LSMD-Net Supplementary Materials

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

Summary of content: In Sec 1, we show some details of our network structure. In Sec. 2, we begin with the statistical information of our dataset. Then we illustrate the error calculation of stereo camera, the data acquisition process and some examples in our dataset. In Sec. 3, we report the underflow problem encountered during training and give a solution. Besides, we report ablation studies about more aspects. Finally, we describe in detail the training/validation/test setting for experiments on KITTI.

1 Network Details

1.1 LiDAR Completion Branch

We summarize the detailed layer-by-layer LiDAR completion branch configurations in Table 1.

1.2 Mixture Density Module

We summarize the detailed layer-by-layer mixture density module configurations in Table 5.

2 Livox-stereo Dataset Details

2.1 Statistics of Subsets

Our dataset contains both indoor and outdoor scenes in residential areas. 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 Table 2.

* Corresponding author

Table 1. Structure details of the LiDAR completion branch. H, W represent the height and the width of the input image. k is the index of the hourglass. S2 denotes a convolution stride of 2. If not specified, each convolution is with a batch normalization and ReLU. $*$ denotes the batch normalization is not included. $**$ denotes convolution only.

Name	Layer properties	Output size
	Image Feature Conv	
conv_0	$G_{ref}[0]$: 3×3 Conv	$H/2 \times W/2 \times 32$
conv_1	$G_{ref}[1]$: 3×3 Conv	$H/4 \times W/4 \times 32$
conv_2	$G_{ref}[2]$: 3×3 Conv	$H/8 \times W/8 \times 32$
conv_3	$G_{ref}[3]$: 3×3 Conv	$H/16 \times W/16 \times 32$
	Basic Depth Hourglass	
conv_d_0$^{k}*$	D_s: 3×3 Conv	$H/2^k \times W/2^k \times 32$
conv_d_1$^{k}*$	3×3 Conv	$H/2^k \times W/2^k \times 32$
conv_d_2$^{k}*$	3×3 Conv S2	$H/2^{k+1} \times W/2^{k+1} \times 32$
conv_d_3$^{k}*$	3×3 Conv	$H/2^{k+1} \times W/2^{k+1} \times 32$
conv_d_4$^{k}*$	3×3 Conv S2	$H/2^{k+2} \times W/2^{k+2} \times 32$
conv_d_5$^{k}**$	3×3 Conv	$H/2^{k+2} \times W/2^{k+2} \times 32$
conv_u_0$^{k}*$	conv_d_5^{k}+conv_(k+2): 3×3 Deconv S2	$H/2^{k+1} \times W/2^{k+1} \times 32$
conv_u_1$^{k}*$	3×3 Conv	$H/2^{k+1} \times W/2^{k+1} \times 32$
conv_u_2$^{k}*$	conv_u_1^{k}+conv_(k+1): 3×3 Deconv S2	$H/2^k \times W/2^k \times 32$
conv_u_3$^{k}*$	3×3 Conv	$H/2^k \times W/2^k \times 32$
conv_u_4$^{k}*$	3×3 Conv	$H/2^k \times W/2^k \times 32$
conv_u_5$^{k}**$	3×3 Conv	$H/2^k \times W/2^k \times 1$

Table 2. Dataset splits

Subset	Train	Val	Test	Total
Indoor	86	13	24	123
Outdoor	268	40	76	384
Total	354	53	100	507

2.2 Error Calculation of Stereo Camera

In a binocular system, B is the length of baseline (the distance between two cameras) and f is the focal length (unit: pixel). Let $\hat{d}$ be the predicted disparity and $\hat{D}$ be the corresponding depth value. The relationship between $\hat{D}$ and $\hat{d}$ can be formulated as

$$\hat{D} = \frac{B \cdot f}{\hat{d}}. \tag{1}$$

Supposing there is a deviation $\triangle\hat{d}$ in disparity $\hat{d}$, the corresponding depth $\hat{D}$ will have a deviation $\triangle\hat{D}$. Then Eq. 1 can be further formulated as

$$\hat{D} + \triangle\hat{D} = \frac{B \cdot f}{\hat{d} + \triangle\hat{d}}. \tag{2}$$

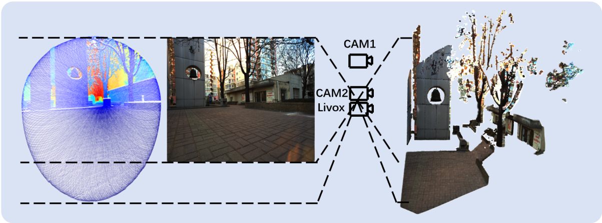

Fig. 1. Illustration of our data acquisition process.

Substituting Eq. 1 into Eq. 2, we can get the relationship between $\triangle\hat{d}$ and $\triangle\hat{D}$

$$\triangle\hat{D} = \hat{D} - \frac{B\cdot f}{\hat{d}+\triangle\hat{d}} = \hat{D} - \frac{1}{\frac{\hat{d}}{B\cdot f}+\frac{\triangle\hat{d}}{B\cdot f}} = \hat{D} - \frac{1}{\frac{1}{\hat{D}}+\frac{\triangle\hat{d}}{B\cdot f}} = \hat{D}(1-\frac{1}{1+\frac{\triangle\hat{d}\cdot\hat{D}}{B\cdot f}}), \quad (3)$$

where B and f is determined by the binocular system. In our case, $B\cdot f$ is about 230 (unit: meter · pixel). Suppose that there is one pixel error in stereo matching, in other words, given $\triangle d = 1$ pixel. We can find that $\triangle\hat{D}$ increases with the increase of $\hat{D}$. It is easy to get from Eq. 3 that in our system, when depth $\hat{D}$ is within 3 meters, depth error $\triangle\hat{D}$ is less than 4 centimeters.

2.3 Data Acquisition Process

The data acquisition process of our dataset is illustrated in Fig. 1. Stereo images were collected by rectified stereo camera. For the sake of simplicity, only the reference figure (captured by the camera below) is shown in Fig. 1. Depth maps are formed by projecting point clouds collected by Livox LiDAR onto the reference image plane. For dense depth maps, we projected point clouds accumulated in 3 seconds, and for sparse depth maps, we projected point clouds accumulated in 0.3 seconds. Note that Livox has a a conical shaped FoV spanning 70.4° and therefore cannot cover the whole image area. Besides, there is a rolling shutter effect when collecting data in dynamic scenes. Rolling shutter is a type of distortion when there exists relative motion between sensor and objects. LiDAR sensors will create streaking artifacts along their direction of motion relative to objects. Considering the long accumulating time, we kept the Livox LiDAR stationary and only captured static scenes to avoid serious rolling shutter effect.

2.4 Examples in Dataset

In Fig. 2 and Fig. 3, we provide some examples of outdoor and indoor subset in home-made Livox-stereo dataset respectively. The outdoor subset contains residential scenes under different lighting conditions.

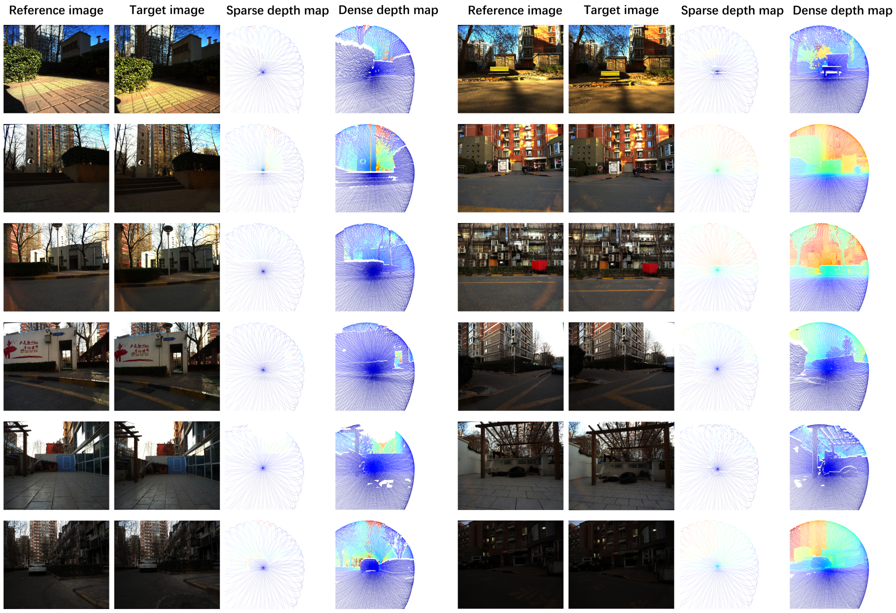

Fig. 2. Some examples of the outdoor subset in our Livox-stereo dataset.

3 Experiment Details

3.1 The Underflow Problem

The underflow problem is that computer system tries to represent a number and this number is too small to be represented by computer. We encountered this problem when computing the negative logarithm of the likelihood loss. If we extend the negative logarithm of the bimodal mixture density distribution P_m, we obtain

$$-\log P_m(d) = -\log(\frac{\alpha}{2b_s}e^{-\frac{\mu_s-d}{b_s}} + \frac{1-\alpha}{2b_l}e^{-\frac{\mu_l-d}{b_l}}). \tag{4}$$

Eq. 4 is part of our loss function. There exist lots of situations when we obtain a very small number and the computer interprets it as absolute zero. These absolute zero may cause underflow problems when there is a logarithm function between operations.

To solve the above problem, we use the log-sum-exp trick to eliminate the numerically unstable behaviour of a logarithm of a sum of exponential expressions

$$-\log P_m(d) = -\max(x_s, x_l) - \log(e^{x_s - \max(x_s,x_l)} + e^{x_l - \max(x_s,x_l)}), \tag{5}$$

where

$$x_s = \log(\alpha) - \log(b_s) - \frac{\mu_s - d}{b_s} - \log 2, \tag{6}$$

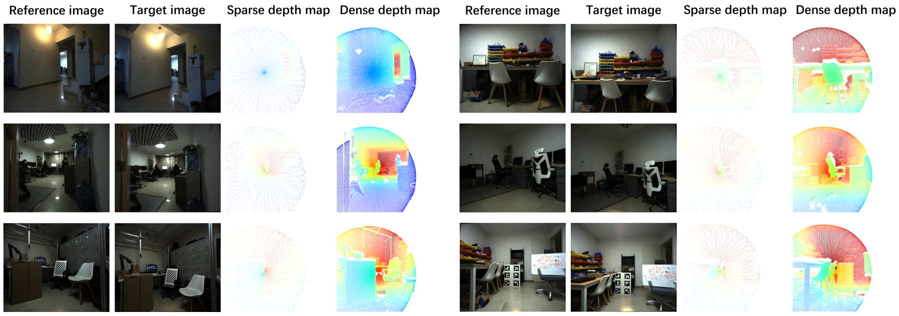

Fig. 3. Some examples of the indoor subset in our Livox-stereo dataset.

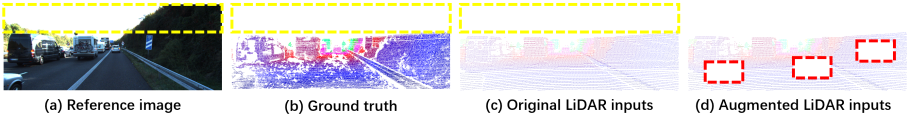

Fig. 4. Illustration of our data agumentation strategy. To make our model work with areas without sparse inputs and supervision (yellow dotted area), we dropout sparse inputs in some patches (red dotted area) randomly to simulate the situation without LiDAR inputs.

$$x_l = \log(1-\alpha) - \log(b_l) - \frac{\mu_l - d}{b_l} - \log 2. \tag{7}$$

Besides, we replace the predicted disparity d with d/D to make our model more stable and easier to converge. D is the maximum disparity value and is set to 192 in our model.

3.2 Random Dropout for data augmentation

Due to the different FoV of LiDAR and cameras, point clouds collected by LiDAR cannot cover the entire image area. Therefore, in LiDAR-based datasets, there exist areas without sparse depth input in training data. Meanwhile, these areas have no dense supervision information. In order to make our model learn to work with areas without sparse disparity inputs, we randomly dropout some patches of disparity inputs as data augmentation, which is illustrated in Fig. 4.

We performed ablation study on our Livox-stereo dataset to study the influence of our data augmentation strategy. Random dropout some patches makes our model learn to deal with areas missing LiDAR inputs, which is illustrated in Fig. 5. There is no ground truth in areas without sparse LiDAR inputs, hence we manually annotated disparities of several points in this areas and report quantitative evaluation results in Table 3.

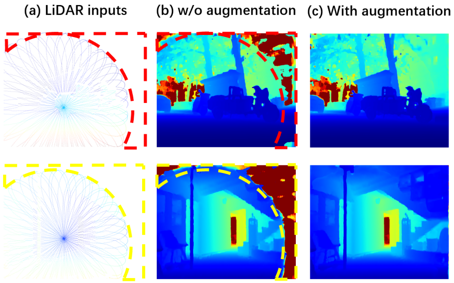

Fig. 5. Qualitative comparison with and without random dropout. (a) and (c) are depth maps with random dropout training, which performs stable out of Livox coverage. (b) and (d) fail with no Livox inputs (the dashed area) for being trainned without random dropout.

Table 3. Comparison with and without random dropout in areas not covered by Livox.

Methods	$> 3px \downarrow$	$EPE \downarrow$	$MAE \downarrow$	$RMSE \downarrow$
w/o random dropout	60.71	6.92	16.7931	25.94
with random dropout	**14.29**	**1.83**	**1.5590**	**2.18**

3.3 Standard for selecting weighting parameters in loss function

Our strategy is first focus on two sub-branches separately (ω_s, ω_l) and then gain weighting to the mixture module (ω_m). Our weighting parameters is selected according to results on KITTI Depth Completion dataset. Ablation study is shown in Tab. a

3.4 Details for Evaluations on KITTI datasets

KITTI Stereo 2015 dataset consists of a training set and a testing set for stereo matching algorithms evaluation. Each sets containing 200 stereo pairs. However, there is no sparse depth maps in the dataset. KITTI Depth Completion dataset provides stereo images and sparse depth maps in training and validation sets, but only monocular images in the testing set. The dataset consists of 42,949 image pairs for training, 3,426 image pairs for validation and 1,000 for testing.

Table a. Ablation study about weighting on KITTI Depth Completion test set.

ω_m	ω_s	ω_l	$MAE \downarrow$	$iMAE \downarrow$	$RMSE \downarrow$	$iRMSE \downarrow$
1	0	0	0.2261	0.85	1.0212	2.41
0.2	0.4	0.4	0.2384	0.89	0.9660	2.17
0.6	0.2	0.2	0.2203	0.83	0.9539	2.02
$0.1 \rightarrow 0.1 \rightarrow 0.7$	$0.8 \rightarrow 0.1 \rightarrow 0.2$	$0.1 \rightarrow 0.8 \rightarrow 0.1$	**0.2100**	**0.79**	**0.8845**	**1.85**

The sparse depth maps are projected by 3D point clouds collected from 11 consecutive LiDAR sweeps.

We are not able to submit our results on the KITTI Stereo or KITTI Depth Completion benchmark since our input is different from them. Also, there is no ground truth in the testing set. Therefore,we evaluated on the training set in KITTI Stereo 2015 dataset and the validation set in KITTI Depth Completion dataset. To be specific, for KITTI Depth Completion dataset, we split the validation set into 1k pairs for validation and another 1k pairs for testing, the scenes in which are not included in the training set. For KITTI Stereo 2015 dataset, we evaluate our model on 142 stereo pairs among the training set which are provided with high-quality disparity maps and are associated with sparse depth maps in KITTI Depth Completion dataset. The above-mentioned 142 stereo pairs cover 29 scenes in KITTI Depth Completion dataset. Hence, we train our model on the remaining non-overlapping 32 scenes containing 32, 918 stereo pairs.

Table 5. Structure details of the mixture density module. H, W represent the height and the width of the input image. D represents the the maximum disparity value. If not specified, each convolution is with a batch normalization and ReLU. $*$ denotes convolution only.

Name	Layer properties	Output size
conv_s_0*	F_s: 1×1 Conv	$H/4 \times W/4 \times 512$
concat_s_0	conv_s_1, conv_s_2: Concat	$H/4 \times W/4 \times (D/4 + 512)$
conv_s_1*	1×1 Conv	$H/4 \times W/4 \times 256$
concat_s_1	conv_s_1, conv_s_3: Concat	$H/4 \times W/4 \times (D/4 + 256)$
conv_s_2*	1×1 Conv	$H/4 \times W/4 \times 128$
concat_s_2	conv_s_1, conv_s_4: Concat	$H/4 \times W/4 \times (D/4 + 128)$
conv_s_3*	1×1 Conv	$H/4 \times W/4 \times 64$
concat_s_3	conv_s_1, conv_s_5: Concat	$H/4 \times W/4 \times (D/4 + 64)$
conv_s_4*	1×1 Conv	$H/4 \times W/4 \times 1$
upsample_s	Upsample	$H \times W \times 1$
activation_s	ELU+1	$H \times W \times 1$
conv_l_0*	F_l: 1×1 Conv	$H/4 \times W/4 \times 512$
concat_l_0	conv_s_1, conv_s_2: Concat	$H/4 \times W/4 \times (64 + 512)$
conv_l_1*	1×1 Conv	$H/4 \times W/4 \times 256$
concat_l_1	conv_s_1, conv_s_3: Concat	$H/4 \times W/4 \times (64 + 256)$
conv_l_2*	1×1 Conv	$H/4 \times W/4 \times 128$
concat_l_2	conv_s_1, conv_s_4: Concat	$H/4 \times W/4 \times (64 + 128)$
conv_l_3*	1×1 Conv	$H/4 \times W/4 \times 64$
concat_l_3	conv_s_1, conv_s_5: Concat	$H/4 \times W/4 \times (64 + 64)$
conv_l_4*	1×1 Conv	$H/4 \times W/4 \times 1$
upsample_l	Upsample	$H \times W \times 1$
activation_l	ELU+1	$H \times W \times 1$
conv_f_0	F_s: 3×3 Conv	$H/4 \times W/4 \times D/4$
conv_f_1	3×3 Conv	$H/4 \times W/4 \times D/4$
conv_f_2	F_l: 3×3 Conv	$H/4 \times W/4 \times 64$
conv_f_3	3×3 Conv	$H/4 \times W/4 \times 64$
concat_f	conv_f_1, conv_f_3: Concat	$H/4 \times W/4 \times (D/4 + 64)$
conv_f_4	3×3 Conv	$H/4 \times W/4 \times (D/4 + 64)$
conv_f_5	3×3 Conv	$H/4 \times W/4 \times (D/4 + 64)$
conv_f_6*	F_l: 1×1 Conv	$H/4 \times W/4 \times 512$
concat_f_6	conv_f_5, conv_f_6: Concat	$H/4 \times W/4 \times (D/4 + 64 + 512)$
conv_f_7*	1×1 Conv	$H/4 \times W/4 \times 256$
concat_f_7	conv_f_5, conv_f_7: Concat	$H/4 \times W/4 \times (D/4 + 64 + 256)$
conv_f_8*	1×1 Conv	$H/4 \times W/4 \times 128$
concat_f_8	conv_f_5, conv_f_8: Concat	$H/4 \times W/4 \times (D/4 + 64 + 128)$
conv_f_9*	1×1 Conv	$H/4 \times W/4 \times 64$
concat_f_9	conv_f_5, conv_f_9: Concat	$H/4 \times W/4 \times (D/4 + 64 + 64)$
conv_f_10*	1×1 Conv	$H/4 \times W/4 \times 1$
upsample_f	Upsample	$H \times W \times 1$
activation_f	Sigmoid	$H \times W \times 1$