From Big to Small: Multi-Scale Local Planar Guidance for Monocular Depth Estimation

Jin Han Lee, Myung-Kyu Han, Dong Wook Ko and Il Hong Suh
Department of Electronics and Computer Engineering, Hanyang University
{jinhanlee, mkhan91, pumpblack, ihsuh}@hanyang.ac.kr

Abstract

Estimating accurate depth from a single image is challenging, because it is an ill-posed problem as infinitely many 3D scenes can be projected to the same 2D scene. However, recent works based on deep convolutional neural networks show great progress achieving plausible results. The networks are generally composed of two parts: an encoder for dense feature extraction and a decoder for predicting the desired depth. In the encoder-decoder schemes, repeated strided convolution and spatial pooling layers lower the spatial resolution of transitional outputs, and several techniques such as skip connections or multi-layer deconvolutional networks are adopted to effectively recover back to the original resolution.

In this paper, for a more effective guidance of densely encoded features to desired depth prediction, we propose a network architecture that utilizes novel local planar guidance layers located at multiple stages in decoding phase. We show that the proposed method outperforms the state-of-the-art works with significant margin evaluating on challenging benchmarks. We also provide results from an ablation study to validate the effectiveness of the proposed core factors. 利用局部平面指导层在解码过程中多个阶段定位

1. Introduction

Depth estimation from 2D images has been studied in computer vision for a long time, and is nowadays applied to robotics, autonomous driving cars, scene understanding and 3D reconstructions. Those applications usually utilize, to perform depth estimation, multiple instances of the same scene such as stereo image pairs [39], multiple frames from moving camera [34] or static captures under different lighting conditions [2, 3]. As depth estimation from multiple observations achieves great progress, it naturally leads to depth estimation with a single image since it demands ultimately less cost and constraint.

However, estimating accurate depth from a single image is challenging, even for human, because it is an ill-posed problem as infinitely many 3D scenes can be projected to the same 2D scene. To understand geometric configuration thus resulting depth from a single image, humans are considered to use not only local cues such as texture appearance in various lighting and occlusion conditions, perspective, or relative scales to the known objects, but also global context such as entire shape or layout of the scene [19].

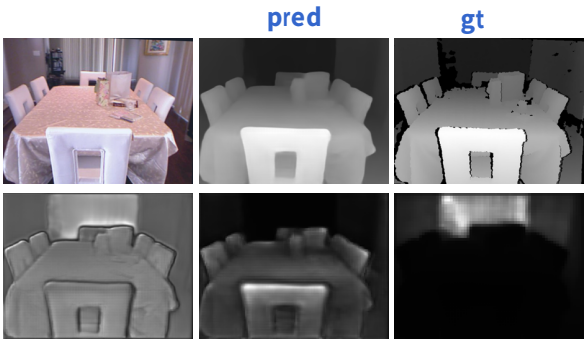


Figure 1: **A depth training result from the proposed network.** Top: from left to right, input image, learned depth map, and the ground truth. Bottom: from left to right, outputs from the proposed local planar guidance layers having input feature resolutions of 1/2, 1/4, 1/8 to the input image, respectively.

After the first learning-based monocular depth estimation work from Saxena et al. [37] was introduced, considerable improvements have been made along with rapid advances in deep learning [12, 11, 28, 29, 43, 35, 21, 25]. While most of the state-of-the-art works apply models based on deep convolutional neural netowrks (DCNNs) in *supervised* fashion, some works proposed *semi-* [24] or *self-supervised* learning methods which do not fully rely on the ground truth depth data.

In the meantime, recent applications based on DCNNs are commonly composed in two parts: encoder for dense feature extraction and decoder for desired prediction. As a dense feature extractor, very powerful deep networks such as VGG [41], ResNet [18] or DenseNet [20] are usually

adopted. In these networks, repeated strided convolution and spatial pooling layers lower the spatial resolution of transitional outputs, which can be a bottleneck to obtain desired predictions in high resolution. Therefore, a number of techniques, for example, multi-scale networks [29, 11], skip connections [17, 45] or multi-layer deconvolutional networks [25, 15, 24] are applied to consolidate feature maps from higher resolutions. Recently, atrous spatial pyramid pooling (ASPP) [7] has been introduced for image semantic segmentation which can capture large scale variations in observation by applying sparse convolutions with various dilation rates. Since the dilated convolution allows larger receptive field size, recent works in semantic segmentation [7, 47] or in depth estimation [13] do not fully reduce the receptive field size by removing last few pooling layers and reconfigure the network with atrous convolutions to reuse pretrained weights. Consequently, their networks have larger dense features (1/8 of input spatial resolution whereas 1/32 or 1/64 in the original base networks) and perform almost all of the decoding process on that resolution followed by a simple upsampling to recover to input resolution.

To enable a more explicit relation in recovering back to the full resolution, we propose a network architecture that utilizes novel local planar guidance layers located at multiple stages in decoding phase. More specifically, based on an encoding-decoding scheme, at each decoding stage which has spatial resolutions of 1/8, 1/4 and 1/2, we place a layer that effectively guides input features to desired depth by leading each feature with local planar assumption. Then, we combine the outputs to predict depth in full resolution. This differs from multi-scale network [11, 12] or image pyramid [17] approaches in two aspects. First, the outputs from the proposed layers are not treated as separated estimation in downsampled resolutions, rather, we let the layers to learn 4-dimensional plane coefficients and use them together to reconstruct depth estimations in the full resolution for the final output. Second, as a consequence of the combination, individual spatial cells in each resolution are distinctively activated according to the spatial extent or depth of the object. We can see an example of outputs from the proposed layers in Figures 1 and 3. Experiments on the challenging NYU Depth V2 dataset [31] and KITTI dataset [16] demonstrate that the proposed method achieves state-of-the-art results.

The rest of this paper is organized as follows. After a concise survey of related works in Section 2, we present in detail the proposed method in Section 3. Then, in Section 4, we provide results on two benchmarks comparing with state-of-the-art works, as well as from an ablation study conducted to validate the effectiveness of the proposed method. We conclude the paper in Section 5.

2. Related Work

2.1. Supervised Monocular Depth Estimation

In monocular depth estimation, supervised approaches take a single image and use depth data measured with range sensors such as RGB-D cameras or multi-channel laser scanners as ground truth for supervision in training. Saxena et al. [37] propose a learning-based approach to get a functional mapping from visual cues to depth via markov random field, and extend it to a patch-based model that first over-segments the input image and learns 3D orientation as well as location of local planes that are well explained by each patch [38]. Eigen et al. [11] introduce a multi-scale convolutional architecture that learns coarse global depth predictions on one network and progressively refine them using another network. Unlike the previous works in single image depth estimation, their network can learn representations from raw pixels without hand crafted features such as contours, super-pixels or low-level segmentations. Several works follow the success of this approach by incorporating strong scene priors for surface normal estimation [44], using conditional random fields to improve accuracy [27, 23, 40] or changing the learning problem from regression to classification [5]. Recent supervised approach from Fu et al. [13] achieves state-of-the-art result by also taking advantages from changing the regression problem to quantized ordinal regression. Xu et al. [46] propose an architecture which exploits multi-scale estimations derived from inner layers by fusing them within a CRF framework. Gan et al. [14] propose to explicitly model the relationships between different image locations with an affinity layer.

2.2. Semi-Supervised Monocular Depth Estimation

There are also attempts to train a depth estimation network in semi-supervised or weakly supervised fashion. Chen et al. [8] propose a new approach that uses information of relative depth and depth ranking loss function to learn depth predictions in unconstrained images. Recently, to overcome difficulty in getting high quality depth data, Kuznietsov et al. [24] introduce a semi-supervised method to train the network using both sparse LiDAR depth data for direct supervision and image alignment loss as an indirect training objective.

2.3. Self-Supervised Monocular Depth Estimation

Self-supervised approach refers to a method that requires only rectified stereo image pairs to train the depth estimation network. Garg et al. [15] and Godard et al. [17] propose *self-supervised* learning methods that smartly cast the problem from direct depth estimation to image reconstruction. Specifically, with a rectified stereo image pair, their networks try to synthesize one view from the other with estimated disparities and define the error between both

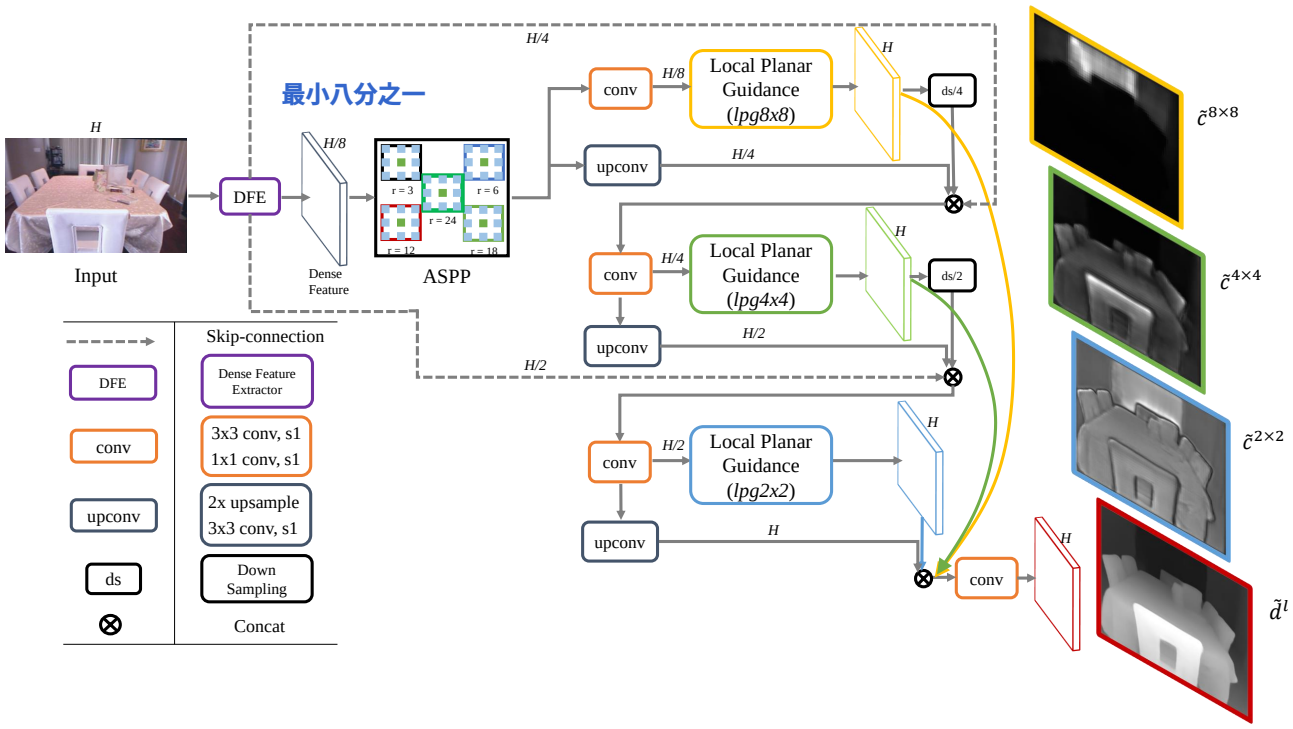


Figure 2: **Overview of the proposed network architecture.** The network is composed of dense feature extractor (the base network), contextual information extractor (ASPP), local planar guidance layers and their dense connection for final depth estimation. Note that the outputs from the local planar guidance layers have the full spatial resolution H. We also use skip-connections from the base network to link with internal outputs in the decoding phase with corresponding spatial resolutions.

as the reconstruction loss for the main training objective. In this way, because learning requires only well rectified, synchronized stereo pairs instead of the ground truth depth data also well associated with the corresponding RGB images, it greatly reduces the effort to acquire datasets for new category of scenes or environments. However, there are some accuracy gap when compared to the current best supervised approach [13]. Garg et al. [15] introduce an encoder-decoder architecture and to train the network using photometric reconstruction error. Xie et al. [45] propose a network which also synthesizes one view from the other, and by using the reconstruction error they produce probability distribution of possible disparities for each pixel. Godard et al. [17] finally propose a network architecture that can perform end-to-end training. They also present a novel left-right consistency loss that improves training and predictions of the network.

2.4. Video-Based Monocular Depth Estimation

There are also approaches using sequential data to perform monocular depth estimation. Yin et al. [48] propose an architecture consists of two generative sub-networks which are jointly trained by adversarial learning for disparity map estimation organized in a cycle to provide mutual constraints. Mahjourian et al. [30] present an approach that explicitly consider the inferred 3D geometry of the whole scene, and enforce consistency of the estimated 3D point clouds and ego-motion across consecutive frames. Wang et al. [42] adopt a differentiable pose predictor and train a monocular depth estimation network in an end-to-end fashion while benefited from the pose predictor.

3. Method

In this section, we describe the proposed monocular depth estimation network with a novel local planar guidance layer located on multiple stages in decoding phase.

在解码中多个阶段使用局部平面指导层

3.1. Network Architecture

As it can be seen from Figure 2, we follow an encoding-decoding scheme that reduces feature map resolution to $H/8$ then recovers back to the original resolution H. After the dense feature extractor which produces an $H/8$ feature map, we place a denser version [47] of atrous spatial pyramid pooling layer [7] as our contextual information ex-

最小到1/8再恢复

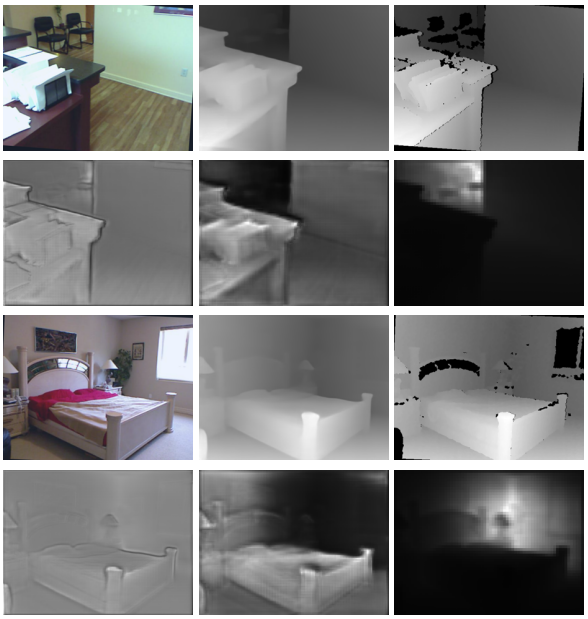


Figure 3: **More training results from the proposed network.** The layout of the figures is same with Figure 1.

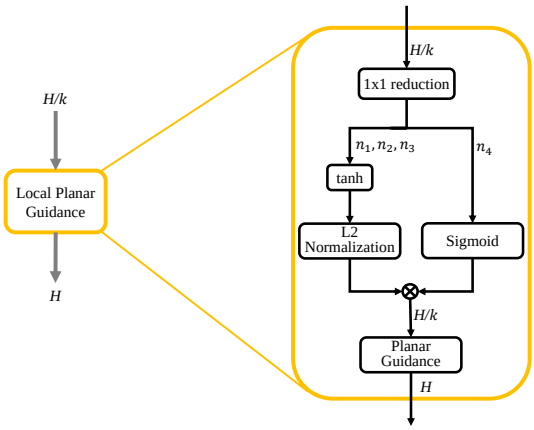


Figure 4: **The local planar guidance layer.** We use a stack of 1×1 convolutions to get 4D plane estimations. (*i.e.*, $H/k \times H/k \times 4$). Then the channels are split to pass through two different activation mechanisms to ensure plane coefficients' constraint. Finally they are fed into the module to compute depths with the estimated planes.

ax+by+cz=d等价于auz+bvz+cz=d，因此z=d/(ax+by+c)

tractor with various dilation rates $r \in \{3, 6, 12, 18, 24\}$. Then, at each stage in the decoding phase where internal outputs are recovered to the full resolution with a factor of 2, we employ the proposed local planar guidance (LPG) layer to more effectively relate the features to the desired depth estimation. Finally, outputs from the proposed layers are concatenated and fed into the final convolution layer to get depth estimation $\tilde{d}$.

解码阶段每次上采样2倍

每层输出融合之后过卷积得到深度

3.2. Multi-Scale Local Planar Guidance

Our key idea in this work is to define more explicit relation between internal features and the final output. Unlike the existing methods that recovers back to the original resolution using simple nearest neighbor upsampling layers and skip connections, we place additional layers which guide features to the full resolution with the local planar assumption, and use them together to get the final depth estimation. Specifically, given a feature map having spatial resolution H/k, the proposed layers estimate for each spatial cell a 4D plane coefficients that fits a $k \times k$ patch on the full resolution H and use it to relate features. At a spatial location in a LPG layer, if the related features describes well the corresponding part of the real world, it would be activated and its coefficients will be trained more precisely, otherwise, it would be deactivated (or weakly activated) and corresponding finer parts at higher resolution or a coarser part at lower resolution get a chance for a stronger activation in the same manner. Since they are concatenated and used together for depth estimation through the final convolutional layer, we expect that the decoding phase to approximate 3D shapes of the given scene with combination of those guided features. Therefore, as it can be seen from Figures 1 and 3, they can have different signs as well as distinct characteristics.

不像直接预测深度，定义了特征与输出的明确关系

用局部平面假设额外层指导特征到全图

预测4D平面系数拟合kxk图像块

如果相关特征描述正确就激活并精确训练

否则

To guide features with the local planar assumption, we convert each estimated 4D plane coefficients to $k \times k$ local depths using the following equation:

用下列公式将4D平面参数转化为kxk局部深度

$$\tilde{c}_i = \frac{n_4 \sqrt{u_i^2 + v_i^2 + 1}}{n_1 u_i + n_2 v_i + n_3}, \tag{1}$$

where (n_1, n_2, n_3, n_4) is the estimated plane coefficients, (u_i, v_i) are normalized coordinate of pixel i, respectively. Its derivative with respect to n is straightforward:

n1-4是平面系数uv是像素i的归一化坐标。

$$\begin{aligned}
\frac{\partial \tilde{c}_i}{\partial n_1} &= \frac{-n_4 u_i \sqrt{u_i^2 + v_i^2 + 1}}{(n_1 u_i + n_2 v_i + n_3)^2}, \\
\frac{\partial \tilde{c}_i}{\partial n_2} &= \frac{-n_4 v_i \sqrt{u_i^2 + v_i^2 + 1}}{(n_1 u_i + n_2 v_i + n_3)^2}, \\
\frac{\partial \tilde{c}_i}{\partial n_3} &= \frac{-n_4 \sqrt{u_i^2 + v_i^2 + 1}}{(n_1 u_i + n_2 v_i + n_3)^2}, \\
\frac{\partial \tilde{c}_i}{\partial n_4} &= \frac{\sqrt{u_i^2 + v_i^2 + 1}}{n_1 u_i + n_2 v_i + n_3}.
\end{aligned} \tag{2}$$

Then, the gradient for back propagation is computed as follows,

$$\frac{\partial C}{\partial n} = \sum_{i=0}^{k^2-1} \frac{\partial \tilde{c}_i}{\partial n} \frac{\partial C}{\partial \tilde{c}_i}, \tag{3}$$

where $k \in \{2, 4, 8\}$ for *lpg2x2*, *lpg4x4* or *lpg8x8* layers, respectively, $\frac{\partial C}{\partial \tilde{c}}$ is the gradient back propagated from the preceding layer.

Method	δ_1	δ_2	δ_3	AbsRel	RMSE	log10
Saxena et al. [38]	0.447	0.745	0.897	0.349	1.214	-
Wang et al. [43]	0.605	0.890	0.970	0.220	0.824	-
Liu et al. [29]	0.650	0.906	0.976	0.213	0.759	0.087
Eigen et al. [11]	0.769	0.950	0.988	0.158	0.641	-
Chakrabarti et al. [6]	0.806	0.958	0.987	0.149	0.620	-
Li et al. [28]	0.789	0.955	0.988	0.152	0.611	0.064
Laina et al. [25]	0.811	0.953	0.988	0.127	0.573	0.055
Xu et al. [46]	0.811	0.954	0.987	0.121	0.586	0.052
Lee at al. [26]	0.815	0.963	0.991	0.139	0.572	-
Fu et al. [13]	0.828	0.965	0.992	0.115	0.509	0.051
Qi et al. [33]	0.834	0.960	0.990	0.128	0.569	0.057
Ours	**0.882**	**0.979**	**0.995**	**0.112**	**0.352**	**0.047**

Table 1: **Evaluation results on NYU Depth v2**. Ours outperforms previous works with a significant margin in all measures.

1x1卷积堆叠每次通道减半知道4通道

Figure 4 shows the detail of the proposed layer. Through a stack of 1×1 convolutions where repeatedly reduce channels by a factor of 2 with 1×1 convolutions until it reaches to 4, if we assume a square input without loss of generality, we get a $H/k \times H/k \times 4$ feature map. Then, pass it through two different ways to ensure constraints of plane coefficients: one way is a series of tanh and L2-Normalization for a unit normal vector, and the other is a sigmoid function followed by scaling with the maximum distance c for location of the plane. Finally, they are concatenated again and used for local depth estimation using Equation 1. Here, we consider the local depth as an additive depth defined locally that can be a small detail on a fine scale or a component of global 3D layout on a coarse scale. By an analysis on representation learning and incorporated priors [4], we can expect that as training progresses the network will try to learn more efficient representations. Since features at the same spatial location in different stages are used together to predict the final depth, for an efficient representation, we can expect that global shapes will be learned at coarser scales while local details will be learned at finer scales. For example, we can see from Figures 1 and 3 that outputs from *lpg8×8* show the global shape of the scene while outputs from *lpg2×2* show fine details of object boundaries.

tanh和L2归一化为一个单位法向量过一个sigmoid最后再cat

全局形状由粗糙尺度学习局部细节由精细尺度学习

3.3. Training Loss

In [12], Eigen et al introduce a scale-invariant error and inspired from it they use a following training loss:

$$D(g) = \frac{1}{n}\sum_i g_i^2 - \frac{\lambda}{n^2}\left(\sum_i g_i\right)^2, \tag{4}$$

where $g_i = \log y_i - \log y_i^*$ and $\lambda = 0.5$ in their work. By simply rewriting Equation 4,

$$D(g) = \frac{1}{n}\sum_i g_i^2 - \left(\frac{1}{n}\sum_i g_i\right)^2 + (1-\lambda)\left(\frac{1}{n}\sum_i g_i\right)^2,$$

就是方差计算公式

we can see that it is a sum of the variance and a weighted squared mean of the error in log space. Therefore, setting a higher λ enforces a more focusing on minimizing the variance of the error, and we use $\lambda = 0.85$ in this work. Also, we empirically found that proper scaling of the domain and range of the loss function improves the convergence and final training result. Finally, our training loss L is defined as follows:

$$L = \alpha D(h), \tag{5}$$

where $\alpha = 10$, $h = \log \beta y_i - \log \beta y_i^*$, and we choose $\beta = 100$ and $\beta = 10$ for NYU and KITTI datasets, respectively.

合适的尺度和损失范围有助于收敛

4. Experiments

To verify the effectiveness of our approach, we provide results from a number of experiments. After presenting the implementation details of our method, we provide experimental results on two challenging benchmarks covering both of indoor and outdoor environments to compare with state-of-the-art works. We also provide scores on the online KITTI evaluation server. Then, we provide an ablation study to discuss detailed analysis of the proposed core factors, and some qualitative results to demonstrate our approach comparing with the competing work.

4.1. Implementation Details

We implement the proposed network using the open deep learning framework *Tensorflow* [1]. Because *Tensorflow* does not provide functionalities for our local planar guidance layer, we implement it using *Tensorflow*'s *C++ API*. For training, we use Adam optimizer [22] with $\beta_1 = 0.9, \beta_2 = 0.999$ and $\epsilon = 10^{-8}$, learning is scheduled via polynomial decay from base learning rate 10^{-4} with power $p = 0.9$. The total number of epochs is set to 50 with batch size 16 for all experiments in this work.

As the encoder for dense feature extraction, we use DenseNet-161 [20] with pretrained weights trained for image classification using ILSVRC dataset [36]. Because

Method	cap	*higher is better*			*lower is better*			
		$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	Abs Rel	Sq Rel	RMSE	RMSE *log*
Saxena et al. [38]	0-80m	0.601	0.820	0.926	0.280	3.012	8.734	0.361
Eigen et al. [12]	0-80m	0.702	0.898	0.967	0.203	1.548	6.307	0.282
Liu et al. [29]	0-80m	0.680	0.898	0.967	0.201	1.584	6.471	0.273
Godard et al. (CS+K) [17]	0-80m	0.861	0.949	0.976	0.114	0.898	4.935	0.206
Kuznietsov et al. [24]	0-80m	0.862	0.960	**0.986**	0.113	0.741	4.621	0.189
Gan et al. [14]	0-80m	0.890	0.964	0.985	0.098	0.666	3.933	0.173
Fu et al. [13]	0-80m	0.897	0.966	**0.986**	0.099	0.593	**3.714**	**0.161**
Ours	0-80m	**0.904**	**0.967**	0.984	**0.091**	**0.555**	4.033	0.174
Garg et al. [15]	0-50m	0.740	0.904	0.962	0.169	1.080	5.104	0.273
Godard et al. (CS+K) [17]	0-50m	0.873	0.954	0.979	0.108	0.657	3.729	0.194
Kuznietsov et al. [24]	0-50m	0.875	0.964	**0.988**	0.108	0.595	3.518	0.179
Gan et al. [14]	0-50m	0.898	0.967	0.986	0.094	0.552	3.133	0.165
Fu et al. [13]	0-50m	0.906	0.968	0.986	0.096	0.503	**2.902**	**0.155**
Ours	0-50m	**0.914**	**0.970**	0.986	**0.088**	**0.437**	3.127	0.165
Godard et al. (CS+K)*	0-80m	0.919	0.982	0.995	0.081	0.487	3.687	0.131
Fu et al.*	0-80m	0.936	0.986	0.995	0.081	0.337	2.930	0.121
Ours*	0-80m	**0.950**	**0.993**	**0.999**	**0.064**	**0.254**	**2.815**	**0.100**
Godard et al. (CS+K)*	0-50m	0.929	0.985	0.996	0.076	0.334	2.613	0.121
Fu et al.*	0-50m	0.944	0.987	0.995	0.078	0.273	2.184	0.115
Ours*	0-50m	**0.959**	**0.994**	**0.999**	**0.060**	**0.182**	**2.005**	**0.092**

Table 2: **Performance on KITTI Eigen split.** (CS+K) denotes a model pre-trained on Cityscapes dataset [10] and fine-tuned on KITTI. * denotes that the method is evaluated using the official ground truth, otherwise, evaluated with raw LiDAR scan data. All methods were evaluated on the central crop proposed by Garg et al. [15].

Method	SILog	sqErrorRel	absErrorRel	iRMSE
Official Baseline	18.19	7.32	14.24	18.50
BTS	**11.67**	**2.21**	**9.04**	**12.23**

Table 3: **Result on the online KITTI evaluation server.**

weights at early convolutions are known to be well trained for primitive visual features, in the base network, we fix *dense1* and *dense2* blocks as well as batch normalization parameters in our training. Following [17], we use exponential linear units [9] as an activation function, and *upconv* uses a nearest neighbor upsampling followed by a 3×3 convolution layer [32]. The total number of parameters is 47M.

To avoid over-fitting, we augment images before input to the network using random horizontal flipping, random contrast, brightness and color adjustment in ranges of [0.8, 1.2], [0.5, 1.5] and [0.8, 1.2], respectively, with 50% of chance. We also use a random rotation of the input images in a range of $[-5, 5]$ degrees. We train our network on a random crop of size 352×704 for KITTI and 416×544 for NYU Depth V2 datasets.

4.2. NYU Depth V2 Dataset

The NYU Depth V2 dataset [31] contains 120K RGB and depth pairs having size of 480×640 acquired as video sequences using a Microsoft Kinect from 464 indoor scenes. We follow the official train/test split as previous works, using 249 scenes for training and 215 scenes (654 images) for testing. From the total 120K image-depth pairs, due to asynchronous capturing rates between RGB images and depth maps, using timestampes we associate and sample them by even-spacing in time, resulting 24231 image-depth pairs for the training set. Using raw depth images and camera projections provided by the dataset, we align the image-depth pairs for accurate pixel registrations.

4.3. KITTI Dataset

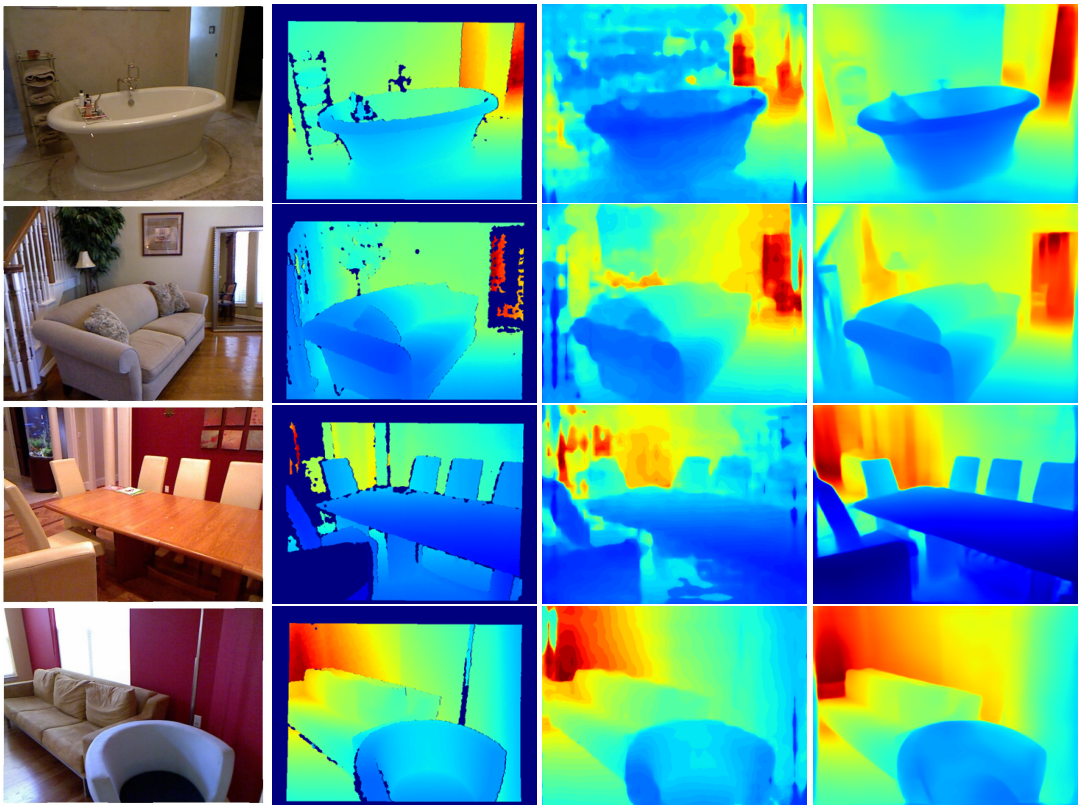
KITTI provides the dataset [16] with 61 scenes from "city", "resiential", "road" and "campus" categories. Because existing works commonly use a split proposed by Eigen et al. [12] for the training and test, we also follow it to compare with those works. Therefore, 697 images covering a total of 29 scenes are used for evaluation and the remaining 32 scenes of 23,488 images are used for the training. In evaluation, we also use a central crop from [15] as in previous works.

4.4. Evaluation Result

For evaluation, we use following metrics used by previous works: Threshold : % of y_i s.t. $\max(\frac{y_i}{y_i^*}, \frac{y_i^*}{y_i}) = \delta < thr$, Abs Rel : $\frac{1}{|T|}\sum_{y \in T}|y - y^*|/y^*$, Sq Rel :

Variant	# Params	*higher is better*			*lower is better*				
		$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	Abs Rel	Sq Rel	RMSE	RMSE *log*	log10
Baseline	42.99M	0.817	0.963	0.992	0.145	0.092	0.438	0.175	0.060
Baseline + ASPP	46.36M	0.833	0.969	0.993	0.135	0.082	0.424	0.166	0.056
Baseline + ASPP + Up	46.96M	0.851	0.971	0.992	0.130	0.080	0.394	0.158	0.053
Baseline + ASPP + Up + LPG	47.00M	0.871	0.977	0.995	0.118	0.067	0.377	0.147	0.049
Ours	47.00M	**0.882**	**0.979**	**0.995**	**0.112**	**0.059**	**0.352**	**0.140**	**0.047**

Table 4: **Result from the ablation study using NYU Depth V2 dataset.** Baseline: a network with the dense feature extractor, ASPP: ASPP module attached after the dense feature extractor, Up: using *upconv* layers in Figure 2, LPG: the proposed local planar guidance layers. All variants are trained using Equation 4 with $\lambda = 0.5$ as the training loss while 'Ours' uses the training loss given in Equation 5.



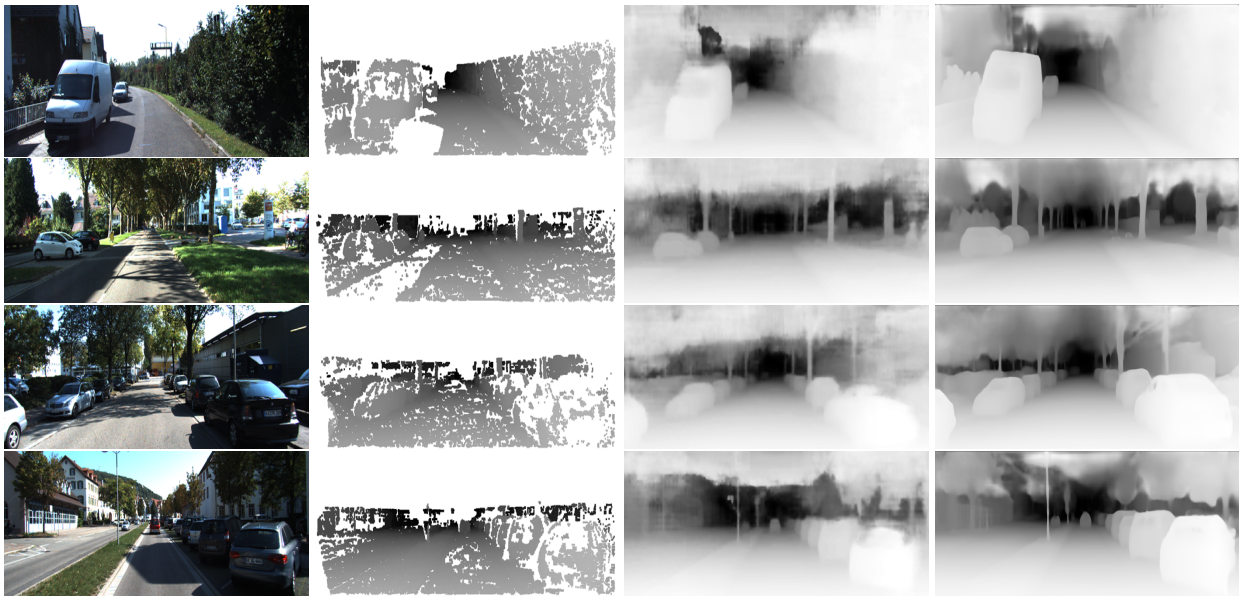
(a) Input (b) Ground truth (c) Fu et al. [13] (d) Ours

Figure 5: **Qualitative results on the NYU Depth V2 test split.** We can see much clearer object boundaries and smoother depth changes from our results.

$\frac{1}{|T|}\sum_{y\in T}||y-y^*||^2/y^*$, RMSE : $\sqrt{\frac{1}{|T|}\sum_{y\in T}||y-y^*||^2}$, RMSE*log* : $\sqrt{\frac{1}{|T|}\sum_{y\in T}||\log y-\log y^*||^2}$, log10 : $\frac{1}{|T|}\sum_{y\in T}|\log_{10} y-\log_{10} y^*|$, where T denotes a collection of pixels that the ground truth values are available, y and y^* are estimation and the ground truth value, respectively.

Using NYU Depth V2 dataset, as we can see from Table 1 the proposed method achieves a state-of-the-art result with a significant margin in both of the inlier measures (*i.e.,* $\delta_1, \delta_2, \delta_3$) and accuracy metrics (*i.e.,* AbsRel, RMSE, log10).

For evaluation using KITTI dataset, it is worth to note here that previous works use raw velodyne scan data as the ground truth for the evaluations. However, there are official post-processed ground truth depth maps recently released by KITTI. Therefore, for a more complete comparison to the state-of-the-art works, we provide the evaluation results using the raw laser scans as well as the official ground truth depth maps. In the evaluation using the official ground truth, because 45 images in Eigen's test split does not have



(a) Input (b) Ground truth (c) Fu et al. [13] (d) Ours

Figure 6: **Qualitative results on the KITTI Eigen test split.** Due to high sparsity in the ground truth depth maps, we interpolate them for visualization purpose.

corresponding ground truth, we use only the valid 652 images in this evaluation and use the whole 697 images for evaluation using the raw velodyne scans. We provide the results in Table 2. As it can be seen from the table, ours outperforms all existing works especially when evaluating with the official ground truth depth maps. We also evaluate the proposed method on the online KITTI benchmark server. At the time of submission, the proposed method (BTS) is ranked on the first place.[1]

4.5. Ablation Study

Here, we conduct evaluations with variants of our network to see effectiveness of the proposed core factors. From the baseline network which only consists of the base network, we increment the network with modules to see how the added factor improves accuracy and the result is given in Table 4. As core factors are added, the overall performance is improved, and the highest improvement is made by adding the proposed local planar guidance layers. Please note that the LPG layers only require additional 0.04M trainable parameters used by *1x1 reduction* layers. The final improvement comes from using the training loss defined in Equation 5.

4.6. Qualitative Result

Finally, we discuss about qualitative results from ours and competing works. As we can see from Figures 5 and 6, ours show much clearer object boundaries and smoother depth changes. However, in results from experiments using KITTI, we can see artifacts on sky or upper part of the outputs. We consider this as a consequence of the very sparse ground truth depth data as it can be seen from the example figures. Because there are certain regions lacking valid depth values across the dataset, the network cannot be trained properly for that regions.

[1] http://www.cvlibs.net/datasets/kitti/eval_depth.php?benchmark=depth_prediction

5. Conclusion

In this work, we have presented a supervised monocular depth estimation network and achieved state-of-the-art results. Benefiting from recent advances in deep learning, we design a network architecture that uses novel local planar guidance layers giving an explicit relation from internal feature maps to desired prediction for better training of the network. By deploying the proposed layer on multiple stages in decoding phase, we have gained a significant improvement, and shown a number of experimental results on challenging benchmarks to verify it. However, the performance gain from experiments using KITTI dataset is lower than that of using NYU Depth V2 dataset. We analyze this as an effect of the high sparsity of the ground truth. As a consequence, we will investigate adopting into our framework a photometric reconstruction loss which can provide far denser supervision to further improve the performance.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. 5

[2] A. Abrams, C. Hawley, and R. Pless. Heliometric stereo: Shape from sun position. In *Computer Vision–ECCV 2012*, pages 357–370. Springer, 2012. 1

[3] R. Basri, D. Jacobs, and I. Kemelmacher. Photometric stereo with general, unknown lighting. *International Journal of Computer Vision*, 72(3):239–257, 2007. 1

[4] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013. 5

[5] Y. Cao, Z. Wu, and C. Shen. Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 2017. 2

[6] A. Chakrabarti, J. Shao, and G. Shakhnarovich. Depth from a single image by harmonizing overcomplete local network predictions. In *Advances in Neural Information Processing Systems*, pages 2658–2666, 2016. 5

[7] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018. 2, 3

[8] W. Chen, Z. Fu, D. Yang, and J. Deng. Single-image depth perception in the wild. In *Advances in Neural Information Processing Systems*, pages 730–738, 2016. 2

[9] D.-A. Clevert, T. Unterthiner, and S. Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015. 6

[10] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 6

[11] D. Eigen and R. 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. 1, 2, 5

[12] D. Eigen, C. Puhrsch, and R. 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. 1, 2, 5, 6

[13] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. 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. 2, 3, 5, 6, 7, 8

[14] Y. Gan, X. Xu, W. Sun, and L. Lin. Monocular depth estimation with affinity, vertical pooling, and label enhancement. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 224–239, 2018. 2, 6

[15] R. Garg, V. K. BG, G. Carneiro, and I. Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *European Conference on Computer Vision*, pages 740–756. Springer, 2016. 2, 3, 6

[16] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2, 6

[17] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, volume 2, page 7, 2017. 2, 3, 6

[18] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[19] I. P. Howard. *Perceiving in depth, volume 1: basic mechanisms*. Oxford University Press, 2012. 1

[20] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *CVPR*, volume 1, page 3, 2017. 1, 5

[21] S. Kim, K. Park, K. Sohn, and S. Lin. Unified depth prediction and intrinsic image decomposition from a single image via joint convolutional neural fields. In *European conference on computer vision*, pages 143–159. Springer, 2016. 1

[22] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[23] P. Knöbelreiter, C. Reinbacher, A. Shekhovtsov, and T. Pock. End-to-end training of hybrid cnn-crf models for stereo. In *Conference on Computer Vision and Pattern Recognition,(CVPR)*, 2017. 2

[24] Y. Kuznietsov, J. Stückler, and B. Leibe. Semi-supervised deep learning for monocular depth map prediction. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6647–6655, 2017. 1, 2, 6

[25] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 239–248. IEEE, 2016. 1, 2, 5

[26] J.-H. Lee, M. Heo, K.-R. Kim, and C.-S. Kim. Single-image depth estimation based on fourier domain analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 330–339, 2018. 5

[27] B. Li, C. Shen, Y. Dai, A. Van Den Hengel, and M. He. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1119–1127, 2015. 2

[28] J. Li, R. Klein, and A. Yao. A two-streamed network for estimating fine-scaled depth maps from single rgb images. In *Proceedings of the 2017 IEEE International Conference on Computer Vision, Venice, Italy*, pages 22–29, 2017. 1, 5

[29] F. Liu, C. Shen, G. Lin, and I. D. Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(10):2024–2039, 2016. 1, 2, 5, 6

[30] R. Mahjourian, M. Wicke, and A. Angelova. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5667–5675, 2018. 3

[31] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 2, 6

[32] A. Odena, V. Dumoulin, and C. Olah. Deconvolution and checkerboard artifacts. *Distill*, 1(10):e3, 2016. 6

[33] X. Qi, R. Liao, Z. Liu, R. Urtasun, and J. 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. 5

[34] R. Ranftl, V. Vineet, Q. Chen, and V. Koltun. Dense monocular depth estimation in complex dynamic scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4058–4066, 2016. 1

[35] A. Roy and S. Todorovic. Monocular depth estimation using neural regression forest. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5506–5514, 2016. 1

[36] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 6

[37] A. Saxena, S. H. Chung, and A. Y. Ng. Learning depth from single monocular images. In *Advances in neural information processing systems*, pages 1161–1168, 2006. 1, 2

[38] A. Saxena, M. Sun, and A. Y. Ng. Make3d: Learning 3d scene structure from a single still image. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):824–840, 2009. 2, 5, 6

[39] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*, 47(1-3):7–42, 2002. 1

[40] A. G. Schwing and R. Urtasun. Fully connected deep structured networks. *arXiv preprint arXiv:1503.02351*, 2015. 2

[41] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[42] C. Wang, J. Miguel Buenaposada, R. Zhu, and S. 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. 3

[43] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. L. Yuille. Towards unified depth and semantic prediction from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2800–2809, 2015. 1, 5

[44] X. Wang, D. Fouhey, and A. Gupta. Designing deep networks for surface normal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 539–547, 2015. 2

[45] J. Xie, R. Girshick, and A. Farhadi. Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks. In *European Conference on Computer Vision*, pages 842–857. Springer, 2016. 2, 3

[46] D. Xu, E. Ricci, W. Ouyang, X. Wang, and N. Sebe. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5354–5362, 2017. 2, 5

[47] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang. Denseaspp for semantic segmentation in street scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3684–3692, 2018. 2, 3

[48] Z. Yin and J. Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1983–1992, 2018. 3