Long Range Stereo Matching by Learning Depth and Disparity

WeiQin Chuah, Ruwan Tennakoon, Reza Hoseinnezhad and Alireza Bab-Hadiashar, Senior Member, IEEE

Abstract—Stereo matching generally involves computation of pixel correspondences and estimation of disparities between rectified image pairs. In many applications including simultaneous localization and mapping (SLAM) and 3D object detection, disparity is particularly needed to calculate depth values. While many recent stereo matching solutions focus on delivering a neural network model that provides better matching and aggregation, little attention has been given to the problems of having bias in training data or selected loss function. As the performance of supervised learning networks largely depends on the properties of training data and its loss function, we will show that by simply allowing the neural network to be aware of a bias, its performance improves. We also demonstrate the existence of bias in both the popular KITTI 2015 stereo dataset and the commonly used smooth L1 loss function. Our solution has two components: The loss is depth-based and has two different parts for foreground and background pixels. The combination of those allows the stereo matching network to evenly focus on all pixels and mitigate the potential of over-fitting caused by the bias. The efficacy of our approach is demonstrated by an extensive set of experiments and benchmarking those against the state-of-the-art results. In particular, our results show that the proposed loss function is very effective for the estimation of depth and disparity for objects at distances beyond 50 meters, which represents the frontier for the emerging applications of the passive vision in building autonomous navigation systems.

Index Terms—Stereo Matching, Depth Estimation, Supervised Learning.

1 INTRODUCTION

Depth estimation is thought to be essential and critical in many applications including robotics [1], augmented reality [2], SLAM [3] and 3D object detection [4] for autonomous navigation. Though in most applications, Light Detection and Ranging (LiDAR) technology is employed to perform depth perception, those devices are expensive and the collected point cloud data are sparse, especially for objects located at greater distance [5]. Alternatively, depth estimation can be accomplished using calibrated stereo images. A calibrated stereo camera system is highly affordable (multiple fold cheaper than LiDARs) and capable of producing dense depth results. However, traditional stereo matching methods such as Semi-Global Matching (SGM) [6] utilises hand-crafted techniques and require user-defined parameters thus making it difficult to build a robust solution for different scenes [7]. Also applications of passive stereo systems remained largely limited to short range (less than 20 meters) tasks, only.

In recent years, researchers had leveraged the strength of deep neural network by replacing the hand-crafted techniques with Convolutional Neural Networks (CNNs) in stereo matching and depth estimation pipeline. Learning-based methods make use of deep neural network to extract unique and robust deep features [8] and to learn strong representations from data [9], achieving promising results even in challenging scenarios such as untextured regions, repetitive patterns and thin structures. Many impressive end-to-end learning based stereo matching methodologies [9], [10], [11], [12], [13], [14] have emerged and some have even managed to top the KITTI [15] stereo benchmark.

Although there has been significant interest in developing passive vision systems, most of the recent works have focused on improving performance by designing more
effective network architectures [11], [14], [16] or robust cost functions [17]. In this context, issues such as (1) imbalance data distribution between foreground and background pixels [18], [19], (2) imbalance data distribution at different depth intervals and (3) disparity based loss function emphasises the training gradient of close distance pixels are largely overlooked. These challenges will be discussed in Section 2. In particular, the imbalance data and cost function problems lead to over-emphasis on background area and objects that are located at close distances, resulting in unreliable depth estimation for objects that are located at further distances. As the above issues cause bias in depth estimations, we often refer to their solutions as bias mitigation strategies. As illustrated in Fig. 1, the result of depth estimation produced by a state-of-the-art method [11] without addressing the bias can be extremely unreliable for far (e.g. more than 40 meters) objects.

To address these issues, we propose two new depth-based loss functions that allow the stereo network to mitigate the bias for depth estimation of close objects caused by the disparity-based loss function and far objects caused by data sparsity. While the disparity-based loss function has strong bias towards close objects, the depth-based loss function has the opposite property. Therefore, by carefully combining the two losses, the stereo network will be able to generate reliable results at all distances with significant improvement for far object measurements. Although the intuition behind this idea is straightforward, the effect is significant and using the proposed loss functions improves the estimation of disparity by a large margin (see section 6.4.5).

Although our aim here is to improve the accuracy of depth estimation for far objects, we also evaluated the proposed loss functions on Scene Flow and KITTI 2015 stereo datasets. Results showed that a simple change of the cost function in the PSMNet [11] method can improve its rank from 98th to 42nd place (recorded on 20th of May 2020). Ablation studies and detailed analysis of our experimental results (presented in section 6.4) illustrate the effectiveness of the proposed solution for depth estimation of far objects. The proposed method not only achieves the best accuracy for far objects, it consistently performs competitively across the whole range.

2 Problem Statement

2.1 Loss function

Performance of a supervised deep learning neural network heavily depends on the choice of its loss function. In the context of supervised stereo matching, disparity-based smooth L1 loss [20] (also known as Huber loss) is commonly selected as the default loss function [9], [11], [13], [16], [21]. However, the mentioned loss function exhibits heavy bias towards nearby objects. This is caused by the reciprocal relationship between disparity D and depth Z which can be express by:

\[ Z(i,j) = \frac{f \times b}{D(i,j)} \]  

where f and b are the focal length of the camera and the baseline distance between the centre of stereo cameras. For example, for KITTI 2015 [15] where the focal length and baseline are 721 and 0.54, one meter depth error at 5 meter distance corresponds to 13 pixel disparity error while it only corresponds to 0.6 pixel error at 25 meter distance. Fig. 2 shows that one metre error at any distance translates to disparity-based losses that would penalise close by objects \( \leq 20m \) much higher than far objects.

2.2 Foreground vs. Background

In addition to the loss function, the performance of supervised deep learning neural network also heavily affected by the underlying distribution of the training data and the accuracy of its ground-truth. As shown in Table 1, the distribution of ground-truth pixels in the KITTI 2015 stereo dataset is heavily skewed towards stationary parts (background) compared to moving ones (foreground). Training a neural network using such data will naturally lead to over-emphasis on the background pixels, which is detrimental to its use in downstream applications such as 3D object detection [18], [19]. In this work, we show that it is beneficial to remove this bias from the training process.

|                  | \( \leq 20m \) | \( > 20m \) | Total |
|------------------|----------------|--------------|-------|
| Foreground       | 14.90%         | 1.91%        | 16.81%|
| Background       | 64.79%         | 18.40%       | 83.19%|
| Total            | 79.69%         | 20.31%       | 100.00%|

**Table 1** Pixel data distribution of KITTI 2015 dataset.

2.3 Close vs. Far distance

Disproportionate pixel distribution in two different depth ranges are also illustrated in Table 1 showing approximately 80% of the pixels in KITTI 2015 stereo dataset have depth value \( \leq 20m \). The extent of the imbalanced pixel distribution along depth values are shown in Fig. 3. This clearly illustrates the heavy bias towards pixels located in 0-20
3 Motivation

We have conducted two experiments on a toy example to demonstrate the adverse effect of depth and proportion bias (similar to what exists in the KITTI dataset) on depth estimation. We also examine the hypothesis of mitigating these biases using weighted loss function in these experiments. The outline of the experiments is to train a multi-layer perceptron (MLP) network to regress a specific function using training data with heavy long tail distribution, mimicking the data distribution of KITTI 2015 stereo dataset (Fig. 3).

The selected function is a combination of sine and quadratic functions and is defined as: \( f(x) = 10 \cdot \sin(x) + 0.01 \cdot x^2 + 10 \). In order to mimic the pixel distribution of KITTI 2015 stereo dataset, we sample our training data using Gaussian distribution \( N(\mu = 20, \sigma = 30) \), and the sampled distribution is shown in Fig. 3 (left). Our MLP model has 15 perceptron layers, with Batch Normalization and ReLU activation, and with hidden size of 64 units. Two sets of experiments namely: (1) vanilla implementation (no weighted loss function) and (2) weighted implementation with weighted loss functions (similar to our proposed method, which will be discussed in Section 5) are conducted. As illustrated in Fig. 3, the vanilla implementation only learnt parts of pattern that were represented by data with high density, leading to accurate predictions for these points but significantly worse performance for parts with lesser density. On the other hand, the results of the latter implementation demonstrated a more accurate prediction for \( x \geq 50 \), where the training data is sparse.

4 RELATED WORKS

4.1 Learning based Stereo Matching Networks

Recent works \cite{8, 22} have shown that stereo matching using deep features has significant performance boost over traditional hand-crafted features like SIFT \cite{23} and ORB \cite{24} features. Existing end-to-end stereo matching networks utilised CNN to (1) extract deep representation from input stereo images, (2) cost volume aggregations and (3) cost volume refinement.

In terms of taxonomy, end-to-end stereo matching networks can be classified into two categories (1) correlation-based and (2) concatenation-based methods. The correlation based networks \cite{12, 25, 26} consists of stacked 2D CNNs layers and has significantly lower processing time due to the high efficiency of 2D convolution. The concatenation based networks \cite{9, 11, 13} consists of combination of 2D CNNs for feature extraction and 3D CNNs for cost volume aggregation and refinement.

A notable example is the work of Mayer et al. \cite{25} that proposed two networks: DispNet, which is inspired by FlowNet \cite{27} and its improved version DispNetC. The former takes concatenated left and right images as input while the latter takes correlated left and right feature maps as inputs. To compute a dense disparity map from those inputs, an hourglass convolutional network was used. Both DispNet and DispNetC outperform MC-CNN \cite{8} and its computation is around 1000 times faster than MC-CNN.

In an end-to-end fashion, Liang et al. \cite{26} proposed iResNet that utilizes the features extracted from the CNN layers to generate and refine the initial disparity map. The initial disparity map and the extracted features were fed to a sub-network that is trained to refine the initial disparity map by enforcing feature consistency.

In contrast to the mentioned approaches, Kendall et al. \cite{9} proposed GC-Net that used Siamese network for feature extraction and constructed a 4D cost volume via shifted concatenation method. Cost aggregation was performed using 3D CNNs. Soft-argmin was implemented to regress aggregated matching cost to disparity at all pixels. The results demonstrated that 3D CNNs can be trained to regularise cost volume and produce unimodal distributions with a single peak.

Similar to Kendall et al. \cite{9}, Chang et al. \cite{11} proposed Pyramid Stereo Matching Network (PSMNet) that included spatial pyramid pooling (SPP) module to incorporate hierarchical context information \cite{28} and 3D stacked hourglass network to regularise the network. Zhang et al. \cite{13} also proposed GA-Net inspired by the popular traditional method for depth estimation called semi-global matching (SGM). GA-Net shares the similar network structure as \cite{9}, \cite{11} but incorporating semi-global and local guided aggregation layers with the aim of aggregating cost volume to incorporate global context while preventing the loss of fine details. Zhang et al. \cite{12} proposed AcfnNet, which aims to handle over-fitting and refine aggregated cost volume by filtering the cost volume using a unimodal distribution peaked at true disparity \cite{17}.

To close the performance gaps between the correlation methods and the shifted concatenation methods, many methods have been suggested to improve correlation based
stereo matching and depth estimation networks without sacrificing their efficiency. For instance, [29] and [30] proposed multi-task learning models. These models, apart from learning to perform stereo matching and depth estimation, are designed to be able to learn how to perform auxiliary tasks such as edge detection [29] and semantic segmentation [30]. In this direction, [21] introduced 2-stage CNN-based cascade residual learning (CRL) model in which the second stage of the model learns to refine the coarse output from the first stage. Similarly, [12] proposed AANet which consists of a new sparse points based representation for intra-scale aggregation and adaptive multi-scale cross aggregation modules.

To leverage the benefits from both correlation and concatenated cost volume, Guo et al. [16] proposed GwcNet that includes both correlation and concatenated cost volumes for matching cost computation and cost aggregation. GwcNet further improved stacked hourglass module proposed in [11]. Their results showed that group-wise correlated features provides better matching cost representations thus the performance drop is less significant when the number of parameters in 3D CNNs is reduced drastically.

4.2 Depth estimation and 3D object detection

Accurate depth information of moving (foreground) objects such as pedestrians, transportation vehicles and cyclist is important in downstream application such as 3D object detection and autonomous navigation. There are several works that concentrate on stereo depth estimation for 3D object detection. For instance, Pon et al. proposed a stereo matching network that focused on objects of interest while neglecting the backgrounds and had proven to be effective in improving the accuracy of 3D object detection [31]. Qian et al. proposed to combine stereo matching and 3D object detection networks into a single pipeline [13] by designing a novel differentiable module to convert predicted depth map to pseudo-LiDAR [4]. They used the same stereo matching network proposed in [32].

4.3 Improving long range depth estimation

Despite the lack of accuracy in stereo-based depth estimation at distances further than 20 meters (discussed in Section 2), the issue has received little attention. You et al. [32] proposed to improve the long range depth estimation by converting a disparity-based stereo matching network [11] into a depth-based stereo matching one. The proposed network converts the disparity cost volume to depth cost volume thus regressing depth value for each pixel instead of disparity. They further proposed a depth propagation algorithm which fuses extremely sparse (4-beam) LiDAR to rectify the initial depth estimation.

5 Proposed Method

In this section, we will discuss our proposed loss functions and the overall framework to address the challenges of existing bias for long range stereo depth estimation as explained in Section 2. The results of our experimental and ablation studies are presented in Section 6.

5.1 Loss Function

As the performance of supervised learning neural network largely depends on its loss function, it is crucial to carefully select the appropriate loss function. Also, a properly designed loss function can mitigate the effect of bias (such as data imbalance, class imbalance) in training dataset, improving overall performance of the trained model [33].

We show that using solely disparity-based smooth L1 loss function would cause the trained model to overfit to nearby objects and background areas, resulting worse accuracy for long range depth estimation. To address this issue, we proposed to redesign the loss function by including foreground and background specific depth-based loss functions.

5.1.1 Disparity-based Loss Function

The disparity-based loss function is to enable the stereo matching network to learn the regression of disparity for each pixel. The disparity-based loss function is defined as:

\[
L_{\text{disp}} = \frac{1}{N} \sum_{i=1}^{N} \text{smooth}_{L_1}(D_i - \hat{D}_i),
\]

in which

\[
\text{smooth}_{L_1}(x) = \begin{cases} 
0.5x^2, & \text{if } |x| < 1 \\
|x| - 0.5, & \text{otherwise}
\end{cases}
\]
where $N$ is the number of valid pixels. $D_i$ and $\hat{D}_i$ are predicted disparity and corresponding ground truth values at pixel $i$, respectively.

### 5.1.2 Depth-based Loss Function

Disparity-based loss functions are often designed to penalise the errors in disparity of close by objects while remaining lenient with errors of objects far from the camera. As it was explained earlier, this generates bias towards nearby objects. To address this issue, we include a depth-based loss function, which is similar to the loss function implemented in the SDN [32].

However, in contrast to the SDN, the disparity cost volume will not be converted to depth cost volume within the network. We instead propose to convert the predicted disparity map to depth map to ensure our proposed network does not regress depth value. As our aim is to build a passive system, we do not include laser measurements to refine our results and use the same network structure as the PSMNet [11]. The predicted dense disparity map, $\hat{D}$, as well as its corresponding ground-truth, $D$, are converted to depth map, $\hat{Z}$ and $Z$ using Eq. (1). Using those, the depth loss function is defined as follows:

$$L_{\text{depth}} = \frac{1}{N} \sum_{i=1}^{N} \text{smooth}_{L_1}(Z_i - \hat{Z}_i).$$  (3)

As depth-based loss functions place more emphasises on pixels with larger depth, they are able to achieve better depth accuracy for far objects. However, this comes at the price of having less accuracy for close by objects (≤ 10 m) [32].

Rather than choosing to implement either disparity-based or depth-based loss function, we proposed to combine the two loss functions in our approach. The depth-based loss function can be seen as a regulariser to prevent the neural network to settle on a solution that is over-fitted to the close distance pixels and vice versa. By combining the merits of both loss functions, the overall framework would be able to predict accurate disparity/depth for objects at a wide range of distances.

### 5.1.3 Weighted Foreground and Background Loss Functions

To mitigate the effect of bias caused by difference in the size of foreground and background areas in training datasets, we proposed to weight the network loss function, accordingly. To generate appropriate weights, we propose to split the depth-based loss function into two terms, one for foreground objects and one for background areas. The loss functions will then be weighted using hyperparameter $\lambda$ to balance the effect on foreground and background learning. We select $\lambda$ using grid search and the effect of $\lambda$ is carefully studied and discussed in Section 6. To extract the foreground from the background, we employ Mask R-CNN [34] pre-trained on CityScapes dataset [35] to perform foreground object segmentation. An example of the segmented foreground object masks is shown in Fig. 5. For simplicity, we only considered transportation vehicles including cars, trucks, vans, buses, bicycles and motorcycles as foreground objects. However, this idea can easily be extended to include other object types. We then combine the object masks and the depth loss function $\hat{L}_{\text{depth}}$ to obtain two new loss functions that are defined as:

$$L_{\text{fg}}^{\text{depth}} = \frac{1}{N} \sum_{i=1}^{N} \text{smooth}_{L_1}(Z_i - \hat{Z}_i) \cdot B_i,$$  (4)

$$L_{\text{bg}}^{\text{depth}} = \frac{1}{N} \sum_{i=1}^{N} \text{smooth}_{L_1}(Z_i - \hat{Z}_i) \cdot (1 - B_i),$$  (5)

where $B$ is the object masks, $L_{\text{fg}}^{\text{depth}}$ is the foreground depth loss and $L_{\text{bg}}^{\text{depth}}$ is the background depth loss. The combined depth loss can be written as:

$$L_{\text{depth}} = \lambda \cdot L_{\text{fg}}^{\text{depth}} + (1 - \lambda) \cdot L_{\text{bg}}^{\text{depth}}$$  (6)

and the overall loss function is proposed as:

$$L = L_{\text{disp}} + \beta \cdot L_{\text{depth}}$$  (7)

where hyperparameter $\beta$ is included to balance the effect of disparity-based and depth-based loss functions.

### 6 Experimental Details

#### 6.1 Datasets

KITTI 2015 stereo dataset contains images of natural scenes of city and rural areas and highways collected in Karlsruhe, Germany. It contains 200 training stereo image pairs with sparse ground-truth disparities collected using LiDAR sensor and 200 testing image pairs without ground-truth disparities. KITTI allows performance evaluation by submitting final results to their evaluation server. Following [11], we perform hold-out validation by splitting the 200 training images into 160 for training and 40 for validation. All the results presented in Section 6.4 are computed using the same validation set unless stated otherwise.

DrivingStereo dataset is a large scale stereo dataset covering a diverse set of driving scenarios and different weather conditions, containing over 174,437 stereo pairs for training and 7751 pairs for testing [36]. Sparse ground-truth disparities are provided for the training sets only. We use the DrivingStereo dataset to pre-trained the stereo matching model before fine-tuning on smaller KITTI dataset. Similarly, the dataset is split into training and validation set.
Four subsets were randomly selected as the validation set while the remaining are used as the training set.

**Scene Flow** dataset is a large collection of synthetic stereo dataset with dense disparity ground truth. Scene Flow comprises three subsets of datasets with different settings, FlyingThings3D, Driving and Monkaa. This dataset consists of 35,454 training and 4,370 testing images. The size of each image is $960 \times 540$. As the maximum disparity in this dataset is larger than our pre-defined maximum disparity value, $D_{\text{max}}$, any pixel with disparity larger than the $D_{\text{max}}$ is neglected in the loss computation. This dataset is used to study the effect of different loss function combinations.

### 6.2 Metrics

We evaluate the performance of disparity estimation of the proposed method using endpoint-error (EPE) and 3-pixel (D1) metric that is implemented by the KITTI benchmark.

EPE computes the mean absolute error of the estimated disparity using the ground truth. The D1 metric on the other hand counts the ratio of pixels with EPE of $< 3$ pixel or $< 5\%$ based on the ground truth.

We also evaluate the depth estimation accuracy for objects up to 80 meters from the camera. Predicted disparity (pixel) are converted into depth (metre) following equation $[\text{1}].$ Evaluation is conducted by calculating the EPE for depth values. This metric provides valuable insights on the performance of depth estimation at different depth range.

### 6.3 Implementation details

The proposed loss functions are implemented in conjunction with network architecture proposed in PSMNet $[\text{11}].$ The network is implemented using PyTorch framework and is trained end-to-end with Adam $(\beta_1 = 0.9, \beta_2 = 0.999)$ optimiser. Our data processing is the same as PSMNet where the input images are normalised and randomly cropped to size $H = 256$ and $W = 512$. The maximum disparity is set to 192. All ground-truth disparity beyond the maximum disparity or below 0 are ignored in our experiments. We trained the model from scratch using the DrivingStereo dataset with a constant learning rate of 0.001 for 10 epochs. The same model was also trained using the Scene Flow dataset to study the effect of disparity-based and depth-based loss function. Similarly, the training was conducted for 10 epochs using constant learning rate of 0.001.

We then used the pre-trained model (with DrivingStereo dataset) and finetuned on KITTI training set for 300 epochs. The learning rate for finetuning process starts at 0.001 and is decreased to 0.0001 after 200 epochs. Following $[\text{11}],$ the finetuning process is prolonged to 1000 epochs with learning rate begins at 0.001 and decreased to 0.0001 after $\frac{2}{3}$ of total epochs before submission to KITTI evaluation server. The batch size is set to 12 for training on 2 NVIDIA RTX 6000 Quadro GPUs.

### 6.4 Experimental Results and Ablation Study

To validate the effectiveness of each component proposed in this work, several experiments with different loss function combinations are conducted using KITTI 2015 validation set as well as DrivingStereo and Scene Flow testing sets.

#### 6.4.1 Ablation study for disparity and depth loss functions

We seek to investigate the regularization property of depth-based loss function and how it impacts the performance of our trained stereo matching network. We do so by conducting four set of experiments using different loss functions. Specifically, we trained the network with (1) disparity loss function only, (2) depth loss function only, (3) disparity and depth loss functions and (4) disparity and weighted foreground/background depth loss functions. Note that in our experiments, the predicted disparity and ground-truth are converted into depth via $[\text{1}]$ and the depth loss is computed by $[\text{6}].$

Depth-based loss function is added to regularize the training, aiming to mitigate the over-fitting caused by the disparity-based loss function, which allows greater training gradient for the pixels located at further distance. The results of these experiments are shown in Table 2 and Table 3. In Table 2 the EPE metric is selected to study the accuracy of different loss functions for depth estimation at different distance ranges. As shown, training using depth-based loss function achieved better accuracy for objects located at greater distances ($\geq 10m$) compared to training using disparity-based loss function.

Also, by combining the two loss functions, the network achieves even better accuracy for objects located beyond 20m. Although the performance of the close distance pixels ($0 - 20m$) have deteriorated slightly (around 1%), this is a relatively small price to pay for significant improvement in the accuracy of long range measurements. It is also interesting to note that depth-based loss function also improves the accuracy of measurements for moving (foreground) objects irrespective of their depth. Table 3 shows that the overall accuracy for moving objects are improved by more than 10%.

It is important to note that in Table 2 training using both loss functions resulted the lowest EPE for all categories (All, Fg, Bg), but the results in Table 3 do not show the same improvement. This is due to the fact the D1 metric counts the ratio of inliers and KITTI dataset has large number of pixels in the 0-20m range and as training with both loss functions achieves less accuracy at these ranges, the overall D1 accuracy is lower. By excluding the short range objects, the overall D1 will still be higher.

We further demonstrate the effectiveness of depth-based loss function using Scene Flow dataset. As illustrated in Fig. 9 by including the depth-based loss function, we have achieved incredibly low error rate for long range objects. The background of the image included in Fig. 9 has disparity values ranging between $[1.1, 1.6]$ pixels.

#### 6.4.2 Ablation study for foreground and background depth loss functions

We tackle the imbalance between foreground and background data by building a novel depth-based loss function using two appropriately weighted depth-based loss functions (foreground specific $L_{\text{depth}}^{fg}$ and background specific $L_{\text{depth}}^{bg}$). The ratio between the foreground and background data, listed in Table 4 suggests a weighting of 0.8 for foreground and 0.2 for background ($\lambda = 0.8$). However, from our experiments, we have found that depth-based loss functions resulted the lowest EPE for all categories (All, Fg, Bg), but the results in Table 3 do not show the same improvement. This is due to the fact the D1 metric counts the ratio of inliers and KITTI dataset has large number of pixels in the 0-20m range and as training with both loss functions achieves less accuracy at these ranges, the overall D1 accuracy is lower. By excluding the short range objects, the overall D1 will still be higher.

We further demonstrate the effectiveness of depth-based loss function using Scene Flow dataset. As illustrated in Fig. 9 by including the depth-based loss function, we have achieved incredibly low error rate for long range objects. The background of the image included in Fig. 9 has disparity values ranging between $[1.1, 1.6]$ pixels.
Accuracy of depth estimation at different depth intervals using different loss functions. The predicted disparity values are converted to depth values following [1]. The pixels are then separated into bins according to their true depth values. The discrepancy between the predicted and ground truth depth is measured using EPE. We showed that depth loss function can effectively improves the accuracy of depth estimation for both foreground and background pixels that are located at far distances.

Fig. 6. These graphs show the relationship between the balancing term $\lambda$ and the D1 error of all pixels (left), foreground objects (middle) and background objects (right). They also show that $\lambda = 0.6$ yields the best performance. Red dotted line indicates the performance of the baseline method (PSMNet).

### Table 2

| Loss Functions | Foreground Depth EPE (m) | Background Depth EPE (m) |
|----------------|--------------------------|--------------------------|
| $L_{\text{disp}}$ | $L_{\text{depth}}$ | $L_{\text{disp}}$ $\&$ $L_{\text{depth}}$ |
| 0-10 | 10-20 | 20-30 | 30-40 | 40-50 | 50-60 | 60-70 | 70-80 | Average |
| $L_{\text{disp}}$ | 0.12 | 0.38 | 0.89 | 1.50 | 2.42 | 3.69 | 5.39 | 6.47 | 2.61 |
| $L_{\text{depth}}$ | 0.12 | 0.35 | 0.87 | 1.51 | 2.48 | 3.68 | 5.12 | 5.89 | 2.50 |
| $L_{\text{disp}}$ $\&$ $L_{\text{depth}}$ | 0.13 | 0.40 | 0.86 | 1.46 | 2.33 | 3.53 | 4.61 | 5.66 | 2.37 |

### Table 3

Ablation study of different loss function combination. DS: DrivingStereo. SF: Scene Flow.

| Loss Functions | KITTI 2015 | DS | SF |
|----------------|------------|----|----|
| $L_{\text{disp}}$ | $L_{\text{depth}}$ | $L_{\text{disp}}$ $\&$ $L_{\text{depth}}$ |
| D1 | D1 | EPE |
| Fg | Bg | All | All | All |
| 1.89 | 1.83 | 1.98 | 0.68 | 1.33 | 1.66 | 1.85 | 1.90 | 0.98 | 1.94 | 1.35 | 1.95 | 1.95 | 0.53 | 1.10 |
| 1.31 | 1.80 | 1.78 | 0.48 | - |

Within expectation, the results demonstrated that $L_{\text{depth}}$ is advantageous for foreground prediction. However, solely including $L_{\text{depth}}$ worsens the accuracy for background objects as well as the overall accuracy. However, $L_{\text{depth}}$ is still required to improve the accuracy of background objects located at far distance. As such, both depth-based loss components are needed to improve the overall accuracy at all distances.

### 6.4.3 Analysis of balancing term $\lambda$

Hyperparameter $\lambda$ balances the contributions of foreground specific $L_{\text{depth}}$ and background specific $L_{\text{depth}}^{Bg}$ components to the total loss. We selected the optimal value for $\lambda$ using grid-search between 0 and 1 with interval of 0.2. As it was mentioned earlier, the ratio between foreground and background data in the KITTI 2015 dataset implies the $\lambda$ to be 0.8 for optimal performance. However, we observed that the overall, foreground and background error curves are similar 'V' shape curves and the optimal results for $\lambda$ is close to 0.6.

Fig. 6 shows that including the $L_{\text{depth}}^{Fg}$ in loss calculations (by setting $\lambda > 0$) lowers the D1 error for foreground objects. However, the effect of including $L_{\text{depth}}$ is less pro-

The discrepancy between the predicted and ground truth depth is measured using EPE. We showed that depth loss function can effectively improves the accuracy of depth estimation for both foreground and background pixels that are located at far distances.
### 6.4.4 Performance analysis of long range depth estimation

In this section, we compare the improvement resulted in long range depth estimation by our proposed method, which is named LR-PSMNet (LR: Long Range) and the current state-of-the-art for long range stereo (SDN [32]). As listed in Table 5, LR-PSMNet improves the performance of PSMNet at all depth ranges and achieves more than 10% improvement beyond 50m. Although SDN offered more accurate measurements at distance between 10 and 60m, those improvements drop significantly at greater distances. It was argued that the poor performance between 50-70m might cause by over-fitting [32]. However, this phenomena is not observe in LR-PSMNet as our loss function combination is based on their experiments where the depth error of PSMNet is different from ours as shown in Table 6. More importantly, our results are on par with the SDN+GDC active method by 4-beam LiDAR, to refine the measurements [32].

#### 6.4.5 KITTI 2015 Leaderboard

Although our work is focused on long range depth estimation, we also subjected the proposed method to KITTI performance evaluation exercise. The overall results of LR-
PSMNet was 2.06% as listed in Table 7. This shows that by carefully redesign the loss function, the rank of the PSMNet [11] method is improved from rank 98 to 42 (recorded on 20th of May 2020). We argue that the higher error in foreground pixels is largely due to the limitation of the designed network as PSMNet, despite being a competitive method, has one of the highest foreground error among all state-of-the-art methods. Although the extension of our approach to other methods is straightforward, as it is, the proposed approach produces better disparity accuracy for foreground and background pixels as compared to other high performing methods [16], [17] - see Fig. 8 for detail.

7 Conclusion
This work addressed the significant issues associated with bias in both dataset composition and disparity loss calculation for long range stereo depth estimation. We provided an effective solution by including foreground and background specific depth-based loss functions. We first tested the proposed solution on a tractable mathematical toy example and showed that using our proposed loss function can significantly mitigate the effect data bias on the network performance. We also showed that by separating the loss calculation for foreground and background objects and weighting those properly to remove the bias, we can improve the passive stereo depth calculation for long range foreground (moving) objects. Our experimental results demonstrated that the proposed method outperforms the state-of-the-art methods for long range stereo depth estimation.

References
[1] T. Fong, J. Nourbakhsh, and K. Dautenhahn, “A survey of socially interactive robots,” Robotics and autonomous systems, vol. 42, no. 3-4, pp. 143–166, 2003.
[2] M. Kanbara, T. Okuma, H. Takemura, and N. Yokoya, “A stereoscopic video see-through augmented reality system based on real-time vision-based registration,” in Proceedings IEEE Virtual Reality 2000 (Cat. No. 00CH37048). IEEE, 2000, pp. 255–262.
[3] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, “Monoslam: Real-time single camera slam,” IEEE transactions on pattern analysis and machine intelligence, vol. 29, no. 6, pp. 1052–1067, 2007.
[4] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Q. Weinberger, “Pseudo-lidar from visual depth estimation: Bridging the gap in 3D object detection for autonomous driving,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 8445–8453.
[5] K. Park, S. Kim, and K. Sohn, “High-precision depth estimation using uncalibrated lidar and stereo fusion,” IEEE Transactions on Intelligent Transportation Systems, 2019.
[6] H. Hirschmuller, “Accurate and efficient stereo processing by semi-global matching and mutual information,” in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), vol. 2. IEEE, 2005, pp. 807–814.
[7] D. Scharstein and R. Szeliski, “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms,” International journal of computer vision, vol. 47, no. 1-3, pp. 7–42, 2002.
[8] J. Zhou and Y. LeCun, “Stereo matching by training a convolutional neural network to compare image patches,” The journal of machine learning research, vol. 17, no. 1, pp. 2287–2318, 2016.
[9] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry, “End-to-end learning of geometry and context for deep stereo regression,” in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 763–772.
[10] Z. Liang, Y. Guo, Y. Feng, W. Chen, L. Qiao, L. Zhou, J. Zhang, and H. Liu, “Stereo matching using multi-level cost volume and multi-scale feature constancy,” IEEE transactions on pattern analysis and machine intelligence, 2019.
[11] J.-R. Chang and Y.-S. Chen, “Pyramid stereo matching network,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5410–5418.
[12] H. Xu and J. Zhang, “Aanet: Adaptive aggregation network for efficient stereo matching,” arXiv preprint arXiv:2004.09548, 2020.
[13] F. Zhang, V. Prisacariu, R. Yang, and P. H. Torr, “Ga-net: Guided aggregation net for end-to-end stereo matching,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 185–194.
[14] X. Cheng, P. Wang, and R. Yang, “Learning depth with convolutional spatial propagation network,” IEEE transactions on pattern analysis and machine intelligence, 2019.
[15] M. Menze, C. Heipke, and A. Geiger, “Object scene flow,” ISPRS Journal of Photogrammetry and Remote Sensing (JPRS), 2018.
[16] X. Guo, K. Yang, W. Yang, X. Wang, and H. Li, “Group-wise correlation stereo network,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3273–3282.
[17] Y. Zhang, Y. Chen, X. Bai, J. Zhou, K. Yu, Z. Li, and K. Yang, “Adaptive unimodal cost volume filtering for deep stereo matching,” arXiv preprint arXiv:1909.03731, 2019.
[18] Qiao, D. Garg, Y. You, Y. You, B. Hariharan, M. Campbell, K. Q. Weinberger, and W.-L. Chao, “End-to-end pseudo-lidar for image-based 3D object detection,” arXiv preprint arXiv:2004.03080, 2020.
[19] X. Wang, W. Yin, T. Kong, Y. Jiang, L. Li, and C. Shen, “Task-aware monocular depth estimation for 3D object detection,” arXiv preprint arXiv:1909.07701, 2019.
[20] R. Girshick, “Fast r-cnn,” in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.
[21] J. Pang, W. Sun, J. S. Ren, C. Yang, and Q. Yan, “Cascade residual learning: A two-stage convolutional neural network for stereo matching,” in Proceedings of the IEEE International Conference on Computer Vision Workshops, 2017, pp. 887–895.
[22] W. Luo, A. G. Schwinger, and R. Urtasun, “Efficient deep learning for stereo matching,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 5695–5703.
[23] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” International journal of computer vision, vol. 60, no. 2, pp. 91–110, 2004.
[24] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: An efficient alternative to sift or surf,” in 2011 International conference on computer vision. Ieee, 2011, pp. 2564–2571.
[25] N. Mayer, E. Ilg, P. Haasner, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, “A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4040–4048.
[26] Z. Liang, Y. Feng, Y. Guo, H. Liu, W. Chen, L. Qiao, L. Zhou, and J. Zhang, “Learning for disparity estimation through feature constancy,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2811–2820.
[27] A. Dosovitskiy, P. Fischer, E. Ilg, P. Haasner, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, “Flownet: Learning optical flow with convolutional networks,” in Proceedings of the IEEE international conference on computer vision, 2015, pp. 2758–2766.
[28] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” IEEE transactions on pattern analysis and machine intelligence, vol. 37, no. 9, pp. 1904–1916, 2015.
[29] X. Song, X. Zhao, H. Hu, and L. Fang, “Edgestereo: A context integrated residual pyramid network for stereo matching,” in Asian Conference on Computer Vision. Springer, 2018, pp. 20–35.
[30] G. Yang, H. Zhao, J. Shi, Z. Deng, and J. Jia, “Segstereo: Exploiting semantic information for disparity estimation,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 636–651.
[31] A. D. Pon, J. Ku, C. Li, and S. L. Waslander, “Object-centric stereo matching for 3d object detection,” arXiv preprint arXiv:1909.07566, 2019.
[32] Y. You, Y. Wang, W.-L. Chao, D. Garg, G. Pleiss, B. Hariharan, M. Campbell, and K. Q. Weinberger, “Pseudo-lidar++: Accurate depth for 3d object detection in autonomous driving,” arXiv preprint arXiv:1906.06310, 2019.
[33] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, “Focal loss for dense object detection,” in Proceedings of the IEEE International Conference on Computer Vision (ICCV), Oct 2017.
[34] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” in *Proceedings of the IEEE international conference on computer vision, 2017*, pp. 2961–2969.

[35] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, “The cityscapes dataset for semantic urban scene understanding,” in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[36] G. Yang, X. Song, C. Huang, Z. Deng, J. Shi, and B. Zhou, “Drivingstereo: A large-scale dataset for stereo matching in autonomous driving scenarios,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 899–908.

[37] S. Duggal, S. Wang, W.-C. Ma, R. Hu, and R. Urtasun, “Deep-pruner: Learning efficient stereo matching via differentiable patch-match,” in *Proceedings of the IEEE International Conference on Computer Vision, 2019*, pp. 4384–4393.
Fig. 7. Visualization of improvement over baseline method (PSMNet) on KITTI 2015 dataset. For each input image, predicted disparity map is illustrated on top row and error map on the bottom row. Improved areas are highlighted with yellow dashed box. The numerical scale for colour mapped on the error maps is provided at the bottom of this page.

Fig. 8. Visualization results on KITTI 2015 dataset comparing our results with AcfNet [17] and GwcNet [16].

Fig. 9. Qualitative results of Scene Flow dataset comparing the performance of disparity-based loss function (right) and the combination of disparity-based and depth-based loss function (middle). The ground truth label is included in the top left and the left RGB image is included in the bottom left. The error map generated using the provided MATLAB script by KITTI is included in the bottom row. By including depth-based loss function, superior results can be obtained especially for the pixels located at very far distance.