic parameters of the stereo cameras. After that, We use the calibrated reference camera to calculate the extrinsic parameters of Livox. Specifically, the Livox extrinsic parameters are derived by PnP and RANSAC after the extraction of corresponding key points between the depth map generated by Livox and the remapped image from the reference camera [39].

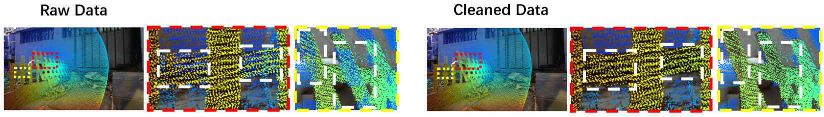


Fig. 5. Illustration of noise filtering in raw point cloud. Background points (blue) on the foreground surface are noise. Most noise have been removed successfully. We zoom-in the areas of interest in dotted area.

4.2 Livox-Stereo Dataset

We collected 507 sets data in both indoor and outdoor scenes in residential areas. Each set of data consists of a pair of stereo images at the resolution of 1224×1024 and a pair of sparse and dense depth maps. We show a set of data in Fig. 4(b). More examples are in the Suppl. The sparse and dense depth maps are collected by the same Livox sensor on the same scene but with different accumulation time. Specifically, we obtain a sparse depth map with the coverage around 10% by setting the accumulation time to 0.3 seconds. The dense depth map is obtained by accumulating the point clouds for 3 seconds to achieve a coverage around 60%. The dense depth maps are used as ground truth to train our fusion model. We split the dataset into train, validation and test subsets at a ratio around 7:1:2. The specific statistics of the subsets is in Suppl.

We present the statistical information of pixel level depth in Fig. 4(b). From the curves in the figure, we can find that the majority pixels are with small depth values. Specifically, the depths of 53.07% pixels fall within 3 meters (80.55% and 44.27% for indoor and outdoor) and 68.75% falls within five meters. The depth distribution matches well with the depth range that the stereo cameras can generate depth with low errors.

During data collection, we observed that the difference of projection centers of Livox and stereo cameras leads to noise when point clouds are projected onto the image plane. In order to remove this noise, we identify inconsistent LiDAR points by applying semi-global matching (SGM) [15] and refuse these points. This point cloud filtering is based on the assumption that passive and activate sensors rarely make the same inaccurate prediction in these problematic areas. We show some examples of this filtering in Fig. 5. Emperically, we found that this works well to produce clean point clouds for our task.

In Table 1, we compare our dataset with other relevant datasets. Our dataset is unique in several senses. First, the point clouds in our dataset are collected by Livox LiDAR. Non-repeating scanning Livox LiDAR allows our dataset to provide sparse LiDAR inputs and to be better than other LiDAR-based datasets in term of the density of depth information for supervision. Besides, our dataset includes both indoor and outdoor scenes which is different from the autonomous driving scene in existing datasets. This is promising to improve the generalization the performance of LiDAR-RGB fusion model in practical application.

Table 1. Comparison between our dataset and other published depth sensing datasets.

Datasets	Tools	Real	Sparse LiDAR	SceneType	DataSize		Coverage
					Train	Test	
Middlebury [32]	structured light scanner	✓		indoor	15	15	≈ 96%
ETH3D [33]	structured light scanner	✓		indoor/outdoor	27	20	≈ 69%
KITTI stereo [25, 10]	Velodyne HDL-64E	✓		autonomous driving	394	395	≈ 19%
KITTI depth completion [36]	Velodyne HDL-64E	✓	✓	autonomous driving	43k	1k	16.1%
FlyingThings3D [24]	software			animation	22k	42k	100%
DrivingStereo [40]	Velodyne HDL-64E S3	✓		autonomous driving	174k	8k	≈ 4%
Ours	Livox Mid-70	✓	✓	indoor/outdoor	407	100	≈ 60%

5 Experiments

In this section, we demonstrate the effectiveness of our proposed depth fusion method on three datasets, KITTI stereo dataset [36], KITTI depth completion dataset [25] and our new Livox-stereo dataset.

5.1 Datasets and Evaluation Metrics

KITTI Stereo 2015 and KITTI Depth Completion are real-world datasets with street views from a driving car. We follow [37, 9, 22, 11] to evaluated our model on the training set in Stereo 2015 dataset and the validation set in Depth Completion dataset (see Suppl. for more details).

For Stereo 2015 dataset, we report several common metrics in stereo matching tasks: end-point error (EPE, the mean average disparity error in pixels) and the percentage of disparity error that is greater than 1, 2 and 3 pixel(s) away from the ground truth ($> 1px$, $> 2px$ and $> 3px$). For Depth Completion dataset, root mean squared error of depth ($RMSE$, m), mean absolute error of depth (MAE, m), root mean squared error of the inverse depth ($iRMSE$, $1/km$) and mean absolute error of the inverse depth ($iMAE$, $1/km$) are reported.

5.2 Implementation Details

The proposed network is implemented in PyTorch and optimized with Adam (β_1=0.9, β_2=0.999) and a learning rate of 1e-3. Our model is trained on NVIDIA GeForce RTX 2080 with random change of brightness and contrast, random dropout part of disparity inputs (see Suppl. for more details) and random cropping to 512×256 as data augmentation. We initialize the network with random parameters. A weight decay of 1e-4 is applied for regularization.

For KITTI datasets, the network is trained on Depth Completion dataset for 20 epochs with a batch size of 4 and is tested on two KITTI datasets. The weighting parameter in loss fuction are set as $\omega_s = 0.8$, $\omega_m, \omega_l = 0.1$ at first 5

Table 2. Comparison on the KITTI Stereo 2015 dataset.

Methods	Input	$> 3px \downarrow$	$> 2px \downarrow$	$> 1px \downarrow$	$EPE \downarrow$
GC-Net [19]	Stereo	4.24	5.82	9.97	-
CoEx [2]	Stereo	3.82	5.59	10.67	1.06
Prob. Fusion [22]	Stereo + LiDAR	5.91	-	-	-
Park et al. [28]	Stereo + LiDAR	4.84	-	-	-
CCVN [37]	Stereo + LiDAR	3.35	4.38	6.79	-
LSMD-Net(ours)	Stereo + LiDAR	**2.37**	**3.18**	**5.19**	**0.86**

Table 3. Comparison on the KITTI Depth Completion dataset.

Methods	Input	$MAE \downarrow$	$iMAE \downarrow$	$RMSE \downarrow$	$iRMSE \downarrow$
MSG-CHN [20]	Mono + LiDAR	0.2496	1.11	0.8781	2.59
Park et al. [28]	Stereo + LiDAR	0.5005	1.38	2.0212	3.39
SCADC [38]	Stereo + LiDAR	0.4015	1.94	1.0096	3.96
CCVN [37]	Stereo + LiDAR	0.2525	0.81	0.7493	**1.40**
LiStereo [41]	Stereo + LiDAR	0.2839	1.10	0.8322	2.19
VPN [6]	Stereo + LiDAR	**0.2051**	0.99	**0.6362**	1.87
LSMD-Net(ours)	Stereo + LiDAR	0.2100	**0.79**	0.8845	1.85

epochs, $\omega_l = 0.8$, ω_m, $\omega_s = 0.1$ for another 5 epochs and $\omega_m = 0.7$, $\omega_s = 0.2$, $\omega_l = 0.1$ after 10 epochs. Following [37], input images are bottom-cropped to 1216×256 for there is no ground truth on the top. For our Livox-stereo dataset, we fine-tune the network pretrained on KITTI Depth Completion dataset for another 200 epochs. The weighting parameter in loss fuction are set as $\omega_m = 1.0$, $\omega_s = 0.25$, $\omega_l = 0.125$.

5.3 Results on KITTI Stereo 2015 dataset

We compared the performance of LSMD-Net with stereo matching methods [19, 2] and other publicly available LiDAR-stereo fusion methods [22, 28, 37]. Note that CoEx [2] is our baseline stereo matching method. Quantitative results in Table 2 shows that our method outperforms other methods in terms of disparity metrics. This further demonstrates the advantage of our LSMD-Net in depth prediction since disparity is inversely proportional to depth.

5.4 Results on KITTI Depth Completion dataset

We converted predicted disparity maps into depth maps and compared our LSMD-Net with other depth prediction methods in Table 3. Our method is comparable to other depth prediction methods in terms of depth prediction.

A qualitative comparison on test set is shown in Fig. 6. LiDAR completion (MSG-CHN) is more accurate than stereo matching (CoEx) in depth measurement, but has poor performance at the upper side of maps due to the absence of point clouds. Stereo matching is less precise and performs bad in fine structure, whereas is more stable than LiDAR. Our methods can leverage the unique characteristics of different sensors and provide accurate depth measurements throughout maps.

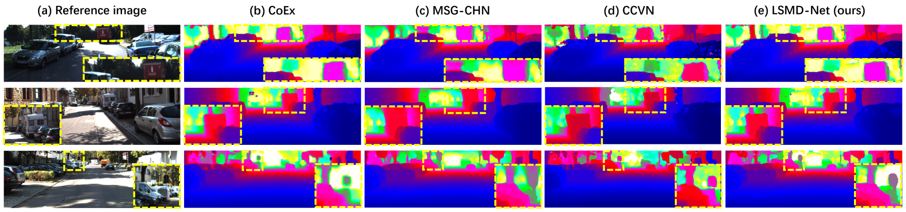


Fig. 6. Qualitative results on KITTI Depth completion test set. Predicted depth map of three scenes from methods based on different sensors are illustrated. We zoom-in the boxes of interest at the bottom on maps.

Table 4. Quantitative results on Livox-stereo test set.

Methods	Input	$> 3px \downarrow$	$EPE \downarrow$	$MAE \downarrow$	$RMSE \downarrow$
CoEx [2]	Stereo	12.72	2.80	-	-
MSG-CHN [20]	Mono + LiDAR	-	-	0.3437	1.08
CCVN [37]	Stereo + LiDAR	6.57	1.86	0.6569	1.85
LSMD-Net(ours)	Stereo + LiDAR	**5.28**	**1.32**	**0.1957**	**0.93**

5.5 Results on Livox-stereo dataset

The proposed method was further evaluated on home-made Livox-stereo dataset. LSMD-Net is compared with MSG-CHN and CoEx on depth maps and disparity maps respectively in Table 4. Our method has obvious advantage over other depth sensing methods in all indicators.

Qualitative results can be found in Fig. 7. As mentioned in Sec. 3.2, the weight of two modes α determines the branch confidence. The α map in Fig. 7(e) presents unique Livox scanning pattens and discontinuous edges of objects, which indicates that LiDAR completion is less reliable in areas without LiDAR measurements and at the edges of objects. Our method can capture the advantages of different sensors using this map.

5.6 Ablation Study

Ablation study is performed on Livox-stereo dataset to study the effect of using different probability distribution models. Four distribution models are tested and their results are reported in Table 5. The Laplacian distribution over disparity we select outperforms others.

Table 5. Comparison of different probability distribution model.

Models	$> 3px \downarrow$	$EPE \downarrow$	$MAE \downarrow$	$RMSE \downarrow$
Gaussian distribution over depth	7.93	1.76	0.3294	1.29
Gaussian distribution over disparity	5.71	1.40	0.2102	0.94
Laplacian distribution over depth	6.15	1.47	0.2750	1.32
Laplacian distribution over disparity	**5.28**	**1.32**	**0.1957**	**0.93**

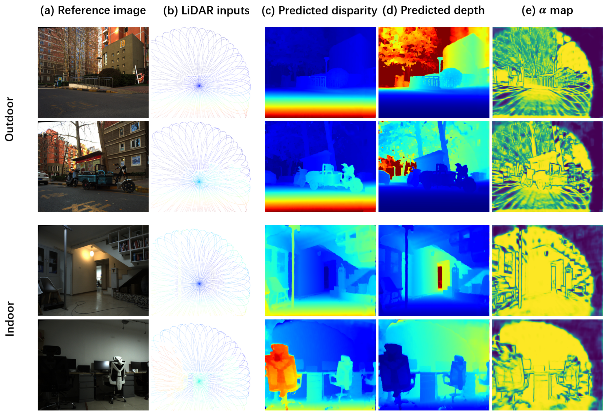


Fig. 7. Qualitative results of our LSMD-Net on Livox-stereo dataset. The brighter part of (e) indicates that LiDAR completion branch is more reliable and the darker part indicates that stereo matching branch is more reliable.

Table 6. Computational time of different methods (unit: millisecond).

Methods	GC-Net [19]	CCVN [37]	VPN [6]	SCADC [38]	CoEx [2]	LSMD-Net(ours)
Time	962	1011	1400	$\approx$ 800	22	27

5.7 Computational Time

We provide a reference for computational time on KITTI in Table 6. The proposed method takes a little bit longer time (5ms) than baseline method CoEx [2], but provide significant improvement in performance, validating the efficiency of our fusion scheme. Other stereo matching method [19] and LiDAR-Stereo fusion method [37, 38, 6] take much more times than our method.

6 Conclusion

In this work, we treat multisensor fusion as a multimodal prediction problem and present a real-time dual-branch LiDAR-Stereo fusion method for the task of efficient depth sensing. The proposed method utilizes mixture density network to predict a bimodal Laplacian distribution at each pixel. Each distribution captures information from stereo matching and LiDAR completion and provide a measure of confidence for them. Our method excels in terms of accuracy and computational time on both KITTI and our home-made Livox-stereo datasets.

Acknowledgement. This work was supported by the National Key Research and Development Program of China under Grant 2020AAA0108302

References

1. Badino, H., Huber, D., Kanade, T.: Integrating lidar into stereo for fast and improved disparity computation. In: 2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission. pp. 405–412 (2011). https://doi.org/10.1109/3DIMPVT.2011.58
2. Bangunharcana, A., Cho, J.W., Lee, S., Kweon, I.S., Kim, K.S., Kim, S.: Correlate-and-excite: Real-time stereo matching via guided cost volume excitation. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3542–3548 (2021). https://doi.org/10.1109/IROS51168.2021.9635909
3. Bishop, C.M.: Mixture density networks. IEEE Computer Society (1994)
4. Chang, J.R., Chen, Y.S.: Pyramid stereo matching network. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5410–5418 (2018). https://doi.org/10.1109/CVPR.2018.00567
5. Cheng, X., Zhong, Y., Dai, Y., Ji, P., Li, H.: Noise-aware unsupervised deep lidar-stereo fusion. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6332–6341 (2019). https://doi.org/10.1109/CVPR.2019.00650
6. Choe, J., Joo, K., Imtiaz, T., Kweon, I.S.: Volumetric propagation network: Stereo-lidar fusion for long-range depth estimation. IEEE Robotics and Automation Letters **6**(3), 4672–4679 (2021). https://doi.org/10.1109/LRA.2021.3068712
7. Civera, J., Davison, A.J., Montiel, J.M.M.: Inverse depth parametrization for monocular slam. IEEE Transactions on Robotics **24**(5), 932–945 (2008). https://doi.org/10.1109/TRO.2008.2003276
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009). https://doi.org/10.1109/CVPR.2009.5206848
9. Gandhi, V., Čech, J., Horaud, R.: High-resolution depth maps based on tof-stereo fusion. In: 2012 IEEE International Conference on Robotics and Automation. pp. 4742–4749 (2012). https://doi.org/10.1109/ICRA.2012.6224771
10. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 3354–3361 (2012). https://doi.org/10.1109/CVPR.2012.6248074
11. Godard, C., Aodha, O.M., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6602–6611 (2017). https://doi.org/10.1109/CVPR.2017.699
12. Guillaumes, A.B.: Mixture density networks for distribution and uncertainty estimation. Ph.D. thesis, Universitat Politècnica de Catalunya. Facultat d'Informàtica de Barcelona (2017)
13. Guo, X., Yang, K., Yang, W., Wang, X., Li, H.: Group-wise correlation stereo network. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3268–3277 (2019). https://doi.org/10.1109/CVPR.2019.00339
14. Guzman-Rivera, A., Batra, D., Kohli, P.: Multiple choice learning: Learning to produce multiple structured outputs. Advances in neural information processing systems **25** (2012)
15. Hirschmuller, H.: Stereo processing by semiglobal matching and mutual information. IEEE Transactions on Pattern Analysis and Machine Intelligence **30**(2), 328–341 (2008). https://doi.org/10.1109/TPAMI.2007.1166

16. Huang, Z., Fan, J., Cheng, S., Yi, S., Wang, X., Li, H.: Hms-net: Hierarchical multi-scale sparsity-invariant network for sparse depth completion. IEEE Transactions on Image Processing **29**, 3429–3441 (2020). https://doi.org/10.1109/TIP.2019.2960589
17. Häger, G., Persson, M., Felsberg, M.: Predicting disparity distributions. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 4363–4369 (2021). https://doi.org/10.1109/ICRA48506.2021.9561617
18. Itseez: Open source computer vision library. https://github.com/itseez/opencv (2015)
19. Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A., Bry, A.: End-to-end learning of geometry and context for deep stereo regression. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 66–75 (2017). https://doi.org/10.1109/ICCV.2017.17
20. Li, A., Yuan, Z., Ling, Y., Chi, W., Zhang, S., Zhang, C.: A multi-scale guided cascade hourglass network for depth completion. In: 2020 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 32–40 (2020). https://doi.org/10.1109/WACV45572.2020.9093407
21. Liu, Z., Zhang, F., Hong, X.: Low-cost retina-like robotic lidars based on incommensurable scanning. IEEE/ASME Transactions on Mechatronics **27**(1), 58–68 (2022). https://doi.org/10.1109/TMECH.2021.3058173
22. Maddern, W., Newman, P.: Real-time probabilistic fusion of sparse 3d lidar and dense stereo. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 2181–2188 (2016). https://doi.org/10.1109/IROS.2016.7759342
23. Makansi, O., Ilg, E., Cicek, Z., Brox, T.: Overcoming limitations of mixture density networks: A sampling and fitting framework for multimodal future prediction. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7137–7146 (2019). https://doi.org/10.1109/CVPR.2019.00731
24. Mayer, N., Ilg, E., Häusser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T.: A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4040–4048 (2016). https://doi.org/10.1109/CVPR.2016.438
25. Menze, M., Geiger, A.: Object scene flow for autonomous vehicles. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3061–3070 (2015). https://doi.org/10.1109/CVPR.2015.7298925
26. Nickels, K., Castano, A., Cianci, C.: Fusion of lidar and stereo range for mobile robots. In: Int. Conf. on Advanced Robotics (2003)
27. Pang, J., Sun, W., Ren, J.S., Yang, C., Yan, Q.: Cascade residual learning: A two-stage convolutional neural network for stereo matching. In: 2017 IEEE International Conference on Computer Vision Workshops (ICCVW). pp. 878–886 (2017). https://doi.org/10.1109/ICCVW.2017.108
28. Park, K., Kim, S., Sohn, K.: High-precision depth estimation with the 3d lidar and stereo fusion. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). pp. 2156–2163 (2018). https://doi.org/10.1109/ICRA.2018.8461048
29. Poggi, M., Pallotti, D., Tosi, F., Mattoccia, S.: Guided stereo matching. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 979–988 (2019). https://doi.org/10.1109/CVPR.2019.00107
30. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)

31. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4510–4520 (2018). https://doi.org/10.1109/CVPR.2018.00474
32. Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nešić, N., Wang, X., Westling, P.: High-resolution stereo datasets with subpixel-accurate ground truth. In: German conference on pattern recognition. pp. 31–42. Springer (2014)
33. Schöps, T., Schönberger, J.L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A.: A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2538–2547 (2017). https://doi.org/10.1109/CVPR.2017.272
34. Shivakumar, S.S., Mohta, K., Pfrommer, B., Kumar, V., Taylor, C.J.: Real time dense depth estimation by fusing stereo with sparse depth measurements. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 6482–6488 (2019). https://doi.org/10.1109/ICRA.2019.8794023
35. Tosi, F., Liao, Y., Schmitt, C., Geiger, A.: Smd-nets: Stereo mixture density networks. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8938–8948 (2021). https://doi.org/10.1109/CVPR46437.2021.00883
36. Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., Geiger, A.: Sparsity invariant cnns. In: 2017 International Conference on 3D Vision (3DV). pp. 11–20 (2017). https://doi.org/10.1109/3DV.2017.00012
37. Wang, T.H., Hu, H.N., Lin, C.H., Tsai, Y.H., Chiu, W.C., Sun, M.: 3d lidar and stereo fusion using stereo matching network with conditional cost volume normalization. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5895–5902 (2019). https://doi.org/10.1109/IROS40897.2019.8968170
38. Wu, C.Y., Neumann, U.: Scene completeness-aware lidar depth completion for driving scenario. In: ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 2490–2494 (2021). https://doi.org/10.1109/ICASSP39728.2021.9414295
39. Xie, Y., Lei, D., Ting, S., Yeyu, F., Jian, L., Xinglong, C., Hanxi, Y., Shuixin, D., Junwei, X., Baohua, C.: A4lidartag: Depth-based fiducial marker for extrinsic calibration of solid-state lidar and camera. IEEE Robotics and Automation Letters pp. 1–1 (2022). https://doi.org/10.1109/LRA.2022.3173033
40. Yang, G., Song, X., Huang, C., Deng, Z., Shi, J., Zhou, B.: Drivingstereo: A large-scale dataset for stereo matching in autonomous driving scenarios. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 899–908 (2019). https://doi.org/10.1109/CVPR.2019.00099
41. Zhang, J., Ramanagopal, M.S., Vasudevan, R., Johnson-Roberson, M.: Listereo: Generate dense depth maps from lidar and stereo imagery. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 7829–7836 (2020). https://doi.org/10.1109/ICRA40945.2020.9196628
42. Zhu, Y., Zheng, C., Yuan, C., Huang, X., Hong, X.: Camvox: A low-cost and accurate lidar-assisted visual slam system. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 5049–5055 (2021). https://doi.org/10.1109/ICRA48506.2021.9561149