NDDepth: Normal-Distance Assisted Monocular Depth Estimation

Shuwei Shao[1,2] Zhongcai Pei[2,1] Weihai Chen[2,1*] Xingming Wu[2,1] Zhengguo Li[3]
[1]School of Automation Science and Electrical Engineering, Beihang University, Beijing, China.
[2]Hangzhou Innovation Institute, Beihang University, Hangzhou, Zhejiang, China
[3] SRO department, Institute for Infocomm Research, A*STAR, Singapore
{swshao, peizc, whchen}@buaa.edu.cn ezgli@i2r.a-star.edu.sg

Abstract

Monocular depth estimation has drawn widespread attention from the vision community due to its broad applications. In this paper, we propose a novel physics (geometry)-driven deep learning framework for monocular depth estimation by assuming that 3D scenes are constituted by piece-wise planes. Particularly, we introduce a new normal-distance head that outputs pixel-level surface normal and plane-to-origin distance for deriving depth at each position. Meanwhile, the normal and distance are regularized by a developed plane-aware consistency constraint. We further integrate an additional depth head to improve the robustness of the proposed framework. To fully exploit the strengths of these two heads, we develop an effective contrastive iterative refinement module that refines depth in a complementary manner according to the depth uncertainty. Extensive experiments indicate that the proposed method exceeds previous state-of-the-art competitors on the NYU-Depth-v2, KITTI and SUN RGB-D datasets. Notably, it ***ranks 1st*** *among all submissions on the KITTI depth prediction online benchmark at the submission time. The source code is available at* `https://github.com/ShuweiShao/NDDepth`.

1. Introduction

Monocular depth estimation plays a more and more important role in computer vision, with applications ranging from robotics [41, 19], scene understanding [17] to augmented reality [25]. The task aims to estimate the depth map from a single RGB image. Considering that a 2D image can be projected from infinite number of 3D scenes, such a task is indeed ill-posed and inherently ambiguous. Hence, solving it reliably presents a formidable challenge for traditional methods [31, 32] due to their inherent limitations, typically involving low-dimensional and sparse distances or known and fixed objects.

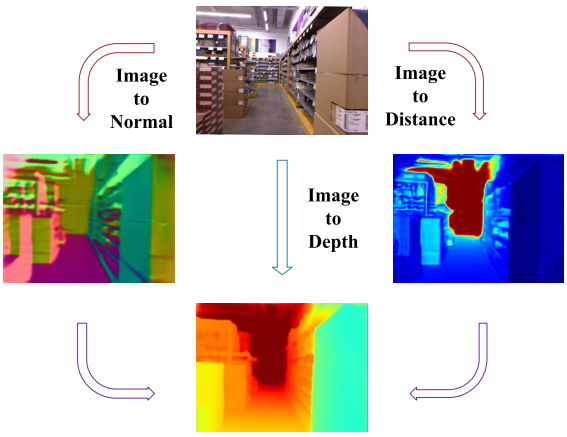

Figure 1. **A brief illustration of the relationship among image, surface normal, plane-to-origin distance and depth map**.

Recently, much progress has been made in this field benefiting from the explosion of deep learning [11, 13, 23, 2, 46]. Most efforts focus on designing increasingly complicated and powerful networks, which renders the task a hard fitting problem without the help of additional guidance. We argue that real-world 3D scenes usually have a high degree of regularity and proper scene priors should be incorporated into the framework to improve the nature of the solution. Planes are a common representation for modeling geometric prior knowledge of 3D scenes [3, 5, 27]. A recent work by Patil *et al.* [34] introduced a piece-wise planarity prior and used the offset vector field to borrow information from co-planar pixels. However, there is no direct constraint imposed on the offset vector field to help it learn about planar regions in their method. In addition, the planarity prior tends to fail in high-curvature regions, for example, bushes, trees, and other clutter, inevitably deteriorating the depth accuracy.

*Corresponding author

In this paper, we propose a novel physics or geometry-driven deep learning framework to estimate a high-quality depth map from a single RGB image, by assuming that 3D scenes are constituted with piece-wise planes. To be more specific, we parametrize the plane representation via surface normal and plane-to-origin distance (the distance from the associated plane to the origin, i.e., camera center in our case), which are predicted by our introduced normal-distance head. Furthermore, a plane-aware consistency constraint is developed to encourage the normal and distance to be piece-wise constant. To acquire planar regions, we adopt the Felzenszwalb segmentation algorithm [12] for online plane detection using the geometric dissimilarity calculated from normal and distance. The detected planes could be noisy in early epochs because of inaccurate normal and distance predictions. Fortunately, they will gradually improve as the normal and distance quality improves, and in turn feedback to obtain better normal and distance predictions, which are afterwards converted to improved depth maps. It should be noted that the converted depth maps are prone to make severe errors in high-curvature regions due to the invalidity of the planarity assumption. To alleviate such failure cases, we integrate a second depth head designed in accordance with the regular paradigms, *e.g.*, [46]. In order to fully exploit the strengths of these two heads, uncertainty maps describing the depth uncertainty are additionally estimated. The depth and uncertainty maps are fed into a contrastive iterative refinement module for depth refinement in a complementary manner. The full pipeline is shown in Figure 2.

In the experiment, the proposed method is verified comprehensively on 3 standard datasets, including NYU-Depth-v2 [39], KITTI [15] and SUN RGB-D [40]. Comparisons to leading approaches show that our method exceeds previous state-of-the-art competitors on the NYU-Depth-v2 and KITTI. In a challenging zero-shot setup, our method surpasses prior methods on the SUN RGB-D. Detailed ablation study is performed to demonstrate the merits of our developed components. In addition, we present qualitative comparisons with leading approaches to evidence the high quality of our depth and point cloud predictions.

To summarize, the contributions of this work are listed as follows:

- We propose a new physics-driven deep learning framework for monocular depth estimation, which contains a normal-distance head and a depth head, built with asymmetric paradigms for depth acquisition.
- We introduce a plane-aware consistency constraint to regularize surface normal and plane-to-origin distance, and a contrastive iterative refinement module to refine depth in a complementary manner.
- The proposed method surpasses previous state-of-the-art approaches on the NYU-Depth-v2, KITTI and SUN RGB-D datasets, and **ranks 1st** on the KITTI depth prediction online benchmark at the submission time.

2. Related work

2.1. Monocular Depth Estimation

Monocular depth estimation attempts to predict the depth map from a single RGB image. As one of the first learning-based studies, Saxena *et al.* [37] took into account both local and global features, and used a Markov Random Field to regress depth. Eigen *et al.* [11] pioneered the usage of convolutional neural network (CNN) in depth estimation, leveraging multi-scale networks for depth inference. Since then, many studies have sprung up to focus on this setting. Laina *et al.* [22] introduced a fully convolutional network based on residual learning and a reverse Huber loss for optimization. Cao *et al.* [4] and Fu *et al.* [13] reformulated the depth regression problem as classification to predict easier depth ranges than the exact depth values. Lee *et al.* [23] designed several multi-scale guidance layers to establish the connection between intermediate layer features and the final depth map. Bhat *et al.* [2] revisited the ordinal regression network [13] and proposed to derive adaptive bins from the image content. Yang *et al.* [43] developed one of the first attempts of leveraging the Vision Transformer (ViT) [9] to capture the long-distance correlation in depth estimation. In order to acquire the long-distance correlation in the decoder, Yuan *et al.* [46] introduced neural window full-connected Conditional Random Fields (CRFs). Shao *et al.* [38] adopted cross-distillation to exploit the strengths of Transformer and CNN. Nevertheless, most of these studies are data-driven approaches while the proposed physics-driven deep learning framework leverages the geometry of real-world 3D scenes.

2.2. Geometric Constraints for Depth

Traditional methods for multi-view stereo [14] and 3D reconstruction [5, 3] leveraged the planarity prior to enable faster optimization and address the challenges induced by poorly textured surfaces. Recently, Qi *et al.* [35] designed a geometric network to infer geometrically consistent surface normal and depth map. Long *et al.* [30] developed an adaptive surface normal to regularize the depth map. Huynh *et al.* [18] and Yin *et al.* [44] introduced non-local coplanarity constraints either by the depth-attention module or the virtual normal. Kusupati *et al.* [21] enforced a consistency between spatial depth gradients calculated from estimated depth and normal. Patil *et al.* [34] proposed a piece-wise planarity prior and utilized the offset vector field to sample information from co-planar pixels. Unfortunately, the offset vector field is not given any direct constraints to aid in its comprehension of planar regions. Moreover, the pla-

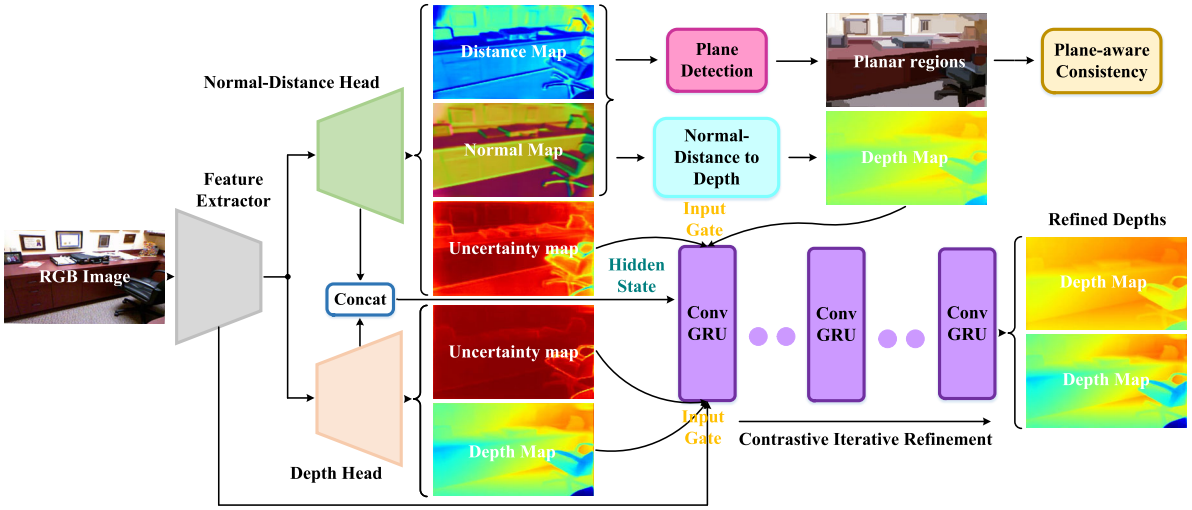

Figure 2. **Overview of the whole framework**. Note that the contrastive iterative refinement is performed at 1/4 resolution and the refined depth maps are upsampled to the full resolution using bilinear interpolation. For simplicity, we omit this detail in the flowchart.

narity prior is not always valid, leading to erroneous depth estimates in high-curvature regions. By contrast, the planar constraint in our method is explicitly enforced inside each planar region, which avoids the interference of pixels from other planes. It is also worth noting that the depth head in our framework allows such failure cases in [34] to be alleviated.

3. Methodology

Real-world 3D scenes usually have a high degree of regularity. It is therefore reasonable to assume that 3D scenes are constituted with piece-wise planes, and the surface normal and plane-to-origin distance are piece-wise constant. In this section, we introduce three main parts of the proposed framework in detail, including depth from normal-distance constraint, plane-aware consistency and contrastive iterative refinement.

3.1. Depth from Normal-Distance Constraint

Let $\mathbf{P} = [X, Y, Z]^{\mathrm{T}}$ be a 3D point and $\mathbf{p} = [u, v]^{\mathrm{T}}$ be its projected 2D point on the image plane, corresponding to a planar region of the 3D scenes. The surface normal $\mathbf{N}(\mathbf{p})$ is defined as the vector starting from $\mathbf{P}$ and perpendicular to the associated plane. The plane-to-origin distance $\mathcal{D}(\mathbf{p})$ is defined as the distance between the associated plane and the origin (camera center in our case). The normal-distance constraint is formulated as

$$\mathbf{N}(\mathbf{p})\,\mathbf{P} = \mathcal{D}(\mathbf{p}). \tag{1}$$

According to the imaging principles of a pinhole camera, the projection from the 3D point $\mathbf{P}$ to the 2D point $\mathbf{p}$ is mathematically described as

$$\mathbf{D}(\mathbf{p})\,\widetilde{\mathbf{p}} = \mathbf{K}\mathbf{P} \tag{2}$$

where $\mathbf{D}(\mathbf{p})$ is the depth at $\mathbf{p}$, $\widetilde{\mathbf{p}}$ denotes the homogeneous coordinate of $\mathbf{p}$, $\mathbf{K}$ denotes the intrinsic matrix.

By substituting Eq. 2 into Eq. 1, we can acquire the depth via

$$\mathbf{D}(\mathbf{p}) = \frac{\mathcal{D}(\mathbf{p})}{\mathbf{N}(\mathbf{p})\,\mathbf{K}^{-1}\widetilde{\mathbf{p}}}. \tag{3}$$

We design a normal-distance head to predict normal and distance, and then applies Eq. 3 to acquire a depth prediction. Compared with the direct prediction of depth, the intermediate normal-distance representation enjoys the benefit of piece-wise constant (not suitable for depth), which enables the plane-aware consistency constraint to establish interactions among pixels and thus results in improved accuracy of depth estimate.

3.2. Plane-aware Consistency

Planar region detection. In order to enforce the plane-aware consistency constraint, we need to detect the planar regions correctly. Following previous works [7, 8, 45, 26], we adopt the Felzenszwalb segmentation algorithm [12] in our approach.

Let $\mathbf{q}$ be an adjacent pixel of $\mathbf{p}$. We define the normal dissimilarity between them as

$$dis_{\mathbf{N}}(\mathbf{p}, \mathbf{q}) = \|\mathbf{N}(\mathbf{q}) - \mathbf{N}(\mathbf{p})\|, \tag{4}$$

where $\|\cdot\|$ denotes the Euclidean distance. Suppose $dis_{\mathbf{N}}^{\max}$ and $dis_{\mathbf{N}}^{\min}$ are the maximum and minimum dissimilarities among all adjacent pixels, which we use to normalize the dissimilarity via

$$\overline{dis}_{\mathbf{N}} = \frac{dis_{\mathbf{N}}(\mathbf{p}, \mathbf{q}) - dis_{\mathbf{N}}^{\min}}{dis_{\mathbf{N}}^{\max} - dis_{\mathbf{N}}^{\min}}. \tag{5}$$

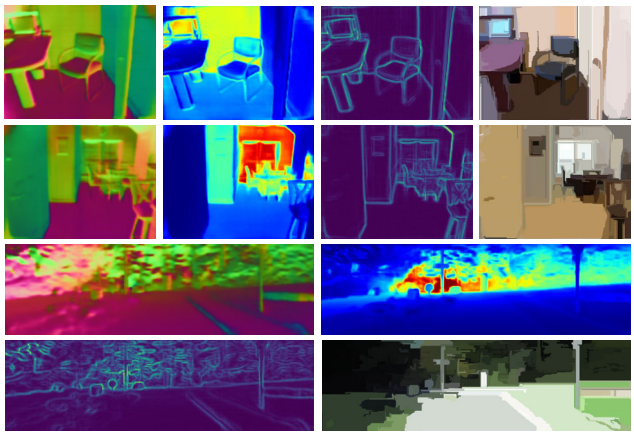

Figure 3. **Visualization of surface normal, plane-to-origin distance, geometric dissimilarity and extracted planar regions.**

The distance dissimilarity is computed as

$$dis_{\mathcal{D}}\left(\mathbf{p},\mathbf{q}\right)=\left|\mathcal{D}(\mathbf{q})-\mathcal{D}(\mathbf{p})\right|. \tag{6}$$

The geometric dissimilarity is a combination of normalized normal and distance dissimilarities

$$\overline{dis}_g=\overline{dis}_{\mathbf{N}}+\overline{dis}_{\mathcal{D}}. \tag{7}$$

Based on the geometric dissimilarity, we use the Felzenszwalb segmentation algorithm to perform online plane detection. Intuitively, the planar regions, for example, floors and walls in indoor scenarios and roads in outdoor scenarios, are more likely to lie within a larger area, and we only select regions larger than 200 pixels. In Figure 3, we show some qualitative results of normal, distance, geometric dissimilarity and detected planes.

Plane-aware consistency loss. After detection of planar regions, we encourage plane-aware consistency by penalizing the first-order gradients of normal and distance within planar regions $\mathcal{M}\left(\mathbf{p}\right)$

$$\mathcal{L}_{pc}=\sum_{\mathbf{p}}\mathcal{M}\left(\mathbf{p}\right)\left|\nabla\mathbf{N}\left(\mathbf{p}\right)\right|+\sum_{\mathbf{p}}\mathcal{M}\left(\mathbf{p}\right)\left|\nabla\mathcal{D}\left(\mathbf{p}\right)\right|. \tag{8}$$

While the initial normal and distance predictions may not be precise, they will gradually improve as the training epochs, leading to a better segmentation and vice versa.

The depth maps converted from normals and distances are not always working, and large errors are prone to occur in high-curvature regions. We make an observation that depth maps derived from regular paradigms have smaller errors in these regions. Hence, the work in [46] is leveraged to define a second head of our framework. To fully exploit the strengths of these two heads, we model the depth uncertainty and develop a contrastive iterative refinement module to refine depth maps in a complementary manner.

3.3. Contrastive Iterative Refinement

Uncertainty modeling. The areas with large errors need to be identified before performing the depth refinement. We model the depth uncertainty with a function inspired by the probability density function of Laplace distribution as

$$\mathbf{U}^{gt}\left(\mathbf{p}\right)=1-\exp\left(-\frac{\left|\mathbf{D}\left(\mathbf{p}\right)-\mathbf{D}^{gt}\left(\mathbf{p}\right)\right|}{b}\right), \tag{9}$$

where $\mathbf{D}^{gt}\left(\mathbf{p}\right)$ denotes the ground-truth depth map, b is a coefficient that controls the error tolerance. Since $\mathbf{D}^{gt}\left(\mathbf{p}\right)$ is not available at test time, uncertainty maps are additionally estimated to approximate $\mathbf{U}_1^{gt}\left(\mathbf{p}\right)$ and $\mathbf{U}_2^{gt}\left(\mathbf{p}\right)$, which is supervised by

$$\mathcal{L}_{\mathbf{U}}=\sum_{\mathbf{p}}\left|\mathbf{U}_1\left(\mathbf{p}\right)-\mathbf{U}_1^{gt}\left(\mathbf{p}\right)\right|+\sum_{\mathbf{p}}\left|\mathbf{U}_2\left(\mathbf{p}\right)-\mathbf{U}_2^{gt}\left(\mathbf{p}\right)\right|, \tag{10}$$

where $\mathbf{U}_1\left(\mathbf{p}\right)$ and $\mathbf{U}_2\left(\mathbf{p}\right)$ denote the uncertainty predictions from normal-distance and depth heads, respectively.

Aside from the uncertainty map, we calculate the absolute difference between $\mathbf{D}_1\left(\mathbf{p}\right)$ and $\mathbf{D}_2\left(\mathbf{p}\right)$ to indicate the complementary regions in these two depth maps, i.e., complementary map,

$$dif_{\mathbf{D}}\left(\mathbf{p}\right)=\left|\mathbf{D}_1\left(\mathbf{p}\right)-\mathbf{D}_2\left(\mathbf{p}\right)\right|, \tag{11}$$

where $\mathbf{D}_1\left(\mathbf{p}\right)$ and $\mathbf{D}_2\left(\mathbf{p}\right)$ denote the depth predictions from normal-distance and depth heads, respectively.

Contrastive iterative update. Grounded on the uncertainty map and complementary map, we refine depth maps iteratively. At each iteration, we update $\mathbf{D}_1\left(\mathbf{p}\right)$ and $\mathbf{D}_2\left(\mathbf{p}\right)$ as

$$\mathbf{D}_1^{t+1}\leftarrow\mathbf{D}_1^t+\Delta\mathbf{D}_1^t,\mathbf{D}_2^{t+1}\leftarrow\mathbf{D}_2^t+\Delta\mathbf{D}_2^t, \tag{12}$$

where $\Delta\mathbf{D}_1^t$ and $\Delta\mathbf{D}_2^t$ denote the updates at iteration t. Inspired by [42], we adopt a convolutional gated recurrent unit (ConvGRU) to yield these updates, since the ConvGRU can memorize the history status and make full use of the temporal information during the refinement process. The ConvGRU operates at 1/4 resolution.

Specifically, we first project $\mathbf{D}_1^t$, $\mathbf{D}_2^t$, $\mathbf{U}_1$, $\mathbf{U}_2$ and $dif_{\mathbf{D}}^t$ into the feature space using two convolutional layers. Then, we concatenate the projected feature and the image contextual feature (1/4 resolution output in the feature extractor) to form a tensor $\mathbf{I}^t$ as the input. The structure inside ConvGRU is as follows

$$\mathbf{z}^{t+1}=\sigma\left(Conv_{5\times5}\left(\left[\mathbf{h}^t,\mathbf{I}^t\right],W_z\right)\right) \tag{13}$$

$$\mathbf{r}^{t+1}=\sigma\left(Conv_{5\times5}\left(\left[\mathbf{h}^t,\mathbf{I}^t\right],W_r\right)\right) \tag{14}$$

$$\widehat{\mathbf{h}}^{t+1}=\tanh\left(Conv_{5\times5}\left(\left[\mathbf{r}^{t+1}\odot\mathbf{h}^t,\mathbf{I}^t\right],W_h\right)\right) \tag{15}$$

Method	Cap	Abs Rel ↓	Sq Rel ↓	RMSE ↓	$\log_{10}$ ↓	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$
Fu *et al.* [13]	0-10m	0.115	-	0.509	0.051	0.828	0.965	0.992
VNL [44]	0-10m	0.108	-	0.416	0.048	0.875	0.976	0.994
BTS [23]	0-10m	0.113	0.066	0.407	0.049	0.871	0.977	0.995
DAV [18]	0-10m	0.108	-	0.412	-	0.882	0.980	0.996
PWA [24]	0-10m	0.105	-	0.374	0.045	0.892	0.985	0.997
Long *et al.* [30]	0-10m	0.101	-	0.377	0.044	0.890	0.982	0.996
TransDepth [43]	0-10m	0.106	-	0.365	0.045	0.900	0.983	0.996
Adabins [2]	0-10m	0.103	-	0.364	0.044	0.903	0.984	0.997
P3Depth [34]	0-10m	0.104	-	0.356	0.043	0.898	0.981	0.996
NeWCRFs [46]	0-10m	0.095	0.045	0.334	0.041	0.922	**0.992**	**0.998**
Ours	0-10m	**0.087**	**0.041**	**0.311**	**0.038**	**0.936**	0.991	**0.998**

Table 1. **Quantitative depth comparison on the NYU-Depth-v2 dataset**.

$$\mathbf{h}^{t+1} = \left(1 - \mathbf{z}^{t+1}\right) \odot \mathbf{h}^{t} + \mathbf{z}^{t+1} \odot \widehat{\mathbf{h}}^{t+1}, \quad (16)$$

where $Conv_{5\times5}$ stands for the separable 5×5 convolution, $\odot$ denotes the element-wise multiplication, σ and tanh denote the sigmoid and tanh activation functions. The hidden state $\mathbf{h}^t$ is initialized by the concatenated penultimate layer feature maps of the normal-distance and depth heads, with the tanh function as activation.

With this refinement module, starting from an initial solution, the depth maps are iteratively refined and eventually converge to the final results as $\mathbf{D}_1^* \leftarrow \mathbf{D}_1^t, \mathbf{D}_2^* \leftarrow \mathbf{D}_2^t$. Finally, we upsample $\mathbf{D}_1^*$ and $\mathbf{D}_2^*$ to the full resolution, and average them as the output of the whole framework,

$$\mathbf{D}^* (\mathbf{p}) = 0.5(\mathbf{D}_1^* (\mathbf{p}) + \mathbf{D}_2^* (\mathbf{p})). \quad (17)$$

3.4. Overall Loss and Architecture

Depth loss. We adopt a scaled Scale-Invariant loss for depth supervision [23],

$$\mathcal{L}_{\mathbf{D}} = \sum_{s=1}^{m} \gamma^{m-s} \begin{pmatrix} \kappa \sqrt{\frac{1}{|\mathbf{T}|} \sum_{\mathbf{p}} \left(\mathbf{g}_1 (\mathbf{p})\right)^2 - \frac{\eta}{|\mathbf{T}|^2} \left(\sum_{\mathbf{p}} \mathbf{g}_1 (\mathbf{p})\right)^2} + \\ \kappa \sqrt{\frac{1}{|\mathbf{T}|} \sum_{\mathbf{p}} \left(\mathbf{g}_2 (\mathbf{p})\right)^2 - \frac{\eta}{|\mathbf{T}|^2} \left(\sum_{\mathbf{p}} \mathbf{g}_2 (\mathbf{p})\right)^2} \end{pmatrix}, \quad (18)$$

where γ is the decay factor and s is the iteration step, set as 0.85 and 3, respectively, $\mathbf{g}(\mathbf{p}) = \log \mathbf{D}(\mathbf{p}) - \log \mathbf{D}^{gt}(\mathbf{p})$, $\mathbf{T}$ stands for a collection of pixels with valid values, κ and η are selected as 10 and 0.85 based on [23].

Normal loss. We adopt a negative cosine loss for normal supervision [10],

$$\mathcal{L}_{\mathbf{N}} = \sum_{\mathbf{p}} 1 - \mathbf{N}(\mathbf{p}) \mathbf{N}^{gt^{\mathrm{T}}}(\mathbf{p}), \quad (19)$$

Distance loss. We adopt an L1 loss for distance supervision,

$$\mathcal{L}_{\mathcal{D}} = \sum_{\mathbf{p}} \left| \mathcal{D}(\mathbf{p}) - \mathcal{D}^{gt}(\mathbf{p}) \right|, \quad (20)$$

As the NYU-Depth-v2 and KITTI datasets do not contain normal and distance ground-truth, we acquire the normal ground-truth following [36]. Then, we acquire the distance ground-truth by $\mathcal{D}^{gt}(\mathbf{p}) = \mathbf{D}(\mathbf{p})^{gt} \mathbf{N}(\mathbf{p})^{gt} \mathbf{K}^{-1} \widetilde{\mathbf{p}}$.

Overall loss. We define the overall loss as

$$\mathcal{L}_{overall=} \lambda_1 \mathcal{L}_{\mathbf{D}} + \lambda_2 \mathcal{L}_{\mathbf{N}} + \lambda_3 \mathcal{L}_{\mathcal{D}} + \lambda_4 \mathcal{L}_{\mathbf{U}} + \lambda_5 \mathcal{L}_{pc}, \quad (21)$$

where λ_1, λ_2, λ_3, λ_4 and λ_5 are empirically set as 1, 5, 0.25, 1 and 0.01, respectively.

Network architecture. The feature extractor adopts the Swin-L [29] when not otherwise specified. For KITTI official split, the feature extractor is SwinV2-L [28]. The depth head follows the decoder design of NeWCRFs [46]. The normal-distance head shares a same architecture as depth head apart from the final layer, which is devised to estimate normal and distance jointly.

4. Experiment

We conduct comprehensive experiments on datasets for indoor and outdoor scenes, which includes the NYU-Depth-v2, KITTI and SUN RGB-D. In this section, we first give a detailed description of the relevant datasets, evaluation metrics and implementation details. Then, we provide quantitative and qualitative comparisons to previous state-of-the-art competitors. Finally, we demonstrate zero-shot generalization and ablation studies to validate its generalizability and the efficacy of each key component.

4.1. Datasets and Evaluation Metrics

NYU-Depth-v2 dataset is collected from indoor scenarios and provides images and ground-truth depth maps at a resolution of 640×480 pixels. We adopt the official data split to evaluate our method. The split employs 249 scenes for training and 654 images from 215 scenes for testing.

KITTI dataset is acquired from outdoor scenes using camera and LiDAR mounted on a moving vehicle, and con-

Method	Cap	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$
VNL [44]	0-80m	0.072	-	3.258	0.117	0.938	0.990	0.998
BTS [23]	0-80m	0.061	0.261	2.834	0.099	0.954	0.992	0.998
PWA [24]	0-80m	0.060	0.221	2.604	0.093	0.958	0.994	**0.999**
TransDepth [43]	0-80m	0.064	0.252	2.755	0.098	0.956	0.994	**0.999**
Adabins [2]	0-80m	0.058	0.190	2.360	0.088	0.964	0.995	**0.999**
P3Depth [34]	0-80m	0.071	0.270	2.842	0.103	0.953	0.993	0.998
NeWCRFs [46]	0-80m	0.052	0.155	2.129	0.079	0.974	0.997	**0.999**
Ours	0-80m	**0.050**	**0.141**	**2.025**	**0.075**	**0.978**	**0.998**	**0.999**
BTS [23]	0-50m	0.058	0.183	1.995	0.090	0.962	0.994	0.999
PWA [24]	0-50m	0.057	0.161	1.872	0.087	0.965	0.995	0.999
P3Depth [34]	0-50m	0.055	0.130	1.651	0.081	0.974	0.997	0.999
Ours	0-50m	**0.048**	**0.107**	**1.513**	**0.071**	**0.981**	**0.998**	**1.000**

Table 2. **Quantitative depth comparison on the Eigen split of KITTI dataset**. "-" indicates not applicable. The best results are marked in **bold**.

Method	SILog ↓	Sq Rel ↓	Abs Rel ↓	iRMSE↓
P3Depth [34]	12.82	9.92	2.53	13.71
VNL [44]	12.65	10.15	2.46	13.02
Fu *et al.* [13]	11.77	8.78	2.23	12.98
BTS [23]	11.67	9.04	2.21	12.23
BA-Full [1]	11.61	9.38	2.29	12.23
PackNet-SAN [16]	11.54	9.12	2.35	12.38
PWA [24]	11.45	9.05	2.30	12.32
NeWCRFs [46]	10.39	8.37	1.83	11.03
Ours	**9.62**	**7.75**	**1.59**	**10.62**

Table 3. **Quantitative depth comparison on the official split of KITTI dataset**. Note that the Sq Rel is calculated in a different way here. The SILog is the main ranking metric.

sists of stereo images and 3D laser scans. The image resolution is around 1241×376 pixels. Here we adopt two mainly used data splits. One is the Eigen split with 23488 training image pairs and 697 test images [11]. The other one is the official split with 42949 training image pairs, 1000 validation image pairs and 500 test images. For the official split, the ground-truth depth maps of the test images are not available, and the results are provided by the online server.

SUN RGB-D dataset is captured from multiple indoor scenes with four sensors, and contains roughly 10K images. We only use this dataset for generalization study in a challenging zero-shot setup. The pre-trained models are evaluated on the official test set of 5050 images.

Evaluation Metrics. Following [46], we adopt the standard evaluation metrics in our experiments.

4.2. Implementation Details

Our framework is implemented in PyTorch [33] and experimented on NVIDIA RTX A5000 GPUs. We use the Adam optimizer [20] where $\beta_1 = 0.9, \beta_2 = 0.999$ and a batch size of 8. The learning rate is scheduled through polynomial decay from a base value of 2e-5 to 2e-6. The training process runs a total number of 25 epochs.

4.3. Comparison to Prior Competitors

NYU-Depth-v2. As summarized in Table 1, our method surpasses previous competing approaches on most metrics, especially notable are results on Abs Rel and RMSE, which emphasizes our contributions in boosting the performance. In Figure 4, we demonstrate qualitative depth comparisons. As can be seen, our method is better at delineating planar regions while preserving local details, such as edges, and has high immunity against texture variations. Furthermore, we convert depth maps into point clouds, and show qualitative point cloud comparisons in Figure 7. Our point clouds have fewer distortions than others. The point clouds of other approaches suffer from severe distortions and struggle to preserve prominent geometric features.

KITTI. To further show the applicability of our method in the outdoor scenario, we evaluate it on the KITTI dataset. We first perform comparison on the Eigen split. As listed in Table 2, our method exceeds previous leading approaches by a noticeable margin on most metrics, such as Sq Rel and RMSE, when the maximum depth is capped at 80m. Regarding the 0-50m cap, our method goes well beyond the P3Depth. Besides, we notice that compared to the PWA, the P3Depth performs better at the 0-50m cap but significantly worse at the 0-80m cap. This is due to the fact that most of the nearby parts are planar regions such as roads, while the distant parts are occupied by high-curvature regions, such as bushes, trees and other clutter, resulting in the planarity prior to fail. Our method relaxes such failure cases with the help of depth head and achieves promising results on both caps. Figure 5 presents qualitative depth comparisons. As

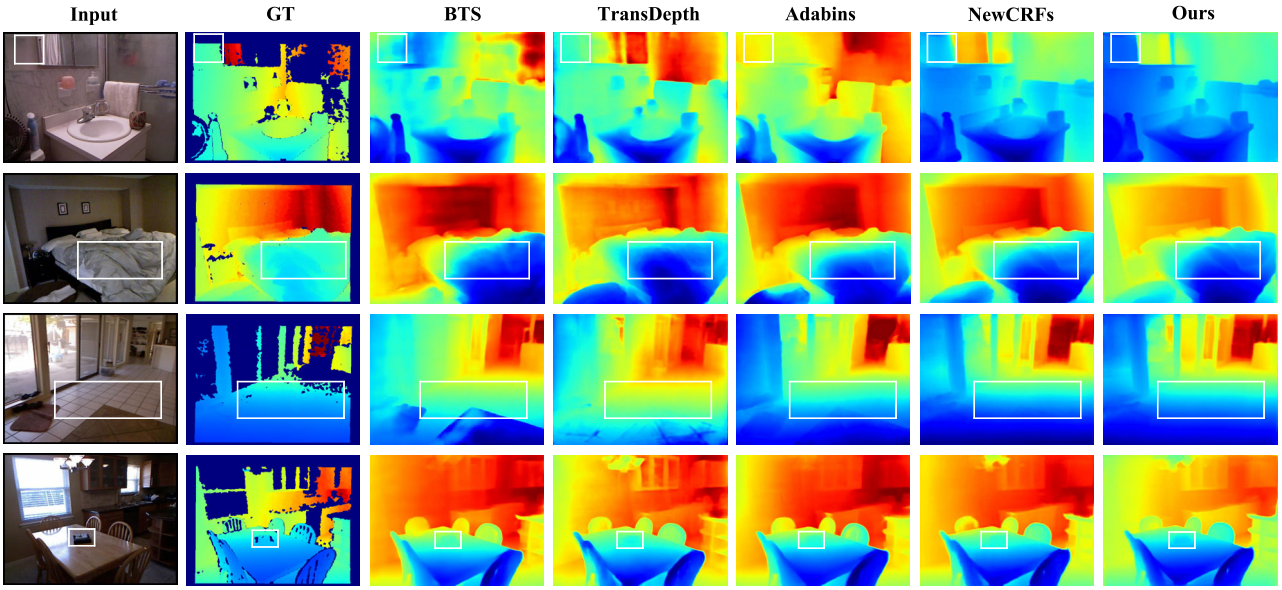

Figure 4. **Qualitative depth results on the NYU-Depth-v2 dataset**. The white boxes highlight the regions to emphasize.

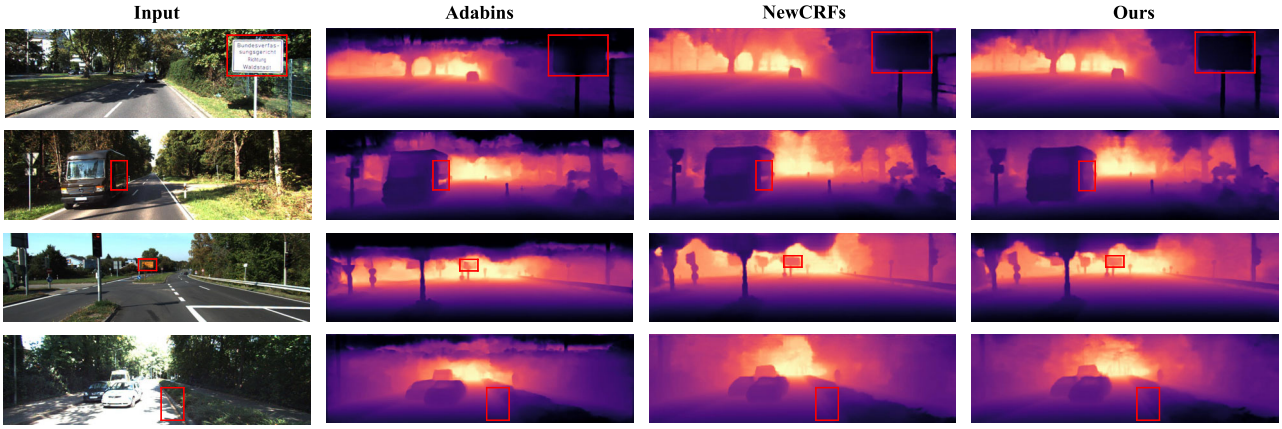

Figure 5. **Qualitative depth results on the KITTI dataset**. The red boxes highlight the regions to emphasize.

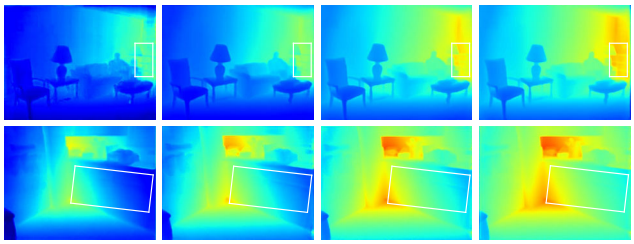

Figure 6. **Visualization of the refinement process in the contrastive iterative refinement module**. The first depth map of the upper row at iter 0 is from the normal-distance head and the first depth map of the lower row at iter 0 is from the depth head. The white boxes show the regions to focus on.

can be seen, our method is more capable of planar regions, e.g., sign board.

We then compare our method against prior leading approaches on the official split. The results are listed in Table 3 and are available from the online server. Here we can see that our method exceeds previous approaches again and achieves consistent improvements on all metrics. Besides, the main ranking metric SILog is reduced markedly, and our method **ranks 1st** among all submissions on the KITTI depth prediction online benchmark at the submission time.

4.4. Zero-shot Generalization

In Table 4, we show generalization comparison in a zero-shot setting where the test dataset has not been seen during training. The models are trained on the NYU-Depthv2 dataset but evaluated on the SUN RGB-D dataset. The superior results indicate that our physics-driven deep learning framework does learn transferable features rather than sim-

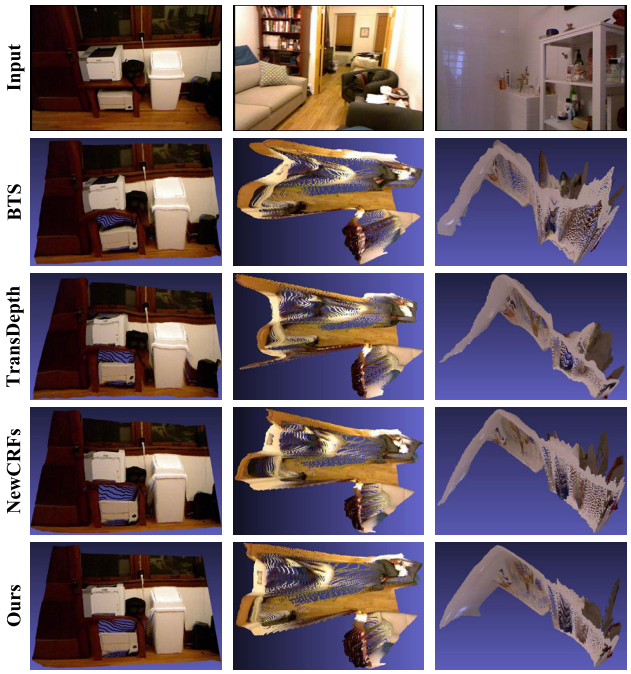

Figure 7. **Qualitative point cloud results on the NYU-Depth-v2 dataset.** The point clouds in the first column are captured from the front view, and the point clouds in the second and third columns are captured from the top view.

Method	Abs Rel ↓	RMSE ↓	$\log_{10}$ ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑
Chen [6]	0.166	0.494	0.071	0.757	0.943
VNL [44]	0.183	0.541	0.082	0.696	0.912
BTS [23]	0.172	0.515	0.075	0.740	0.933
Adabins [2]	0.159	0.476	0.068	0.771	0.944
Ours	**0.137**	**0.411**	**0.060**	**0.820**	**0.970**

Table 4. **Generalization to the SUN RGB-D dataset with models trained on the NYU-Depth-v2 dataset.**

Setting	PC	CIR	Abs Rel ↓	RMSE ↓	$\log_{10}$ ↓	$\delta < 1.25$ ↑
D			0.095	0.334	0.041	0.922
N$\mathcal{D}$			0.092	0.324	0.039	0.926
N$\mathcal{D}$	✓		0.089	0.318	0.039	0.929
D&N$\mathcal{D}$	✓		0.088	0.315	0.038	0.931
D&N$\mathcal{D}$	✓	✓	**0.087**	**0.311**	**0.038**	**0.936**
D (planar)			0.094	0.316	0.040	0.925
N$\mathcal{D}$ (planar)	✓		0.088	0.301	0.038	0.934
D (non-planar)			0.102	0.390	0.043	0.909
N$\mathcal{D}$ (non-planar)	✓		0.104	0.394	0.044	0.908

Table 5. **Results of ablation study**. D&N$\mathcal{D}$: combined depth and normal-distance heads; D: depth head; N$\mathcal{D}$: normal-distance head; PC: plane-aware consistency constraint; CIR: contrastive iterative refinement module.

ply memorizing statistics of training data.

4.5. Ablation Study

To better understand the effect of each key component in our framework, we conduct an ablation study and present the results in Table 5.

We find that the normal-distance head works better than the depth head in a standalone setting, which seems to be contrary to the finding in [34] that directly predicting depth is better than predicting plane coefficients. The reason may be that compared with the implicit plane representation used in [34], we adopt a more explicit normal-distance representation, which allows us to enforce direct supervisory signals to guide the normal and distance learning, hence leading to more accurate depth estimates. By first enforcing the plane-aware consistency constraint to normal and distance, we observe a consistent improvement on most metrics. On the basis of the normal-distance head, we further integrate the depth head into the framework. The resulting improvement indicates that the depth estimates from these two heads are inherently complementary to each other. Finally, we add the contrastive iterative refinement module and form the complete framework, achieving the best results.

In Figure 6, we visualize the refinement process of CIR module. As the number of iterations increases, the locker in the depth maps from the normal-distance head becomes clearer, and the kitchen cabinet surface in the depth maps from the depth head becomes more and more continuous.

We also compare the depth accuracy of normal-distance head and depth head in planar and non-planar regions, respectively, and the results support our standpoint.

4.6. Conclusion

In this paper, we introduce a physics-driven deep learning framework for monocular depth estimation that leverages the planar information in real-world 3D scenes. The framework contains two heads, a normal-distance head and a depth head. The predicted normal and distance are regularized by a developed plane-aware consistency constraint. Besides, we develop a contrastive iterative refinement module to make full use of the strengths of these two heads according to the depth uncertainty. Extensive experiments indicate that the proposed method outperforms previous competitors on the NYU-Depth-v2, KITTI and SUN RGB-D datasets. Existing works on monocular depth estimation ignore possible saturations in the input image. However, this is not always true in real-world applications [47]. It is desired to study 3D imaging and high dynamic range imaging together and this topic will be studied in our future research.

4.7. Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under grant 51975029 and U1909215, in part by the A*STAR Singapore under

MTC Programmatic Funds grant M23L7b0021, in part by the Key Research and Development Program of Zhejiang Province under Grant 2021C03050, in part by the Scientific Research Project of Agriculture and Social Development of Hangzhou under Grant No. 20212013B11, and in part by the National Natural Science Foundation of China under grant 61620106012 and 61573048.

References

[1] Shubhra Aich, Jean Marie Uwabeza Vianney, Md Amirul Islam, and Mannat Kaur Bingbing Liu. Bidirectional attention network for monocular depth estimation. In *2021 IEEE International Conference on Robotics and Automation*, pages 11746–11752. IEEE, 2021.

[2] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4009–4018, 2021.

[3] András Bódis-Szomorú, Hayko Riemenschneider, and Luc Van Gool. Fast, approximate piecewise-planar modeling based on sparse structure-from-motion and superpixels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 469–476, 2014.

[4] Yuanzhouhan Cao, Zifeng Wu, and Chunhua Shen. Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(11):3174–3182, 2017.

[5] Anne-Laure Chauve, Patrick Labatut, and Jean-Philippe Pons. Robust piecewise-planar 3d reconstruction and completion from large-scale unstructured point data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1261–1268. IEEE, 2010.

[6] Xiaotian Chen, Xuejin Chen, and Zheng-Jun Zha. Structure-aware residual pyramid network for monocular depth estimation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 694–700, 2019.

[7] Alejo Concha and Javier Civera. Using superpixels in monocular slam. In *IEEE International Conference on Robotics and Automation*, pages 365–372. IEEE, 2014.

[8] Alejo Concha and Javier Civera. Dpptam: Dense piecewise planar tracking and mapping from a monocular sequence. In *IEEE International Conference on Intelligent Robots and Systems*, pages 5686–5693. IEEE, 2015.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations*, 2021.

[10] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2650–2658, 2015.

[11] 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.

[12] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59:167–181, 2004.

[13] 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.

[14] David Gallup, Jan-Michael Frahm, and Marc Pollefeys. Piecewise planar and non-planar stereo for urban scene reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1418–1425. IEEE, 2010.

[15] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[16] Vitor Guizilini, Rares Ambrus, Wolfram Burgard, and Adrien Gaidon. Sparse auxiliary networks for unified monocular depth prediction and completion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11078–11088, 2021.

[17] Caner Hazirbas, Lingni Ma, Csaba Domokos, and Daniel Cremers. Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. In *Asian Conference on Computer Vision*, pages 213–228. Springer, 2016.

[18] Lam Huynh, Phong Nguyen-Ha, Jiri Matas, Esa Rahtu, and Janne Heikkilä. Guiding monocular depth estimation using depth-attention volume. In *European Conference on Computer Vision*, pages 581–597. Springer, 2020.

[19] Weibin Jia, Wenjie Zhao, Zhihuan Song, and Zhengguo Li. Object servoing of differential-drive service robots using switched control. *Journal of Control and Decision*, pages 1–12, 2022.

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[21] Uday Kusupati, Shuo Cheng, Rui Chen, and Hao Su. Normal assisted stereo depth estimation. In *Proceedings of the IEEEF Conference on Computer Vision and Pattern Recognition*, pages 2189–2199, 2020.

[22] 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*, pages 239–248. IEEE, 2016.

[23] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv preprint arXiv:1907.10326*, 2019.

[24] Sihaeng Lee, Janghyeon Lee, Byungju Kim, Eojindl Yi, and Junmo Kim. Patch-wise attention network for monocular depth estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1873–1881, 2021.

[25] Wonwoo Lee, Nohyoung Park, and Woontack Woo. Depth-assisted real-time 3d object detection for augmented reality. In *ICAT*, volume 11 (2), pages 126–132, 2011.

[26] Boying Li, Yuan Huang, Zeyu Liu, Danping Zou, and Wenxian Yu. Structdepth: Leveraging the structural regularities for self-supervised indoor depth estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 12663–12673, 2021.

[27] Chen Liu, Kihwan Kim, Jinwei Gu, Yasutaka Furukawa, and Jan Kautz. Planercnn: 3d plane detection and reconstruction from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4450–4459, 2019.

[28] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 12009–12019, 2022.

[29] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10012–10022, October 2021.

[30] Xiaoxiao Long, Cheng Lin, Lingjie Liu, Wei Li, Christian Theobalt, Ruigang Yang, and Wenping Wang. Adaptive surface normal constraint for depth estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 12849–12858, October 2021.

[31] Jeff Michels, Ashutosh Saxena, and Andrew Y Ng. High speed obstacle avoidance using monocular vision and reinforcement learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 593–600, 2005.

[32] Takaaki Nagai, Takumi Naruse, Masaaki Ikehara, and Akira Kurematsu. Hmm-based surface reconstruction from single images. In *Proceedings of the International Conference on Image Processing*, volume 2, pages II–II. IEEE, 2002.

[33] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *Advances in Neural Information Processing Systems Workshop Autodiff*, 2017.

[34] Vaishakh Patil, Christos Sakaridis, Alexander Liniger, and Luc Van Gool. P3depth: Monocular depth estimation with a piecewise planarity prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1610–1621, 2022.

[35] Xiaojuan Qi, Renjie Liao, Zhengzhe Liu, Raquel Urtasun, and Jiaya Jia. Geonet: Geometric neural network for joint depth and surface normal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 283–291, 2018.

[36] Jiaxiong Qiu, Zhaopeng Cui, Yinda Zhang, Xingdi Zhang, Shuaicheng Liu, Bing Zeng, and Marc Pollefeys. Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3313–3322, 2019.

[37] Ashutosh Saxena, Sung H Chung, Andrew Y Ng, et al. Learning depth from single monocular images. In *Advances in Neural Information Processing Systems*, volume 18, pages 1–8, 2005.

[38] Shuwei Shao, Zhongcai Pei, Weihai Chen, Ran Li, Zhong Liu, and Zhengguo Li. Urcdc-depth: Uncertainty rectified cross-distillation with cutflip for monocular depth estimation. *arXiv preprint arXiv:2302.08149*, 2023.

[39] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012.

[40] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 567–576, 2015.

[41] Keisuke Tateno, Federico Tombari, Iro Laina, and Nassir Navab. Cnn-slam: Real-time dense monocular slam with learned depth prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6243–6252, 2017.

[42] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Proceedings of the European Conference on Computer Vision*, pages 402–419. Springer, 2020.

[43] Guanglei Yang, Hao Tang, Mingli Ding, Nicu Sebe, and Elisa Ricci. Transformer-based attention networks for continuous pixel-wise prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 16269–16279, October 2021.

[44] Wei Yin, Yifan Liu, Chunhua Shen, and Youliang Yan. Enforcing geometric constraints of virtual normal for depth prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5684–5693, 2019.

[45] Zehao Yu, Lei Jin, and Shenghua Gao. P 2 net: patch-match and plane-regularization for unsupervised indoor depth estimation. In *Proceedings of the European Conference on Computer Vision*, pages 206–222. Springer, 2020.

[46] Weihao Yuan, Xiaodong Gu, Zuozhuo Dai, Siyu Zhu, and Ping Tan. Neural window fully-connected crfs for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3916–3925, June 2022.

[47] Chaobing Zheng, Weibin Jia, Shiqian Wu, and Zhengguo Li. Neural augmented exposure interpolation for two large-exposure-ratio images. *IEEE Transactions on Consumer Electronics*, 69(1):87–97, 2022.