PLADE-Net: Towards Pixel-Level Accuracy for Self-Supervised Single-View Depth Estimation with Neural Positional Encoding and Distilled Matting Loss

Juan Luis Gonzalez Bello
juanluisgb@kaist.ac.kr

Munchurl Kim†
mkimee@kaist.ac.kr

Korea Advanced Institute of Science and Technology

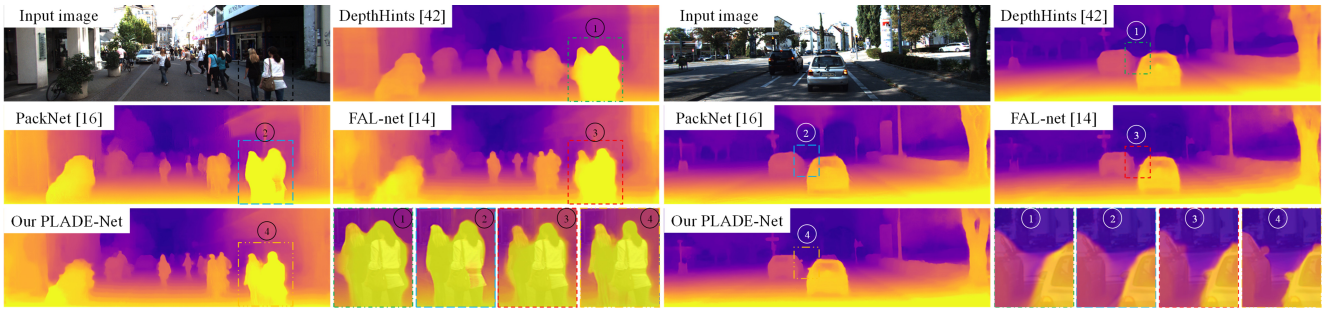


Figure 1. Our proposed PLADE-Net with Neural Positional-Encoding and distilled matting loss estimates depths with pixel-level accuracy.

Abstract

In this paper, we propose a self-supervised single-view pixel-level accurate depth estimation network, called PLADE-Net. The PLADE-Net is the first work that shows remarkable accuracy levels, exceeding 95% in terms of the δ^1 metric on the challenging KITTI dataset. Our PLADE-Net is based on a new network architecture with neural positional encoding and a novel loss function that borrows from the closed-form solution of the matting Laplacian to learn pixel-level accurate depth estimation from stereo images. Neural positional encoding allows our PLADE-Net to obtain more consistent depth estimates by letting the network reason about location-specific image properties such as projection (and potentially lens) distortions. Our novel distilled matting Laplacian loss allows our network to predict sharp depths at object boundaries and more consistent depths in highly homogeneous regions. Our proposed method outperforms all previous self-supervised single-view depth estimation methods by a large margin on the challenging KITTI dataset, with unparalleled levels of accuracy. Furthermore, our PLADE-Net, naively extended for stereo inputs, outperforms the most recent self-supervised stereo methods, even without any advanced blocks like 1D correlations, 3D convolutions, or spatial pyramid pooling. We present extensive ablation studies and experiments that support our method's effectiveness on the KITTI, CityScapes, and Make3D datasets.

1. Introduction

Recent advances in deep learning have shown state-of-the-art (SOTA) results on the challenging single-view depth estimation (SVDE) and stereo disparity estimation (SDE) tasks. In particular, self-supervised methods for SVDE have reached performance levels similar or even superior to the fully-supervised networks [16, 17, 14]. However, the previous SOTA self-supervised methods are unable to predict accurate pixel-level depth estimates, which are often observed along the object's depth boundaries. Predicting pixel-level accurate 3D geometries is essential for robotic grasping, augmented reality, navigation, and 3D object detection.

In this paper, we present a pixel-level accurate depth estimation network (PLADE-Net) with neural positional encoding and a distilled matting Laplacian loss, both of which allow for consistent depth estimates in homogeneous areas and sharp depth predictions along the object boundaries. Our PLADE-Net outperforms the most recent self-supervised SOTA methods [16, 17, 14, 41, 1] (both mono and stereo) by large margins, achieving unparalleled accuracy on the challenging KITTI dataset while keeping a low number of parameters. This paper's contributions are:

1. We propose to exploit and distill the closed-form solution of the matting Laplacian [28] for self-supervision, leading to a novel loss function that allows for pixel-level

†Corresponding author.

accuracy in self-supervised single- and stereo-view DE.

2. We show that neural positional encoding (NPE) can be usefully incorporated into CNNs for depth estimation, as it allows the network to reason about camera distortions, scene orientation, and non-local relationships.
3. We present PLADE-Net, a novel network architecture that incorporates NPE. Our PLADE-Net incorporates multi-scale inputs and a single-scale output, opposite to single-scale inputs and multi-scale outputs in previous works [11, 10, 16, 42]. Relative to previous works, our PLADE-Net doubles the number of filter channels in the early feature extraction layers, and halves the number of filter channels in its bottleneck. These seemingly trivial design choices, already make our PLADE-Net, even without our newly proposed loss functions, to outperform the previous SOTA methods.
4. The PLADE-Net is the first work that shows unmatched accuracy levels for SVDE, exceeding 95% in terms of δ^1 metric on the challenging KITTI[9] dataset.

Figure 1 compares the depth estimate performances of the most recent SOTA methods [42, 16, 14] with respect to our PLADE-Net. As shown in the detailed view of the estimated depth regions (dotted boxes numbered from ① to ④), our PLADE-Net produces very precisely estimated depths along the object boundaries. Simultaneously, the SOTA methods fail by yielding inaccurate object depths that partially leak into the background.

Our paper is organized as follows: In Section 2, we review relevant self-supervised methods for our work; Section 3 presents our PLADE-Net with neural positional encoding and a distilled matting Laplacian loss with in-depth explanations; In Section 4, we provide extensive ablation studies and experiments that support the effectiveness of our contributions; We conclude our work in Section 5.

2. Related Works

Learning-based self-supervised single view depth estimation (SVDE) is a relatively new problem and has rapidly advanced since it was first proposed in the work of Garg *et al*. [8]. Self-supervised SVDE is usually achieved by exploiting the 3D information embedded in datasets that contain multiple captures from the same scene. Previous methods have successfully learned SVDE from stereo pairs [8, 11, 34, 33, 35, 18, 42, 38, 47, 14] and video [46, 10, 16, 17, 15], by training their CNNs for the backward or forward synthesis of the training image samples, given the target view as input. Other works [11, 43, 26, 41, 1] have addressed the less ill-posed problem of stereo depth estimation (SDE), where the left and right views are available during training and testing. As our proposed PLADE-Net learns from stereo images, we only review the methods that learn from stereo in this section for the sake of simplicity.

Learning SVDE from stereo. Among the top-performing SVDE methods that learn from stereo we find the works of [38, 42, 14]. The contemporary works of Tosi *et al*. [38] and Watson *et al*. [42] proposed to guide the training of their SVDE networks with distilled stereo disparity estimates obtained from the classical approach of semi-global matching (SGM) [20, 21]. While Watson *et al*. [42] used the SGM disparity as a proxy label when the resulting photometric loss is lower than the CNN-estimated depth, Tosi *et al*. [38] distilled the SGM proxy label via left-right (LR) consistency checks. Inspired by [38, 42], we distill the matting Laplacian with both photometric and LR-consistency checks in this work.

The recent work of Gonzalez and Kim [14], proposed to "forget about the LiDAR", by learning high-quality depths with a multi-view occlusion module and exponentially quantified disparity volumes. Additionally, they proposed a two-stage training strategy to learn view synthesis and refine their network, called FAL-net[14], for SVDE. While their method obtains the SOTA metrics on the KITTI [9] dataset, their approach is still far from generating pixel-level accurate depths, as shown in Figure 1-③. Their second stage loss functions cannot enforce sharp object depth boundaries, as they are limited by the computed occlusions' quality, leading to sub-optimal estimates.

Learning SDE. Interestingly, the less ill-posed problem of learning stereo disparity estimation (SDE) in a self-supervised manner has been studied less extensively than the single-view case. The most prominent works include those of Wang *et al*. [41] and Aleotti *et al*. [1]. Wang *et al*. [41] proposed to exploit spatiotemporal information by learning SDE from stereo videos. Their "UnOS" learns optical flow, stereo disparity, and camera pose by spatially and temporally projecting the target views into the spatiotemporal reference images and measuring the reconstruction errors to provide means of self-supervision. The work of Aleotti *et al*. proposes to distill the disparity estimates from a monocular disparity competition network to provide additional proxy labels for the SDE task. Aleotti *et al*. achieve the SOTA by training existing networks [10, 3] with their monocular proxy labels, which remove the well-known stereo artifacts caused by occlusions [11, 20, 21, 1].

3. Method

We propose a novel Pixel-Level Accurate Depth Estimation network, called PLADE-Net, with neural positional encoding and a distilled matting Laplacian loss. Architecture-wise, neural positional encoding is incorporated into our PLADE-Net to learn location-specific image features. Training-wise, our PLADE-Net learns single-view depth from stereo pairs in a two-stage training strategy following the previous work [14]. In the first stage of training, our PLADE-Net is trained for simple stereoscopic view

synthesis with a combination of l_1, perceptual [24], and smoothness losses. In the second stage, our network is fine-tuned with an occlusions-free reconstruction loss with the multi-view occlusion module and other secondary smoothness and mirror losses defined in [14]. More importantly, a distilled matting Laplacian loss is newly proposed, allowing the learning of highly accurate pixel-level depth estimates.

In the following subsections, we introduce an image (and inverse depth) formation model (which follows the one defined in [14]) and describe the intuition behind our main contributions for *neural positional encoding* and *a distilled matting Laplacian loss*. We then describe the details of our network architecture and training loss functions in Subsections 3.4 and 3.5, respectively.

3.1. Stereoscopic Image Formation Model

We build our PLADE-Net based on the work of Gonzalez and Kim [14], as their method showed SOTA results for learning single-view depth from stereo images. Therefore, we adopt their image formation model in which the convolutional neural network (CNN) outputs a disparity probability logit volume $\mathbf{D}_L^L$ for a given left input view $\mathbf{I}_L$. $\mathbf{D}_L^L$ can be either progressively projected to the right-view and soft-maxed channel-wise to form the right-from-left disparity probability volume $\mathbf{D}_L^{PR}$ or simply soft-maxed to generate the left disparity probability volume $\mathbf{D}_L^{PL}$. $\mathbf{D}_L^{PR}$ can be used for stereoscopic view synthesis by

$$\mathbf{I}'_R = \sum_{n=0}^{N} g\left(\mathbf{I}_L, d_n\right) \odot \mathbf{D}_{L_n}^{PR}, \tag{1}$$

where $\odot$ indicates the Hadamard product, $g(\cdot)$ denotes a shifting of the input image to the left by d_n pixels, and N is the number of planes in the probability volume $\mathbf{D}_L^{PR}$. On the other hand, $\mathbf{D}_L^{PL}$ can be used to extract the disparity map $\mathbf{D}'_L$, which is learned as a by-product from the view synthesis task, as defined by

$$\mathbf{D}'_L = \sum_{n=0}^{N} d_n \mathbf{D}_{L_n}^{PL} \tag{2}$$

We also adopt the exponential disparity quantization in [14], as it is well-posed for the depth estimation task. Exponential quantization takes into account the inverse relationship between disparity and depth by distributing far- and close-by quantization levels more evenly and is given by

$$d_n = d_{max} e^{\ln d_{max}/d_{min}(n/N-1)}, \tag{3}$$

where d_{max} and d_{min} are the minimum and maximum disparity hyper-parameters. For a fair comparison with [14], we set $d_{max} = 300$ and $d_{min} = 2$ for all our experiments.

3.2. Neural Positional Encoding

It is well-known that convolutional neural networks on their own are very well capable of encoding positional information [22]. However, since the local CNN filters are shared across spatial locations, a network, trained with randomly cropped patches from the original image data, will struggle to learn location-specific features, such as lens or projection distortions, ground versus sky regions, and potentially non-local relationships. In particular, projection distortions make the objects near image borders appear to be more stretched than those in the image centers, the degree of which often depends on the camera focal length. This is a potential source of confusion for the CNNs, as two objects in the same distance to a camera will be projected differently on the camera plane, depending on their relative position to the resulting image, as illustrated in Figure 2.

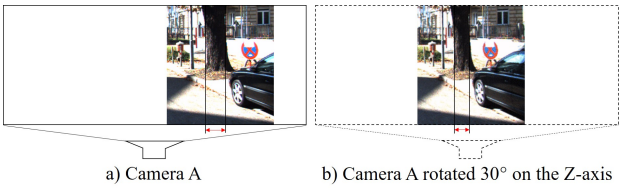


Figure 2. An illustration of projection distortions in image borders. Objects closer to the image borders appear stretched and closer, like the three in (a) in comparison with the three in (b).

The relative object size is an essential cue for depth estimation. It can affect the estimation accuracy if a network does not have a means of understanding the locations of the training patches in their original images. To provide the network with a mechanism to account for the likelihood of objects being stretched when they are located close to the image borders, we propose neural positional encoding (NPE) for depth estimation. We realize NPE into our PLADE-Net as the concatenation of deep positional features at each encoder stage. A deep positional feature map $\mathbf{F}_{npe}$ is obtained by processing the pixel location $\boldsymbol{p} = (x, y)$ information of each patch with two fully-connected layers with exponential linear unit (ELU) activations, which is given by:

$$\mathbf{F}_{npe}(\boldsymbol{p}) = elu(\boldsymbol{w}_2 \cdot elu(\boldsymbol{w}_1 \cdot \boldsymbol{p} + b_1) + b_2), \tag{4}$$

where $w_{1,2}$ and $b_{1,2}$ are the learnable weights and biases of our neural positional encoding layers. The operation in Eq. 4 can be trivially realized with 1×1 convolutions in available deep learning libraries. Note that in contrast with [6], we do not concatenate x- and y-coordinates of $\boldsymbol{p}$, but do concatenate our deep positional feature maps into the downstream convolutional layers in our PLADE-Net. It should be noted again that, in our neural positional encoding, $\boldsymbol{p} = (x, y)$ are the pixel locations of the patches relative to their original images before cropping.

3.3. Distilling the Matting Laplacian

The closed-form solution to the matting Laplacian [28] is a useful tool in classical low-level computer vision. It can sharply segment an input image based on pixel intensity and proximity in the input image, a roughly estimated or user-defined segmentation map, and a confidence map.

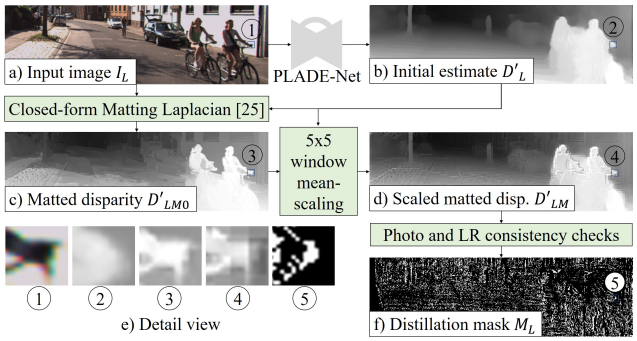


Figure 3. Matted disparity distillation process.

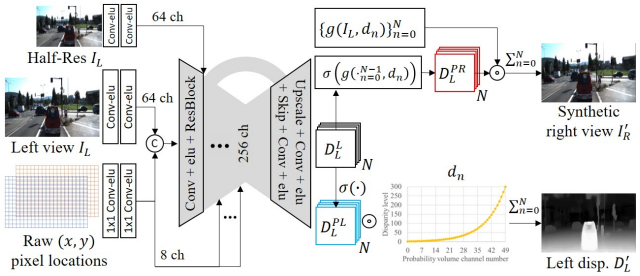


Figure 4. High-level overview of our proposed PLADE-Net.

Image matting and the matting Laplacian have been used to refine depth estimates [30, 23, 2, 48], generating structurally sharp but incorrectly labeled matted depth maps. We exploit the strong features of the matting Laplacian to learn highly accurate pixel-level depth in a self-supervised fashion, remedying its weak points by distilling the matted depth maps with photometric and left-right consistency [11] checks.

Our matting Laplacian distillation process is depicted in Figure 3. Given an input image $\mathbf{I}_L$ and its initial depth estimate $\mathbf{D}'_{L0}$ by our PLADE-Net, we generate a matted disparity $\mathbf{D}'_{LM0}$ following [28]. As can be observed in Figure 3-(c), $\mathbf{D}'_{LM0}$ is very sharp, but many pixels are wrongly labeled, such as the tree at the left-hand side of $\mathbf{I}_L$, the biker in the background, and the overall road depths. To remedy this, we apply a 5×5 local window mean scaling to obtain the locally scaled matted disparity map $\mathbf{D}'_{LM}$, as shown in Figure 3-(d). Then, we estimate a distillation mask $\mathbf{M}_L$ via photometric and left-right consistency checks as given by:

$$\mathbf{M}_L = \left[\left|\mathbf{I}_L - g(\mathbf{I}_R, \mathbf{D}'_{LM})\right| < \left|\mathbf{I}_L - g(\mathbf{I}_R, \mathbf{D}'_L)\right|\right] \odot \left[\left|\mathbf{D}'_{LM} - g(\mathbf{D}'_{RM}, \mathbf{D}'_{LM})\right| < \left|\mathbf{D}'_L - g(\mathbf{D}'_R, \mathbf{D}'_L)\right|\right], \quad (5)$$

where $\mathbf{D}'_{R0}$ and $\mathbf{D}'_{RM}$ are the initial disparity estimate and the locally scaled matted disparity map for the corresponding right-view input image $\mathbf{I}_R$, respectively. $g(\cdot)$ works as a backward-warping operation. In Eq. 5, if the inequalities in both brackets are satisfied at a pixel location, the resulting mask value at that location is 1, otherwise 0. The distillation map for the right view $\mathbf{M}_R$ is obtained by swapping the L and R sub-scripts in Eq. 5. Eq. 5 selects as a source for self-supervisions the pixel depths in $\mathbf{D}'_{LM}$ that (i) generate better backward warped images and (ii) are more consistent with their corresponding right view pixel depths. As can be noted in the detailed view of the hand of the biker in Figure 3-(e), the matted disparity becomes dramatically sharper than the initial disparity estimate but has incorrect values. The mean-scaled matted disparity is both sharper and correctly scaled. As expected, the distillation mask is active on the biker's hand edges, which guides our network to generate pixel-level accurate depth estimates.

3.4. Network Architecture

Our proposed PLADE-Net adopts the simple auto-encoder backbone from [14], with residual blocks in the encoder side and nearest-upscale-based up-convolutions followed by skip-connections in the decoder side. However, we considerably change the learned feature maps' distribution by doubling the extracted features in the shallow convolutional layers (from 32 to 64) and halving the number of feature maps in the bottleneck (from 512 to 256). Our PLADE-Net is depicted in Figure 4 and incorporates our proposed NPE by concatenating (denoted by ⓒ) deep positional features at each encoder stage's input.

Contrary to the previous works [11, 10, 42, 16, 14] that incorporate a single-scale input and multi-scale outputs, our PLADE-Net adopts multi-scale inputs and a single scale output. In our PLADE-Net, low-level features are extracted from a bilinearly downscaled version of the input image $\mathbf{I}_L$ and concatenated into the second encoder stage's input, as depicted to the left-hand side of Figure 4. Our PLADE-Net outputs a single-scale disparity logit volume $\mathbf{D}_L^L$, which can be employed for novel view synthesis and SVDE, as shown to the right-hand side of Fig. 4 and described by (1) and (2).

Our PLADE-Net delivers higher performance with an equal or lower number of parameters than the previous works, with 15M versus 17M of the previous SOTA [14]. It is worth noting that our PLADE-Net achieves the SOTA performance without the need for any advanced layer such as attention, batch/group normalization, sub-pixel convolution, or spatial pyramid pooling. Detailed architecture layer information can be found in the supplementary materials.

3.5. Loss Functions

Following the training strategy in [14], we train our PLADE-Net in two stages. In the first stage, we focus on learning stereoscopic view synthesis, which can be understood as training the top output branch of our PLADE-Net in Figure 4, which generates a synthetic right view. In the second stage, we train our PLADE-Net with an occlusions-free reconstruction loss. Still, more importantly, we incorporate additional loss functions that affect the lower output branch in Figure 4, which generates a near pixel-accurate disparity estimate. The total loss function (l_{s1}) in the first stage of training is a combination of l_1, perceptual [24] (l_p),

Methods	Data	abs rel	sq rel	rmse	rmse$_{log}$	δ^1
Effects of NPE in the 1st stage of training						
w/o PE	K+CS	0.075	0.317	2.990	0.111	0.938
PE	K+CS	0.070	0.292	2.988	0.109	0.939
NPE	K	0.071	0.318	3.236	0.113	0.934
NPE	K+CS	0.070	0.291	2.910	0.107	0.942
Effects of NPE in the 2nd stage of training (a_{dc}=0.01, a_{dm}=0.25)						
FAL-net [14]	K+CS	0.071	0.287	2.905	0.109	0.941
w/o PE	K+CS	0.074	0.298	2.842	0.108	0.942
PE	K+CS	0.067	0.268	2.797	0.104	0.945
NPE	K	0.066	0.274	2.881	0.105	0.944
NPE	K+CS	0.066	0.263	2.726	0.102	0.949
Effects of l_{dc} and l_{dm} in the 2nd stage of training						
a_{dc}=0, a_{dm}=0	K+CS	0.067	0.270	2.777	0.104	0.943
a_{dc}=0.01, a_{dm}=0	K+CS	0.067	0.267	2.775	0.104	0.945
a_{dc}=0, a_{dm}=0.25	K+CS	0.068	0.268	2.741	0.103	0.948

Table 1. Ablation studies of our PLADE-Net on KITTI[9]. Metrics are the lower the better and the higher the better.

and disparity smoothness (l_{ds}) losses, as given by

$$l_{s1} = l_1 + \alpha_p l_p + \alpha_{ds} l_{ds}, \quad (6)$$

where α_p and α_{ds} are empirically set to 0.01 and 0.0004, respectively, to balance their contributions. The total loss l_{s2} for the second training stage adds our novel distilled matting Laplacian loss (l_{dm}), a deep corr-l_1 loss (l_{dc}), and the mirror loss (l_m) in [14] to l_{s1}, and is defined by:

$$l_{s2} = l_{s1} + l_m + \alpha_{dm} l_{dm} + \alpha_{dc} l_{dc}, \quad (7)$$

where $\alpha_{dm} = 0.25$ and $\alpha_{dc} = 0.01$ are empirically set to weight the contributions of the distilled matting Laplacian and the deep corr-l_1 losses, respectively. In the second stage, l_{s2} is computed for both left and right views, giving the actual total loss of $l = (l^L_{s2} + l^R_{s2})/2$. We describe l_{s1} and l_m in detail in our supplemental materials.

Deep Corr-l_1 Loss. Inspired by [44], we explored training our PLADE-Net with a deep corr-l_1 loss, to encourage the generation of depth estimates with structural details similar to the ones in the single-view input. However, we observed marginal performance improvements and depth artifacts, as further shown in Section 4. Nevertheless, we observed an affinity between the deep corr-l_1 loss and our proposed distilled matting Laplacian loss. Our deep corr-l_1 loss (l_{dc}) penalizes the deep-auto-correlation difference between the input image and the predicted depth map. Deep-auto-correlation is obtained by measuring the auto-correlation of the deep features of $\mathbf{I}_L$ or $\mathbf{D}'_L$, extracted by the third max-pool layer of a pre-trained VGG19 [37] for the image classification task, denoted by $\phi(\cdot)$. l_{dc} is then given by:

$$l_{dc} = ||\text{acorr}(\phi(\mathbf{D}'_L), k) - \text{acorr}(\phi(\mathbf{I}_L), k)||_1, \quad (8)$$

where acorr$(\cdot, k)$ is the auto-correlation operator on a $k \times k$ window, empirically set to $k = 3$ in all our experiments.

Distilled Matting Laplacian Loss. We previously detailed our matting Laplacian distillation process in Subsection 3.3. Given the locally scaled matted left disparity map $\mathbf{D}'_{LM}$ and distillation mask $\mathbf{M}_L$, our distilled matting Laplacian loss l_{dm} is simply given by:

$$l_{dm} = (1/\max(\mathbf{D}'_L))||\mathbf{M}_L \odot (\mathbf{D}'_L - \mathbf{D}'_{LM})||_1, \quad (9)$$

where $\max(\mathbf{D}'_L)$ is the maximum disparity value in the scene and normalizes the loss between 0 and 1. By incorporating $\mathbf{M}_L$ into Eq. (9), we can keep the highly detailed matted depths while filtering out the incorrectly labeled pixels commonly present in image matting.

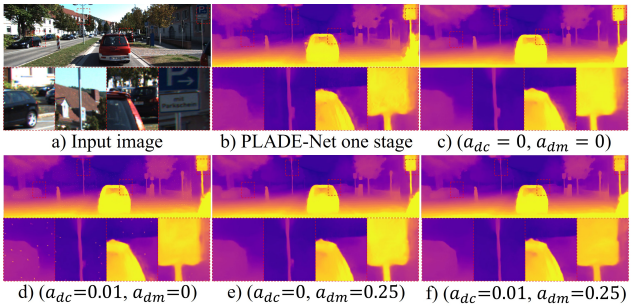


Figure 5. Ablation studies on our distilled matting Laplacian loss.

4. Experiments and Results

4.1. Datasets

KITTI[9]. To compare with a wider spectrum of recent works, we utilize the Eigen train split [5] (K), which is a subset of the KITTI [9] training set, consisting of 22,600 left-right training image pairs captured from a moving car. Following the standard practice, we test our method on the KITTI Eigen test split in its original [5] and improved [39] versions, which contain 697 and 652 images with projected LiDAR ground truths (GT), respectively. The improved Eigen test split contains denser GTs by selectively accumulating LiDAR points from 5 consecutive frames. Performance is measured with the metrics defined in [5] (up to 80m). Additionally, in our experiments, we propose a "naive" stereo input extension of our PLADE-Net, which is trained with a split obtained from the intersection of the KITTI Eigen train set [5] and the KITTI Split [11]. The resulting Stereo-Split excludes scenes from the KITTI Eigen test split [5] and the KITTI2015 training set [32]. The KITTI2015 [32] training set consists of 200 image pairs with CAD-refined LiDAR GT and is the default benchmark to evaluate self-supervised stereo networks.

CityScapes[4]. In most of our ablation studies, we concurrently train the PLADE-Net variations with the CityScapes [4] dataset (following the multi-dataset training procedure in [13, 14]) to ensure that they do not underperform due to the lack of enough data. The CityScapes [4] dataset consists of 24,500 stereo pairs without depth GTs, and similar to KITTI [9], it is captured from a driving perspective. We follow the car hood and border artifacts removal procedures from [11, 13, 14].

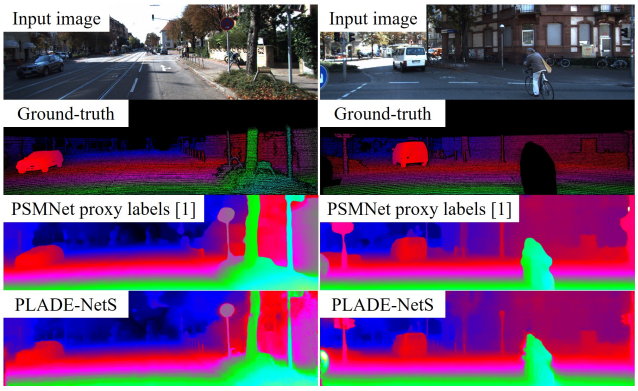


Figure 6. Stereo results on the KITTI2015[32] dataset.

Make3D[36]. To test the generalization power of our PLADE-Net, we evaluate it on the Make3D [36] Test134 dataset, which is made of 134 high-resolution RGB outdoor images with low-resolution depth GTs. We followed the evaluation procedure defined in [11], with the C1 metrics (up to 70m) defined in [29].

4.2. Implementation Details

Following the training procedure in [14], we trained our PLADE-Net for 50 epochs in the first training stage and 10 epochs in the second stage with a batch size of 8 by the Adam[25] optimizer with an initial learning rate of 1×10^{-4} and 5×10^{-5}, respectively. The learning rate was reduced to half at epochs [30, 40, 50] in the first training stage and epochs [5, 10] in the second stage. Data augmentations on-the-fly were incorporated into our network training. For a fair comparison with previous works, we adopted random resizing from 0.75 to 1.5, followed by 192×640 random cropping, random left-right flipping, random gamma, random brightness, and random individual color brightness. Inference was run at full image resolution.

4.2.1 Computing the Matting Laplacian

Computing the closed-form solution to the matting Laplacian [28] is expensive, taking up to 30 (60) seconds in matting a KITTI [9] (CityScapes [4]) depth-RGB sample. For this reason, we first generated a matted disparity complementary dataset instead of matting on-the-fly. We used our PLADE-Net with one stage of training to build such a matted disparity dataset. We procured to apply the corresponding spatial data augmentations (resizes, crops, and flips) to the matted disparity samples during training. Our distillation process re-scales the matted disparity values, thus not requiring to apply scaling factors during data sampling.

4.3. Ablation Studies

Table 1 shows our ablation studies on the improved KITTI Eigen test split [39]. We first ablate the effects of our

Method	abs rel	sq rel	rmse	$\text{rmse}_{\log}$	δ^1	δ^2	δ^3
Monodepth [11]	0.068	0.835	4.392	0.146	0.942	0.978	0.989
Lai *et al.* [26]	0.062	0.747	4.113	0.146	0.948	0.979	0.990
UnOS[41]	0.060	0.833	4.187	0.135	0.955	0.981	0.990
UnOS[41] (SV)	0.049	0.515	3.404	0.121	0.965	0.984	0.992
Aleotti *et al.* [1]	-	-	3.764	0.115	**0.974**	0.988	0.993
PLADE-NetS	0.053	0.323	2.758	0.100	0.965	0.989	0.995
PLADE-NetS	**0.050**	**0.300**	**2.723**	**0.096**	0.967	**0.990**	**0.996**

Table 2. Comparison of existing self-supervised SDE methods on the KITTI2015 [32] training set. SV: Training from stereo videos. **Best** and second-best metrics. Results capped to 80m.

Method	Sup	Data	abs rel	sq rel	rmse
Liu *et al.* [29]	D	M3D	0.475	6.562	10.05
Laina *et al.* [27]	D	M3D	**0.204**	**1.840**	**5.683**
Monodepth2 [10]	V	K	0.322	3.589	7.417
Wang *et al.* [40]	S	K	0.387	4.720	8.090
Glez. and Kim [12]	S	K	0.323	4.021	7.507
Zhou *et al.* [45]	V	K	0.318	2.288	6.669
FAL-net [14] (PP)	S	K	0.284	2.803	6.643
FAL-net [14] (PP)	S	K+CS	0.254	2.140	6.139
PLADE-Net	S	K	0.276	2.635	6.546
PLADE-Net (PP)	S	K	0.265	2.469	6.373
PLADE-Net	S	K+CS	0.257	2.146	6.097
PLADE-Net (PP)	S	K+CS	**0.253**	**2.100**	**6.031**

Table 3. Results on Make3D [36]. All self-supervised methods benefit from median scaling. M3D: Training on the Make3D[36].

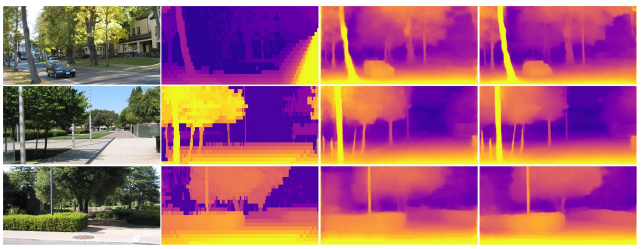


Figure 7. Qualitative comparisons on the Make3D dataset [36].

NPE in the PLADE-Net for the first stage of training. As it can be noted, the positional encoding (PE) that simply concatenates the pixel location (x, y) values directly into the encoder stage can even yield slight performance improvements in most metrics in comparison with the cases without it (denoted as "w/o PE"). However, our PLADE-Net shows substantial performance improvements in all metrics by incorporating our *neural* positional encoding (NPE).

The ablation studies on the effects of our NPE in the second stage of training are shown in the second section of Table 1. Interestingly, our PLADE-Net w/o PE gets stuck in bad local minima, with marginal performance improvements in the second training stage. In contrast, our PLADE-Net with simple PE outperforms all previous SOTA methods in terms of δ^1 accuracy. On the other hand, our PLADE-Net with NPE exhibits the best performance by consider-

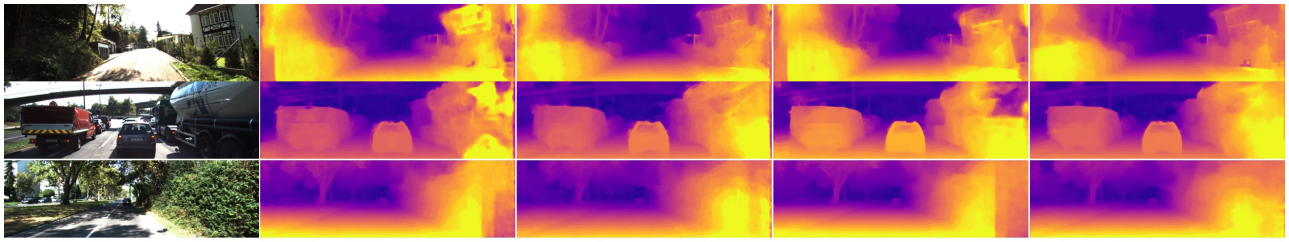


Figure 8. Qualitative ablation studies on our proposed neural positional encoding (NPE) in our PLADE-Net.

able margins. Our PLADE-Net with NPE trained only with the KITTI Eigen train split [5] shows the efficacy of robust learning regardless of the training data size. The effects of our NPE in both training stages (TS1 and TS2) are depicted in Figure 8. Our PLADE-Net without NPE struggles to estimate depths for objects close to the image borders, yielding depth artifacts regardless of the training stage.

The third section of Table 1 and Figure 5 show respectively the quantitative and qualitative ablation studies for our proposed loss functions. As can be noted, our distilled matting Laplacian loss is the main contributor to achieving such high-performance. Our PLADE-Net trained with distilled matting Laplacian loss only ($a_{dc} = 0$, $a_{dm} = 0.25$) achieves an δ^1 accuracy of 94.8% while our PLADE-net with deep corr-l_1 loss only ($a_{dc} = 0.01$, $a_{dm} = 0$) obtains the lower accuracy of 94.5%. In addition, training with only deep corr-l_1 loss induces depth artifacts seen as bright spots in Figure 5-(d). Our PLADE-Net without our proposed loss functions ($a_{dc} = 0$, $a_{dm} = 0$) shows the lowest performance with the most blur depth estimates as shown in Figure 5-(c). Interestingly, our PLADE-Net with (a_{dc}=0, a_{dm}=0) still shows considerably better metrics than the previous SOTA FAL-net [14], demonstrating the effectiveness of our NPE and design choices of multi-scale inputs and more learned features on the shallow feature extractors, suggesting that SVDE benefits from richer low-level features.

4.4. Resutls

Results on KITTI. Table 4 and Figure 9 present quantitative and qualitative comparisons among the previous methods and our PLADE-Net on the KITTI Eigen test split [5]. Our PLADE-Net clearly outperforms all previous self-supervised methods in most metrics on the original Eigen test split [5], and in all the metrics on the improved test split [39]. Our PLADE-Net shows sharper and pixel-level accurate depth estimates in complex and cluttered image regions, as shown in every zoom-box of Figure 9. Quantitatively, our PLADE-Net without any post-processing (PP), even trained only on the KITTI (K) dataset, outperforms the previous methods that were trained on KITTI + CityScapes (K+CS). Following the PP step in [14], our method achieves even higher accuracies and lower error metrics.

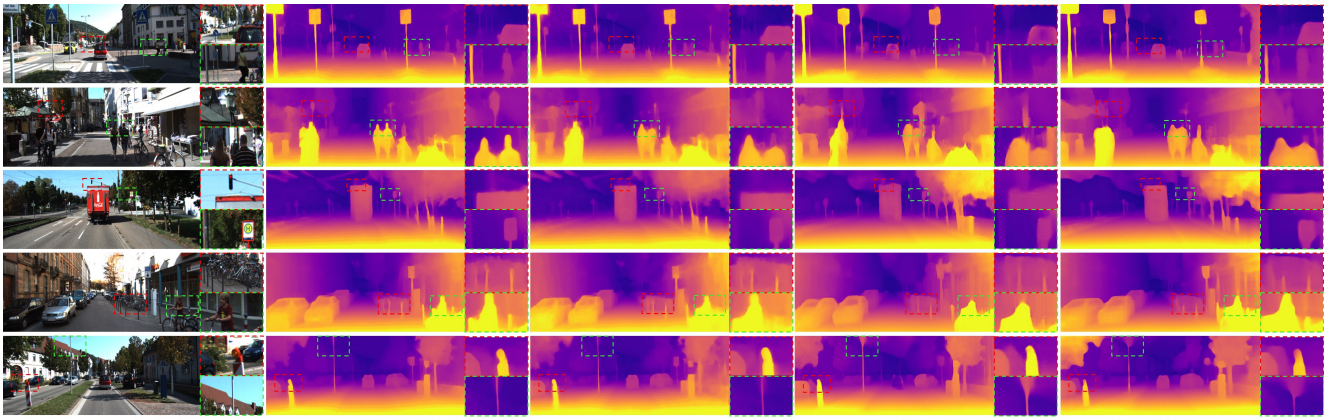
Results on KITTI (stereo) To further evaluate the effectiveness of our PLADE-Net, we define an stereo input variant, the PLADE-NetS, which is evaluated and compared with the SOTA methods on the KITTI2015 dataset. Our stereo variant is a clone of the PLADE-net, with the difference that the PLADE-NetS "naively" incorporates the right-view image information in a second encoder, whose bottleneck features are concatenated to the left view bottleneck features. In our PLADE-NetS we do not incorporate any advanced stereo matching layers such as 1D-Correlation or 3D convolutions, and still, our network manages to outperform the most recent self-supervised SOTA methods [1, 41] in most metrics by a considerable margin, as indicated in Table 2. Figure 6 shows that our network with stereo inputs keeps generating very sharp and pixel-level accurate depth estimates with clear object boundaries.

Results on Make3D. Table 3 compares our PLADE-Net against the SOTA self-supervised methods on Make3D [36]. Our approach generalizes the best among the self-supervised methods under comparison and is very close to the fully-supervised method of Laina *et al*. [27]. It is clear in Figure 7 that our PLADE-Net generates sharper depth estimates on the previously unseen Make3D dataset [36] in comparison with the recent FAL-net [14] SOTA.

5. Conclusions

We showed that our PLADE-Net with neural positional encoding (NPE) could generalize better than the conventional CNN approaches. NPE allows our PLADE-Net to learn location-specific features, which aid in predicting consistent disparities in all image regions. Furthermore, our proposed distilled matting Laplacian loss provides strong self-supervision signals to learn sharp and pixel-level accurate depth estimation. Our PLADE-Net outperforms all previous self-, semi-, and fully-supervised methods on the challenging KITTI dataset with remarkable accuracy levels and exhibits superior generalization capacities on the Make3D and CityScapes datasets.

Acknowledgment. This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2017-0-00419, Intelligent High Realistic Visual Processing for Smart Broadcasting Media).



Input image | Our PLADE-Net | FAL-net [14] | PackNet [16] | DepthHints [42]

Figure 9. Qualitative comparisons on the KITTI Eigen test split [5]. Our PLADE-Net consistently estimates much more detailed depths.

Ref	Methods	PP	Sup	Data	#Par	abs rel	sq rel	rmse	$\text{rmse}_{\log}$	δ^1	δ^2	δ^3
				Original Eigen Test Split [5]								
[19]	Gur *et al.*		DoF	K	-	0.110	0.666	4.186	0.168	0.880	0.966	**0.988**
[31]	Luo *et al.*		D+S	K	-	0.094	0.626	4.252	0.177	0.891	0.965	0.984
[10]	Monodepth2		V	K	14	0.115	0.882	4.701	0.190	0.879	0.961	0.982
[16]	PackNet		V	K	120	0.107	0.802	4.538	0.186	0.889	0.962	0.981
[15]	Gordon *et al.*		V	K+CS	-	0.124	0.930	5.120	0.206	0.851	0.950	0.978
[16]	PackNet		V	CS→K	120	0.104	0.758	4.386	0.182	0.895	0.964	0.982
[17]	Guizilini *et al.*		V+Se	CS→K	140	0.100	0.761	4.270	0.175	0.902	0.965	0.982
[33]	SuperDepth		S	K	-	0.112	0.875	4.958	0.207	0.852	0.947	0.977
[38]	Tosi *et al.*	✔	S_{SGM}	K	42	0.111	0.867	4.714	0.199	0.864	0.954	0.979
[34]	Refine&Distill		S	K	-	0.098	0.831	4.656	0.202	0.882	0.948	0.973
[42]	DepthHints	✔	S_{SGM}	K	35	0.096	0.710	4.393	0.185	0.890	0.962	0.981
[38]	Tosi *et al.*	✔	S_{SGM}	CS→K	42	0.096	0.673	4.351	0.184	0.890	0.961	0.981
[47]	Edge-of-depth	✔	S+Se	K	-	0.091	0.646	4.244	0.177	0.898	0.966	0.983
[14]	FAL-net	✔	S	K+CS	17	0.088	**0.547**	4.004	0.175	0.898	0.966	0.984
[14]	FAL-net	✔	S	K	17	0.094	0.597	4.005	0.173	0.900	0.967	0.985
our	PLADE-Net		S	K	15	0.092	0.626	4.046	0.175	0.896	0.965	0.984
our	PLADE-Net	✔	S	K	15	0.089	0.590	4.008	0.172	0.900	0.967	0.985
our	PLADE-Net		S	K+CS	15	0.090	0.577	3.880	0.170	0.903	0.968	0.985
our	PLADE-Net	✔	S	K+CS	15	**0.087**	0.550	**3.837**	**0.167**	**0.908**	**0.970**	0.985
				Improved Eigen Test Split [39]								
[7]	DORN		D	K	51	0.072	0.307	2.727	0.120	0.932	0.984	0.995
[10]	Monodepth2		V	K	14	0.092	0.536	3.749	0.135	0.916	0.984	0.995
[16]	PackNet (LR)		V	K	120	0.078	0.420	3.485	0.121	0.931	0.986	0.996
[16]	PackNet		V	CS→K	120	0.071	0.359	3.153	0.109	0.944	0.990	0.997
[10]	Monodepth2		S	K	14	0.084	0.503	3.646	0.133	0.920	0.982	0.994
[42]	DepthHints	✔	S_{SGM}	K	35	0.074	0.364	3.202	0.114	0.936	0.989	0.997
[14]	FAL-net	✔	S	K	17	0.071	0.281	2.912	0.108	0.943	0.991	0.998
[14]	FAL-net	✔	S	K+CS	17	0.068	0.276	2.906	0.106	0.944	0.991	0.998
our	PLADE-Net		S	K	15	0.066	0.274	2.881	0.105	0.944	0.992	0.998
our	PLADE-Net	✔	S	K	15	0.066	0.272	2.918	0.104	0.945	0.992	0.998
our	PLADE-Net		S	K+CS	15	0.066	0.263	2.726	0.102	0.949	**0.992**	**0.998**
our	PLADE-Net	✔	S	K+CS	15	**0.065**	**0.253**	**2.710**	**0.100**	**0.950**	**0.992**	**0.998**
our	PLADE-NetS		S	K	15	0.036	0.094	1.791	0.060	0.988	0.998	0.999
our	PLADE-NetS	✔	S	K	15	0.035	0.091	1.748	0.058	0.989	0.998	1.000

Table 4. Evaluations on the KITTI Eigen test split [5]. Models are trained on the KITTI Eigen[5] train-split (K) and CityScapes[4] (CS). CS→K indicates CS pre-training. K+CS indicates concurrent K and CS training. DoF and D denote depth-of-field and depth supervision. S, S_{SGM}, S+Se, V, V+Se indicate stereo, stereo + SGM, stereo + semantics, video, and video + semantics self-supervision. V methods benefit from median-scaling. **Best** and second-best metrics. Methods that use post-processing (PP) are checked ✔. Results capped to 80m.

References

[1] Filippo Aleotti, Fabio Tosi, Li Zhang, Matteo Poggi, and Stefano Mattoccia. Reversing the cycle: self-supervised deep stereo through enhanced monocular distillation. In *16th European Conference on Computer Vision (ECCV)*. Springer, 2020. 1, 2, 6, 7

[2] Martin Čadík, Daniel Sỳkora, and Sungkil Lee. Automated outdoor depth-map generation and alignment. *Computers & Graphics*, 74:109–118, 2018. 4

[3] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018. 2

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016. 5, 6, 8

[5] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems*, pages 2366–2374, 2014. 5, 7, 8

[6] Jose M. Facil, Benjamin Ummenhofer, Huizhong Zhou, Luis Montesano, Thomas Brox, and Javier Civera. Camconvs: Camera-aware multi-scale convolutions for single-view depth. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[7] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2002–2011, 2018. 8

[8] Ravi Garg, Vijay Kumar BG, Gustavo Carneiro, and Ian Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *European Conference on Computer Vision*, pages 740–756. Springer, 2016. 2

[9] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012. 2, 5, 6

[10] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3828–3838, 2019. 2, 4, 6, 8

[11] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 270–279, 2017. 2, 4, 5, 6

[12] Juan Luis Gonzalez and M. Kim. A novel monocular disparity estimation network with domain transformation and ambiguity learning. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 474–478, Sep. 2019. 6

[13] Juan Luis Gonzalez and M. Kim. Deep 3d pan via local adaptive "t-shaped" convolutions with global and local adaptive dilations. In *International Conference on Learning Representations*, 2020. 5

[14] Juan Luis GonzalezBello and Munchurl Kim. Forget about the lidar: Self-supervised depth estimators with med probability volumes. In *Advances in Neural Information Processing Systems*, volume 33, pages 12626–12637. Curran Associates, Inc., 2020. 1, 2, 3, 4, 5, 6, 7, 8

[15] Ariel Gordon, Hanhan Li, Rico Jonschkowski, and Anelia Angelova. Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8977–8986, 2019. 2, 8

[16] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2, 4, 8

[17] Vitor Guizilini, Rui Hou, Jie Li, Rares Ambrus, and Adrien Gaidon. Semantically-guided representation learning for self-supervised monocular depth. In *International Conference on Learning Representations*, 2020. 1, 2, 8

[18] Xiaoyang Guo, Hongsheng Li, Shuai Yi, Jimmy Ren, and Xiaogang Wang. Learning monocular depth by distilling cross-domain stereo networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 484–500, 2018. 2

[19] Shir Gur and Lior Wolf. Single image depth estimation trained via depth from defocus cues. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7683–7692, 2019. 8

[20] Heiko 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)*, volume 2, pages 807–814. IEEE, 2005. 2

[21] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2):328–341, 2007. 2

[22] Md Amirul Islam*, Sen Jia*, and Neil D. B. Bruce. How much position information do convolutional neural networks encode? In *International Conference on Learning Representations*, 2020. 3

[23] Jinbeum Jang, Sangwoo Park, Jieun Jo, and Joonki Paik. Depth map generation using a single image sensor with phase masks. *Optics express*, 24(12):12868–12878, 2016. 4

[24] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 3, 4

[25] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 6

[26] Hsueh-Ying Lai, Yi-Hsuan Tsai, and Wei-Chen Chiu. Bridging stereo matching and optical flow via spatiotemporal correspondence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1890–1899, 2019. 2, 6

[27] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth international conference on 3D vision (3DV)*, pages 239–248. IEEE, 2016. 6, 7

[28] Anat Levin, Dani Lischinski, and Yair Weiss. A closed-form solution to natural image matting. *IEEE transactions on pattern analysis and machine intelligence*, 30(2):228–242, 2007. 1, 3, 4, 6

[29] Miaomiao Liu, Mathieu Salzmann, and Xuming He. Discrete-continuous depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–723, 2014. 6

[30] Shaojun Liu, Fei Zhou, and Qingmin Liao. Defocus map estimation from a single image based on two-parameter defocus model. *IEEE Transactions on Image Processing*, 25(12):5943–5956, 2016. 4

[31] Yue Luo, Jimmy Ren, Mude Lin, Jiahao Pang, Wenxiu Sun, Hongsheng Li, and Liang Lin. Single view stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 155–163, 2018. 8

[32] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 5, 6

[33] Sudeep Pillai, Rareş Ambruş, and Adrien Gaidon. Superdepth: Self-supervised, super-resolved monocular depth estimation. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 9250–9256. IEEE, 2019. 2, 8

[34] Andrea Pilzer, Stephane Lathuiliere, Nicu Sebe, and Elisa Ricci. Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9768–9777, 2019. 2, 8

[35] Matteo Poggi, Fabio Tosi, and Stefano Mattoccia. Learning monocular depth estimation with unsupervised trinocular assumptions. In *2018 International Conference on 3D Vision (3DV)*, pages 324–333. IEEE, 2018. 2

[36] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3d: Learning 3d scene structure from a single still image. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):824–840, 2008. 6, 7

[37] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[38] Fabio Tosi, Filippo Aleotti, Matteo Poggi, and Stefano Mattoccia. Learning monocular depth estimation infusing traditional stereo knowledge. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9799–9809, 2019. 2, 8

[39] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *2017 International Conference on 3D Vision (3DV)*, pages 11–20. IEEE, 2017. 5, 6, 7, 8

[40] Chaoyang Wang, José Miguel Buenaposada, Rui Zhu, and Simon Lucey. Learning depth from monocular videos using direct methods. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2022–2030, 2018. 6

[41] Yang Wang, Peng Wang, Zhenheng Yang, Chenxu Luo, Yi Yang, and Wei Xu. Unos: Unified unsupervised optical-flow and stereo-depth estimation by watching videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2, 6, 7

[42] Jamie Watson, Michael Firman, Gabriel J Brostow, and Daniyar Turmukhambetov. Self-supervised monocular depth hints. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2162–2171, 2019. 2, 4, 8

[43] Zhenheng Yang, Peng Wang, Yang Wang, Wei Xu, and Ram Nevatia. Every pixel counts: Unsupervised geometry learning with holistic 3d motion understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018. 2

[44] Yu Zhang, Dongqing Zou, Jimmy S. Ren, Zhe Jiang, and Xiaohao Chen. Structure-preserving stereoscopic view synthesis with multi-scale adversarial correlation matching. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 5

[45] Junsheng Zhou, Yuwang Wang, Kaihuai Qin, and Wenjun Zeng. Unsupervised high-resolution depth learning from videos with dual networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6872–6881, 2019. 6

[46] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1851–1858, 2017. 2

[47] Shengjie Zhu, Garrick Brazil, and Xiaoming Liu. The edge of depth: Explicit constraints between segmentation and depth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 8

[48] C Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. High-quality video view interpolation using a layered representation. *ACM transactions on graphics (TOG)*, 23(3):600–608, 2004. 4