Towards Interpretable Deep Networks for Monocular Depth Estimation

Zunzhi You
Sun Yat-sen University
youzunzhi@gmail.com

Yi-Hsuan Tsai
NEC Laboratories America
ytsai@nec-labs.com

Wei-Chen Chiu
National Chiao Tung University
walon@cs.nctu.edu.tw

Guanbin Li*
Sun Yat-sen University
liguanbin@mail.sysu.edu.cn

Abstract

Deep networks for Monocular Depth Estimation (MDE) have achieved promising performance recently and it is of great importance to further understand the interpretability of these networks. Existing methods attempt to provide post-hoc explanations by investigating visual cues, which may not explore the internal representations learned by deep networks. In this paper, we find that some hidden units of the network are selective to certain ranges of depth, and thus such behavior can be served as a way to interpret the internal representations. Based on our observations, we quantify the interpretability of a deep MDE network by the depth selectivity of its hidden units. Moreover, we then propose a method to train interpretable MDE deep networks without changing their original architectures, by assigning a depth range for each unit to select. Experimental results demonstrate that our method is able to enhance the interpretability of deep MDE networks by largely improving the depth selectivity of their units, while not harming or even improving the depth estimation accuracy. We further provide comprehensive analysis to show the reliability of selective units, the applicability of our method on different layers, models, and datasets, and a demonstration on analysis of model error. Source code and models are available at https://github.com/youzunzhi/InterpretableMDE.

1. Introduction

Monocular Depth Estimation (MDE) has drawn a lot of attention since it is critical for further applications like 3D scene understanding or autonomous driving, due to the less requirement and cost compared to depth estimation using stereo image pairs. Eigen *et al*. [10] first utilize convolutional neural networks to perform MDE; since then numerous approaches based on deep neural networks have been proposed and significantly improve state-of-the-art performance [13, 18, 42, 26]. However, only few studies focus on the interpretability of these MDE networks [46]. Since depth estimation can be closely related to downstream tasks like autonomous driving, the lack of interpretability on MDE models could potentially cause critical consequences.

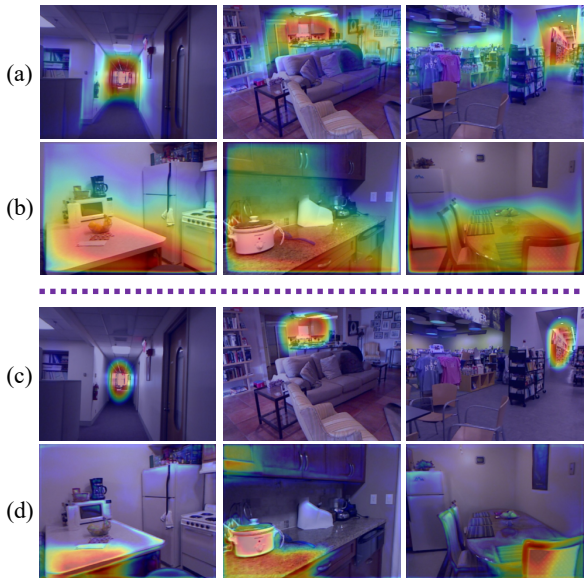


Figure 1. Visualization of feature maps. (a) and (b) refer to the feature map visualization of Unit 5 in layer MFF and Unit 26 in layer D (cf. Section 5) of [18] (ResNet-50), respectively. (c) and (d) refer to Unit 63 in layer D and Unit 0 in layer MFF of the interpretable counterpart trained by our method, respectively (best viewed in color). We show that (b) has activations over different depth ranges, while our results in (c) and (d) focus on distant or close depth, which allows more interpretability of the model.

*Corresponding author is Guanbin Li.

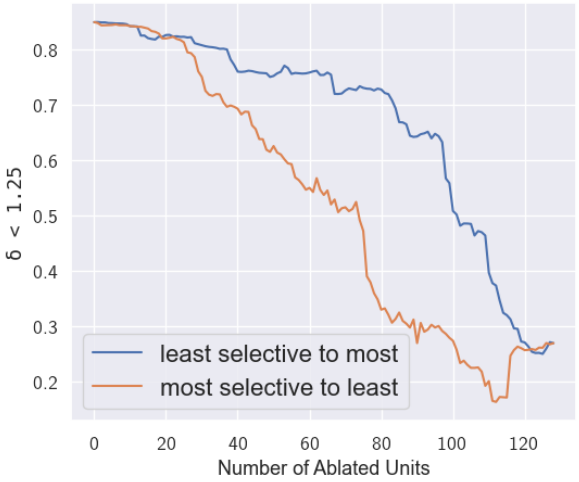


Figure 2. A comparison of the accuracy drop rate when units are ablated successively in different orders. The units are sorted by their depth selectivity and then successively ablated in two reversed order. The accuracy in the y-axis drops faster when units with the higher selectivity are ablated before the less selective ones.

In general, understanding deep networks is of great necessity. Previous works on the interpretability of deep networks for vision mainly focus on image classification [44, 2] or image generation [3]. On depth estimation, Hu *et al*. [20] and Dijk *et al*. [9] analyze how deep networks estimate depth from single images by investigating the visual cues in input images, on the level of pixels or semantics, respectively. However, they still treat the networks as black boxes, resulting in less exploration of the internal representations learned by the MDE networks. In addition, such post-hoc explanations may not present the whole story of interpretable machine learning models as discussed in [33]. Although there exists interpretable models for computer vision tasks, such as image classification [45, 5], object detection [41] or person re-identification [28], these tasks have quite a different characteristics from MDE and are not directly applicable to MDE.

Recently, numerous methods try to discover what neurons in neural networks look for [30, 2, 11, 32]. It is shown that neuron units generally extract features that can be interpreted as various levels of semantic concept, from textures and patterns to objects and scenes. Moreover, to learn interpretable neural networks, one option is to disentangle the representations learned by internal filters, which makes the filters more specialized [45, 27]. Inspired by these works, we observe that in deep MDE networks, some hidden units are selective to some ranges of depth. For example, in Fig. 1(a), we visualize several feature maps of one unit in a layer of the network from [18]. This unit is obviously more activated in the distant regions of the input images. We further dissect the units by collecting their averaged response

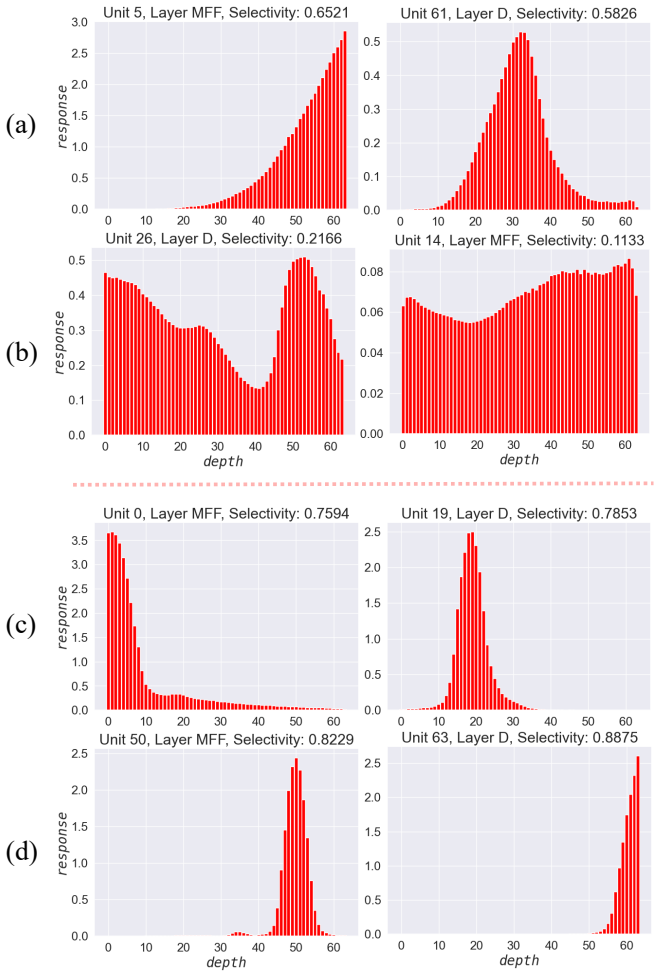


Figure 3. Dissection results on units. (a) and (b) are units of layer D and layer MFF in [18] (ResNet-50), where it shows diverse ranges of selectivity. Using our proposed interpretable model, we consistently increase the selectivity over all the units, e.g., (c) and (d), which improves the model interpretability.

on depth ranges (see Section 3.1 later for more details), and Fig. 3(a) shows that for some units, activations are higher for some certain ranges of depth.

To quantify this observation, we then compute its depth selectivity for each unit (detailed in Section 3.2). To evaluate the meaningfulness of depth selectivity, we successively ablate units and see how the performance of the network drops accordingly. We first sort the 128 units of the network from [18] by their selectivity and then successively ablate units from the most selective unit to the least one, and then do the same thing similarly in the reversed way. In Fig. 2, the performance of the MDE model drops much quicker when more selective units are ablated earlier than less selective ones. Based on the observations stated above, we argue that for an MDE deep network, a unit is more important when it is more depth selective, and the behavior of its units can be interpreted by telling which ranges of depth

activated most by those units. Therefore, the interpretability of a deep network for MDE can be quantified by the depth selectivity of its internal units.

However, in the existing MDE model, despite that some units can be interpreted as being selective for some ranges of depth, most of them have little interpretability. For example, Fig. 1(b) and Fig. 3(b) show feature map visualizations and dissection results of typical units in the network from [18], which have less interpretability. Therefore, to achieve an MDE model with better interpretability, we propose a simple yet effective interpretable deep network for MDE by maximizing the selectivity of internal units. Our method can be applied to existing deep MDE networks without modifying their original architectures or requiring any additional annotations. More importantly, we show that it is possible to learn our interpretable model without harming its depth performance, which creates potential discussions in explainable AI along the trade-off between interpretability and model performance [34]. The experimental results show that the our interpretable models achieve competitive or even better performance than the original MDE models, while the interpretability is largely improved.

Contributions. To summarize, this work has the following contributions: (1) we quantify the interpretability of deep networks for MDE based on the depth selectivity of models' internal units; (2) we propose a novel method to learn interpretable deep networks for MDE without modifying the original network's architecture or requiring any additional annotations; (3) we empirically show that our method effectively improves the interpretability of deep MDE networks, while not harming or even improving the depth accuracy, and further validate the reliability and applicability of the proposed method.

2. Related Work

2.1. Monocular Depth Estimation

Estimating depth from images is an important problem towards scene understanding, and recently monocular depth estimation has been studied extensively. Numerous methods based on deep convolutional neural networks have been proposed to achieve better performance on this task, including the usage of geometric constraints, adopting multi-scale network architecture, or sharing features with semantic segmentation [25, 13, 18, 42, 22, 26, 48]. Nevertheless, few studies analyze what these deep networks have learned. By modifying input images, Dijk *et al*. [9] investigate the visual cues of what a network [15] exploits when predicting the depth. Hu *et al*. [20] hypothesize that deep networks can estimate depth from only a selected set of image pixels fairly accurately, and train another network to predict those pixels. Despite that some of their findings are interesting and useful to help understand deep networks for MDE, they still treat the networks as black boxes and their post-hoc explanations do not lead to inherently interpretable models.

2.2. Post-hoc Explanations for Deep Networks

Recently, many studies aim to explain deep networks in a post-hoc fashion. Among them, a line of research can be categorized into saliency methods or attribution methods, where the "important" pixels are highlighted in input images for networks to give their predictions [43, 35, 37, 24, 40]. While some recent studies discuss their reliability [23, 38, 1], these methods are not directly applicable to the task of MDE, since MDE is required to predict a depth value for every pixel, and thus it is not reasonable to use highlighted pixels to attribute the dense prediction of all pixels.

Another group of studies on interpretability of deep neural networks explore the properties or the behavior of single units [43, 47, 2, 30, 31, 29, 3, 32], where our work generally falls in this group as we quantify the interpretability of networks for MDE. The fundamental difference between the task of MDE and image classification makes our work distinct from theirs. Moreover, these methods still focus on the explanations of deep networks, instead of designing interpretable models.

2.3. Interpretable Deep Networks for Vision

Instead of providing explanations, some studies attempt to design inherently interpretable models to alleviate the lack of model interpretability in computer vision tasks. Chen *et al*. [5] propose an interpretable model for object recognition that finds prototypical parts and reasons from them to make final decisions. Liao *et al*. [28] propose an approach to enhance the interpretability of person re-identification networks by making the matching process of feature maps explicit. Moreover, other methods that share a similar concept to our method are to learn more specialized filters. In interpretable CNNs from [45], each filter represents a specific object part, while a more recent study [27] trains interpretable CNNs by alleviating filter-class entanglement, i.e. each filter only responds to one or few classes. In this paper, our proposed interpretable model focuses on the MDE task by increasing the depth selectivity of units internally in MDE models, which differs from the aforementioned approaches.

3. Interpretability of Deep Networks for MDE

In this section, we present how we quantify the interpretability of the units by calculating their depth selectivity with their average response on different ranges of depth.

3.1. Average Response of Units on Depth

We first dissect a deep network for MDE by collecting the average response of its units on depth. Denote images

and the corresponding depth maps in a depth dataset D as $(\mathbf{x}_i, \mathbf{d}_i) \in D$, where $i \in \{1, 2, ..., N\}$ and N is the number of samples in D. For every internal unit k in a layer l of the deep network, the activation map $A_{l,k}(\mathbf{x}_i)$ is scaled up to the resolution of depth map using bilinear interpolation, denoted as $\tilde{A}_{l,k}(\mathbf{x}_i)$. Depth values in $\mathbf{d}_i$ can be discretized into N_b bins to capture the meaningful depth distribution. Then, for every discretized depth value d (*i.e.*, the index of a bin) in the discretized depth map $\hat{\mathbf{d}}_i$, we can obtain a binary mask M_i^d calculated by $\mathbb{I}(\hat{\mathbf{d}}_i = d)$, where $\mathbb{I}(\cdot)$ is the indicator function. The average response $R_{l,k}^d$ of unit k in layer l for depth d is then computed over the entire dataset:

$$R_{l,k}^d = \frac{\sum_{i=1}^N S(\tilde{A}_{l,k}(\mathbf{x}_i) \odot M_i^d)}{\sum_{i=1}^N S(M_i^d)}, \tag{1}$$

where $S(\cdot)$ sums over all the elements of a matrix and $\odot$ denotes the element-wise multiplication.

3.2. Depth Selectivity

Based on the average response, we compare how each unit is activated by different depth ranges and observe that some units are selective to a certain range of depth. Inspired by the commonly-used selectivity index in systems neuroscience [8, 4, 12], Morcos *et al*. [29] propose a metric to calculate the class-selectivity of a unit based on its class-conditional average activity, for the task of image classification. Here we adopt this metric to the domain of depth estimation. We define the depth selectivity of a unit as:

$$DS_{l,k} = \frac{|R_{l,k}^{max}| - |\bar{R}_{l,k}^{-max}|}{|R_{l,k}^{max}| + |\bar{R}_{l,k}^{-max}|}, \tag{2}$$

where $|R_{l,k}^{max}|$ is the absolute value of the max response of unit k in layer l over all discretized depth d, and $|\bar{R}_{l,k}^{-max}|$ is the average of all the other non-maximum absolute responses. We use the absolute value to make it applicable for units that may have negative output (e.g., units that use ELU [7] as the activation function). The value of DS is in the range $[0, 1]$, and a DS value close to 1 indicates that corresponding unit is highly selective (e.g., Fig. 3(c)(d)). To give a more concrete idea about this quantity, we calculate its expectation when unit's response is totally randomized (see supplementary material for the derivation).

$$\mathbb{E}_{|R_{l,k}^d|}[DS_{l,k}] = \frac{1}{3}, \quad |R_{l,k}^d| \sim U[0, b], \tag{3}$$

where b is an arbitrary positive number as the upper bound of $|R_{l,k}^d|$, in which its value would not affect the outcome of the expectation. This expectation can be considered as a random baseline to be further compared with the depth selectivity of actual MDE networks.

4. Interpretable Deep Networks for MDE

As motivated previously, here we would like to consider an important problem: *Is it possible to enhance the interpretability of an MDE deep network without modifying its architecture and harming its performance?* In this section, we first present a naïve thought (i.e., regularizing selectivity) together with pointing out its potential issue, and then describe our proposed approach (i.e., assigning depth ranges to units).

4.1. Regularizing Selectivity

As we have the metric of depth selectivity to quantify the interpretability, we in turn aim to enhance the interpretability of an MDE network by increasing its depth selectivity. A straightforward approach that first comes to our mind is adding an additional regularization term $\mathcal{L}_{reg}$ to the objective of the MDE model, which encourages the depth selectivity of all the units in layer $l \in L$ to increase:

$$\begin{aligned} \mathcal{L}_{reg} &= -\lambda \sum_{l \in L} \frac{1}{K_l} \sum_k DS_{l,k} \\ &= -\lambda \sum_{l \in L} \frac{1}{K_l} \sum_k \frac{|R_{l,k}^{max}| - |\bar{R}_{l,k}^{-max}|}{|R_{l,k}^{max}| + |\bar{R}_{l,k}^{-max}|}, \end{aligned} \tag{4}$$

where K_l is the number of units in layer l, and $\lambda > 0$ is a hyperparameter to balance between the original depth estimation loss and our regularization term of depth selectivity.

However, we experimentally find that such naïve approach leads to unsatisfactory results. Fig. 4 shows the dissection results of some units in the network trained via regularizing the depth selectivity. Despite that some units are still depth selective as we expect, many others are not or collapse (i.e., having no response to any depth values). This is due to the fact that, during the process of batch-wise optimization, the discretized depths that activate units are mostly different within each batch. Here can be two reasons: (1) at the beginning of training, units are not selective at all, and (2) even if a unit is depth selective, the selected depth could be absent in a batch, and then the unit will be encouraged to activate more on the other depth ranges (e.g., focusing on the range that it activates most in this batch).

4.2. Assigning Depth to Units

In order to tackle the above-mentioned issue happened while regularizing the depth selectivity, we propose a simple yet effective method by assigning each unit a specific depth range for it to select, which is realized by an objective function $\mathcal{L}_{assign}$:

$$\mathcal{L}_{assign} = -\lambda \sum_{l \in L} \frac{1}{K_l} \sum_k \frac{|R_{l,k}^{d_k}| - |\bar{R}_{l,k}^{-d_k}|}{|R_{l,k}^{d_k}| + |\bar{R}_{l,k}^{-d_k}|}, \tag{5}$$

Table 1. Depth selectivity and performance of baseline networks for MDE from [18] and our interpretable counterparts.

Model	Training	Testing	$\delta_{1.25}$	$\delta_{1.25^2}$	$\delta_{1.25^3}$	RMS	REL	log10
[18] (ResNet-50)	0.4617	0.4286	0.849	0.972	**0.994**	0.443	0.124	0.054
Interpretable [18] (ResNet-50)	**0.8357**	**0.7529**	**0.861**	**0.973**	**0.994**	**0.422**	**0.119**	**0.051**
[18] (SENet-154)	0.4906	0.4691	0.874	**0.979**	**0.995**	0.409	0.111	0.049
Interpretable [18] (SENet-154)	**0.8411**	**0.7693**	**0.882**	**0.979**	**0.995**	**0.396**	**0.109**	**0.047**

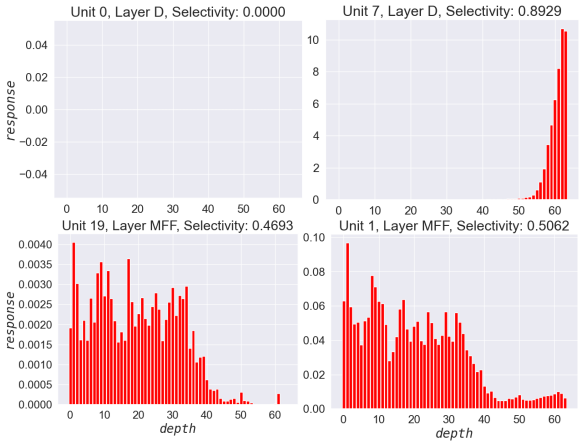


Figure 4. Dissection results of typical units in networks trained by the approach of directly regularizing the depth selectivity via (4).

where d_k is the discretized depth being assigned to unit k. As a result, the calculation of selectivity of a unit k is now based on the assigned discretized depth d_k, where $|\bar{R}_{l,k}^{-d_k}|$ is the average of all other absolute responses other than d_k. The assignment of depth range to units is based on the following principle:

$$d_k = \lfloor \frac{k}{K_l / N_b} \rfloor, \tag{6}$$

where the number of depth bins N_b is set to K_l if $K_l \leq N_b$, such that every discretized depth d is assigned to at least one unit. If d_k is absent in a batch, the unit k will be simply disregarded from the computation of $\mathcal{L}_{assign}$. As a result, this approach does not suffer from problems caused by batch sampling. Moreover, the interpretability of the deep network is enhanced from another perspective: the behavior of a unit becomes interpretable and predictable as it is now specifically assigned to a particular depth. Please note that in the following sections and experiments, such proposed approach of assigning depth to units is abbreviated to "our method" unless otherwise stated.

5. Experimental Results

For simplicity, we follow the choice in [20] and use the network proposed in [18] as our target model on the NYUD-V2 dataset [36] to show experimental results of our method. We first choose the layer after the multi-scale feature fusion module (referred as layer "MFF") and after the decoder module (referred as layer "D") in [18] to apply our method, as these two layers are closer to the depth output. Nevertheless, we also demonstrate that our approach can be applied to other layers, models, and datasets, and validate the applicability of our method in Section 5.3. For networks from [18], we consider two variants with different backbones, i.e., ResNet-50 [17] and SENet-154 [19]. During training, we follow exactly the same training scheme with the original implementation, including data augmentation, optimizers, total training epochs, etc.

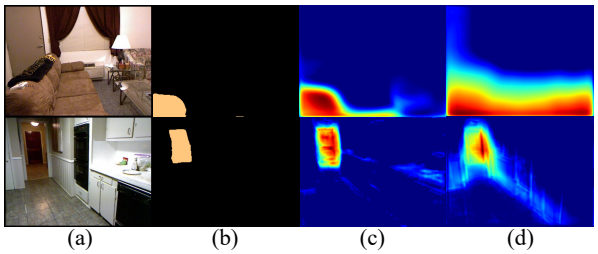


Figure 5. Comparison of the feature maps from our interpretable model and the baseline model [18]. (a) Input images. (b) Mask of pixels whose predicted depth is assigned to the corresponding units. (c) Feature maps of our selective units. (d) Feature maps of units in the baseline.

We set the number of discretized depth bins N_b to 64, since the number of units in most of deep networks for MDE is a power of two, which enables simpler assignment of depth to units. Space-increasing discretization proposed in [13] is adopted to discretize the depth maps, and λ in (5) is set to 0.1.

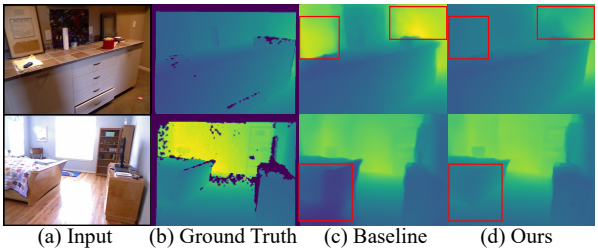
5.1. Depth Selectivity and Performance

Setting and Evaluation Metric. First, we conduct experiments to compare the depth selectivity and performance of the baseline models with our interpretable counterparts. We calculate depth selectivity on both training and testing datasets. For depth estimation performance, we follow previous works on MDE to use the following metrics: accuracy under threshold ($\delta_i < 1.25^i, i = 1, 2, 3$), root mean squared error (RMS), mean absolute relative error (REL) and mean log_{10} error (log10).

Main Results. It is first observed in Table 1 that depth selectivity of baseline models is above the random baseline, $1/3$, indicating that MDE deep networks have some level of

Table 2. Comparison of direct regularizing selectivity (cf. Section 4.1) and assigning depth to units (cf. Section 4.2).

Model	Method	Selectivity ↑		Depth Accuracy ↑			Depth Error ↓		
		Training	Testing	$\delta_{1.25}$	$\delta_{1.25^2}$	$\delta_{1.25^3}$	RMS	REL	log10
[18] (ResNet-50)	$\mathcal{L}_{reg}$ in (4)	0.7417	0.6039	0.857	**0.973**	0.993	0.428	0.121	0.052
	$\mathcal{L}_{assign}$ in (5)	**0.8357**	**0.7529**	**0.861**	**0.973**	**0.994**	**0.422**	**0.119**	**0.051**
[18] (SENet-154)	$\mathcal{L}_{reg}$ in (4)	0.7314	0.5694	0.881	0.978	**0.995**	0.399	**0.109**	**0.047**
	$\mathcal{L}_{assign}$ in (5)	**0.8411**	**0.7693**	**0.882**	**0.979**	**0.995**	**0.396**	**0.109**	**0.047**



(a) Input (b) Ground Truth (c) Baseline (d) Ours

Figure 6. Qualitative comparison of predicted depth maps by our interpretable model and the baseline model [18]. The red boxes highlight the difference of the two models.

depth selectivity in their units, while our interpretable models achieve much higher depth selectivity on both training and testing datasets. Fig. 1 (c)(d) also visualize the feature maps of some units in our interpretable networks. Qualitatively, it is shown that these units are more activated on the regions of input images where the depth is assigned to them based on their indices, e.g., distant or close regions. In Fig. 3 (c)(d), we further plot the dissection results of some units in our interpretable networks, showing the selectivity is consistent through the entire dataset.

Fig. 5 further shows some example comparisons of feature maps. To demonstrate the effectiveness of our method, pixels are highlighted if its predicted depth is assigned to the corresponding units. We observe that our model has feature maps that are more consistent with pixels of the corresponding depth, showing better interpretability. From these quantitative and qualitative results, we conclude that our method is able to significantly improve the interpretability of deep networks for MDE. Meanwhile, across all depth prediction metrics in Table 1, our interpretable models are competitive or even outperform the baseline counterparts, showing that it is possible to enhance the interpretability of an MDE deep network without harming its accuracy. Fig. 6 provides some qualitative comparisons of depth predictions between our interpretable model and the baseline.

Direct Regularizing versus Assigning. We quantitatively compare our method of assigning depth to units (cf. Section 4.2) with the direct approach of regularizing the depth selectivity (c.f Section 4.1). As shown in Table 2, although models trained via directly regularizing the selectivity achieve comparable performance with those trained by our assigning approach, their depth selectivity is much lower due to the issue stated in Section 4.1.

Table 3. Performance evaluation before and after correction, where R50 and S154 denote ResNet-50 and SENet-154, respectively (cf. Section 5.2). In each result, we indicate the performance change from the model without correction to the one after correction using the $\rightarrow$ symbol.

Model	$\delta_{1.25}$ ↑	RMS ↓
[18] (R50)	0.849 $\rightarrow$ 0.779	0.443 $\rightarrow$ 0.582
Interpretable [18] (R50)	0.861 $\rightarrow$ 0.947	0.422 $\rightarrow$ 0.362
[18] (S154)	0.874 $\rightarrow$ 0.856	0.409 $\rightarrow$ 0.466
Interpretable [18] (S154)	0.882 $\rightarrow$ 0.927	0.396 $\rightarrow$ 0.354

5.2. Reliability of Selective Units via Correction

We further design an experiment to validate the reliability of the selective units. Considering a case where internal units are selective but have zero or little effect to the final output of the model, these interpretable units do not enhance the interpretability of the entire model. Previous works evaluate the importance of units by ablation, but it is shown that there is only little impact to the accuracy of the model [29] when one-by-one ablating each unit. Here, we propose another method to evaluate the reliability of these units, by *correcting* the units instead of ablating them.

Fig. 7 illustrates the process of the correction. Here, we define the correct response of a unit for a pixel as its average response in the training data on that pixel's ground truth depth. To be specific, the ground truth depth map is resized to the size of feature maps of a unit using nearest interpolation. Then, for every pixel of the feature map, its value is corrected based on its corresponding ground truth depth and its average response collected from training data.

Table 3 shows performance evaluation before and after we conduct the correction operation. It is shown that the performance of our models is largely improved after the units' response is corrected, which indicates that units are responsible for the final prediction of the network. Furthermore, our interpretable models demonstrate larger improvement compared to the gain on baseline models using the same correction method. One reason is that our models are more depth-selective than baseline models, such that the average response contains more information that is related to depth. We also note that, the purpose of this evaluation is to validate the reliability of our interpretable units and their effect to the final depth prediction, where the ground truth depth maps are used to achieve such verification but not used in real testing.

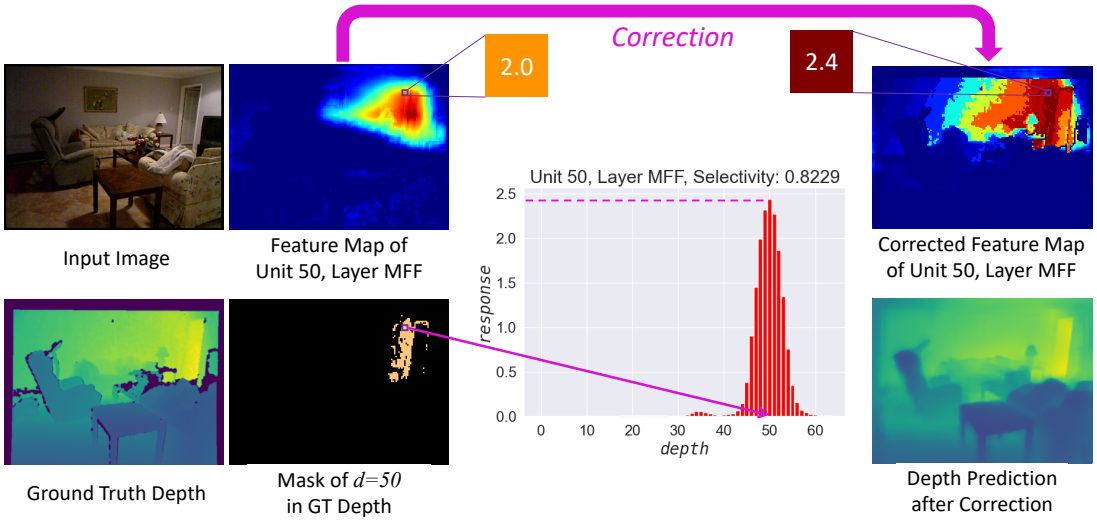


Figure 7. Illustration of our correction operation (cf. Section 5.2). Suppose a pixel of a unit's feature map has the value of 2.0, and its corresponding depth ground truth is 50 after being discretized. From dissection, we know the correct response for this unit on depth 50 is 2.4, so we can get a new depth prediction map after the response is corrected.

Table 4. Selectivity comparisons on different layers of [18].

Model	Layer	Selectivity ↑ (base) Training	Selectivity ↑ (base) Testing	Selectivity ↑ (ours) Training	Selectivity ↑ (ours) Testing
[18] (R50)	D&MFF	0.4617	0.4286	**0.8357**	**0.7529**
	Rconv0	0.4877	0.4531	**0.7608**	**0.6846**
	Rconv1	0.4712	0.4399	**0.7436**	**0.6701**
[18] (S154)	D&MFF	0.4906	0.4691	**0.8411**	**0.7693**
	Rconv0	0.5306	0.5068	**0.7582**	**0.6945**
	Rconv1	0.4404	0.4095	**0.7217**	**0.6626**

Table 5. Depth estimation performance of applying our method on different layers of [18]. Note that the first row of each model (denoted as "-" in Layer) shows the performance of the original baseline model.

Model	Layer	$\delta_{1.25}$ ↑	RMS ↓	REL ↓	log10 ↓
[18] (R50)	-	0.849	0.443	0.124	0.054
	D&MFF	0.861	0.422	0.119	0.051
	Rconv0	0.860	0.423	0.119	0.051
	Rconv1	0.862	0.423	0.119	0.051
[18] (S154)	-	0.874	0.409	0.111	0.049
	D&MFF	0.882	0.396	0.109	0.047
	Rconv0	0.881	0.396	0.110	0.047
	Rconv1	0.883	0.395	0.108	0.047

5.3. Applicability of Our Method

More Results on Layers, Models, and Datasets. We further apply our method on different layers, models, and datasets to explore its effectiveness. For networks from [18], we consider layers after the first and second convolutional layers in the refine module (referred as layer "Rconv0" and "Rconv1"). Table 4 and Table 5 show that for all different layers, our method improves the interpretability (selectivity) over baseline models, while these interpretable models perform competitively in depth estimation accuracy. We further consider the current state-of-the-art model from [26] with the backbone of DenseNet-161 [21] using its four layers, i.e., the layer before the final convolutional layer, the first, second and third upconv layer nearest to the final output (referred as "iconv1", "upconv1", "upconv2", "upconv3" following the definition in the supplementary material of the original paper). We also provide experimental results on another commonly-used dataset in outdoor environment, i.e. KITTI [14]. We show the results of selectivity and depth estimation accuracy in Table 7 and Table 8[1], which validate that our approach is applicable to these various models on another dataset.

Application in Depth Completion. To show the applicability of our interpretable model, we conduct experiments to apply our method on the monocular depth completion model. Monocular depth completion is a task highly related to monocular depth estimation, while it additionally takes sparse depth pixels acquired from depth sensors (e.g. LiDAR) or with ground truth depth values as the condition for solving the scale ambiguities and improving the performance of depth estimation. Here we select CSPN [6] as our target model. Following their original paper, we adopt the evaluation metrics including accuracy under threshold ($\delta_i < t, t \in \{1.02, 1.05, 1.10, 1.25, 1.25^2, 1.25^3\}$), RMS, and REL. Table 6 shows that our approach works well on this model for depth completion, while providing a much

[1]For KITTI, we use an improved set of ground truth depth maps provided by [39] to train and evaluate both the baseline and our models, so the performance is better than the reported one in the original paper [26].

(a) (b) (c) (d) (e) (f)

Figure 8. Demonstration of explaining the prediction error on adversarial samples. (a) An adversarial image. (b) Ground truth depth map (c) Predicted depth map. (d) Pixels whose δ error is above 1.25. (e) Pixels whose predicted depth is within the 11^{th} depth bin. (f) Feature map of Unit 11 in layer MFF.

Table 6. Depth selectivity and performance of the monocular depth completion model CSPN [6] and our interpretable counterpart.

Model	Selectivity ↑		Depth Accuracy ↑						Depth Error ↓	
	Training	Testing	$\delta_{1.02}$	$\delta_{1.05}$	$\delta_{1.10}$	$\delta_{1.25}$	$\delta_{1.25^2}$	$\delta_{1.25^3}$	RMS	REL
CSPN	0.4022	0.4213	0.832	0.934	0.971	**0.992**	**0.999**	**1.000**	**0.117**	0.016
Interpretable CSPN	**0.9394**	**0.9475**	**0.860**	**0.948**	**0.976**	**0.992**	0.998	0.999	0.119	**0.015**

Table 7. Selectivity of applying our method on different layers of [26] on the dataset of NYUD-V2 [36] and KITTI [14].

Dataset	Layer	Selectivity ↑ (base)		Selectivity ↑ (ours)	
		Training	Testing	Training	Testing
NYUD-V2	iconv1	0.5202	0.3799	**0.8580**	**0.7667**
	upconv1	0.6763	0.5507	**0.9117**	**0.8072**
	upconv2	0.5271	0.4262	**0.9051**	**0.7929**
	upconv3	0.5476	0.4390	**0.7981**	**0.7434**
KITTI	iconv1	0.5000	0.4319	**0.8321**	**0.8000**
	upconv1	0.7128	0.5988	**0.8935**	**0.8658**
	upconv2	0.4919	0.4181	**0.8896**	**0.8616**
	upconv3	0.5192	0.4795	**0.8053**	**0.7893**

better selectivity.

Analysis of Model Error. Our interpretable model has the advantage of providing a cue to explain why the model makes mistakes. Here we show this application by analyzing the internal representations when predicting on adversarial samples in Fig. 8. We first generate adversarial samples by commonly used white-box attacks FGSM [16] ($\epsilon = 0.05$). As expected, the prediction is not as accurate as before (δ_1 dropped to 0.488 from 0.843). When looking into the mistakes of the prediction, which can be defined as pixels whose δ error is above 1.25 (*i.e.*, not counted in $\delta_{1.25}$), we find that the predicted depth of a large portion of errors is caused by the 11^{th} depth bin. As we trace back to the neurons, the feature map of Unit 11 shows that the unit is activated on the region with errors, well explaining why those mistakes have been made. The process allows developers and users to know why the model gives unsatisfactory predictions, making the model more trustworthy.

6. Conclusions

In this paper, we propose to investigate the interpretability of deep networks for monocular depth estimation via exploring their internal representations, and advance to make the networks more interpretable. We first find that some hidden units in deep networks for MDE are selective to certain ranges of depth, which inspires us to quantify the interpretability of these networks as the depth selectivity of their internal units. Furthermore, we propose a simple yet effective method that is applicable to existing deep MDE networks without modifying their original architectures or requiring any additional annotations, showing that it is possible to largely improve the interpretability while not harming or even improving the depth estimation accuracy. Experimental results demonstrate the effectiveness, reliability and applicability of our method. In the future work, we will extend our studies on the behavior of internal units in MDE networks to other concepts, such as occlusion boundary, surface normal, and semantics, aiming for a more comprehensive quantification of interpretability and better understanding of deep networks for MDE.

Table 8. Depth estimation performance of applying our method on different layers of [26] on the dataset of NYUD-V2 [36] and KITTI [14].

Dataset	Layer	$\delta_{1.25}$ ↑	RMS ↓	REL ↓	log10 ↓
NYUD-V2	-	0.885	0.392	0.110	0.047
	iconv1	0.882	0.389	0.110	0.047
	upconv1	0.882	0.388	0.110	0.047
	upconv2	0.880	0.392	0.111	0.047
	upconv3	0.882	0.392	0.110	0.047
KITTI	-	0.963	2.430	0.056	0.025
	iconv1	0.961	2.435	0.059	0.026
	upconv1	0.959	2.477	0.059	0.026
	upconv2	0.960	2.436	0.059	0.026
	upconv3	0.960	2.415	0.058	0.026

Acknowledgement. This paper is supported in part by the National Natural Science Foundation of China under Grant No.61976250 and No.U1811463, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant No.2020B1515020048 and in part by MOST 110-2636-E-009-001, Taiwan. We are also grateful to the National Center for High-performance Computing, Taiwan, for providing computing services and facilities.

References

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 3

[2] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 3

[3] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B. Tenenbaum, William T. Freeman, and Antonio Torralba. GAN dissection: Visualizing and understanding generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2019. 2, 3

[4] Kenneth H Britten, Michael N Shadlen, William T Newsome, and J Anthony Movshon. The analysis of visual motion: a comparison of neuronal and psychophysical performance. *Journal of Neuroscience*, 1992. 4

[5] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: deep learning for interpretable image recognition. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 2, 3

[6] Xinjing Cheng, Peng Wang, and Ruigang Yang. Depth estimation via affinity learned with convolutional spatial propagation network. In *European Conference on Computer Vision (ECCV)*, 2018. 7, 8

[7] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). In *International Conference on Learning Representations (ICLR)*, 2016. 4

[8] Russell L De Valois, E William Yund, and Norva Hepler. The orientation and direction selectivity of cells in macaque visual cortex. *Vision Research*, 1982. 4

[9] Tom van Dijk and Guido de Croon. How do neural networks see depth in single images? In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 2, 3

[10] 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 (NeurIPS)*, 2014. 1

[11] Ruth Fong and Andrea Vedaldi. Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[12] David J Freedman and John A Assad. Experience-dependent representation of visual categories in parietal cortex. *Nature*, 2006. 4

[13] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 3, 5

[14] 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. 7, 8

[15] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[16] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 8

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 5

[18] Junjie Hu, Mete Ozay, Yan Zhang, and Takayuki Okatani. Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019. 1, 2, 3, 5, 6, 7

[19] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 5

[20] Junjie Hu, Yan Zhang, and Takayuki Okatani. Visualization of convolutional neural networks for monocular depth estimation. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 2, 3, 5

[21] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 7

[22] Jianbo Jiao, Ying Cao, Yibing Song, and Rynson Lau. Look deeper into depth: Monocular depth estimation with semantic booster and attention-driven loss. In *European Conference on Computer Vision (ECCV)*, 2018. 3

[23] Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T. Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. The (un)reliability of saliency methods. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, volume 11700 of *Lecture Notes in Computer Science*, pages 267–280. Springer, 2019. 3

[24] Pieter-Jan Kindermans, Kristof T. Schütt, Maximilian Alber, Klaus-Robert Müller, Dumitru Erhan, Been Kim, and Sven Dähne. Learning how to explain neural networks: Patternnet and patternattribution. In *International Conference on Learning Representations (ICLR)*, 2018. 3

[25] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *International Conference on 3D Vision (3DV)*, 2016. 3

[26] 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:1907.10326*, 2019. 1, 3, 7, 8

[27] Haoyu Liang, Zhihao Ouyang, Yuyuan Zeng, Hang Su, Zihao He, Shu-Tao Xia, Jun Zhu, and Bo Zhang. Training interpretable convolutional neural networks by differentiating class-specific filters. In *European Conference on Computer Vision (ECCV)*, 2020. 2, 3

[28] Shengcai Liao and Ling Shao. Interpretable and generalizable person re-identification with query-adaptive convolution and temporal lifting. In *European Conference on Computer Vision (ECCV)*, 2020. 2, 3

[29] Ari S. Morcos, David G. T. Barrett, Neil C. Rabinowitz, and Matthew Botvinick. On the importance of single directions for generalization. In *International Conference on Learning Representations (ICLR)*, 2018. 3, 4, 6

[30] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization, 2017. https://distill.pub/2017/feature-visualization. 2, 3

[31] Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. The building blocks of interpretability, 2018. https://distill.pub/2018/building-blocks. 3

[32] Ivet Rafegas, Maria Vanrell, Luís A Alexandre, and Guillem Arias. Understanding trained cnns by indexing neuron selectivity. *Pattern Recognition Letters*, 2020. 2, 3

[33] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 2019. 2

[34] Cynthia Rudin and Joanna Radin. Why are we using black box models in ai when we don't need to? a lesson from an explainable ai competition. *Harvard Data Science Review*, 2019. 3

[35] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 3

[36] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision (ECCV)*, 2012. 5, 8

[37] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In Doina Precup and Yee Whye Teh, editors, *International Conference on Machine Learning (ICML)*, 2017. 3

[38] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning (ICML)*, 2017. 3

[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. 7

[40] Yulong Wang, Hang Su, Bo Zhang, and Xiaolin Hu. Learning reliable visual saliency for model explanations. *IEEE Transactions on Multimedia (TMM)*, 2019. 3

[41] Tianfu Wu and Xi Song. Towards interpretable object detection by unfolding latent structures. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 2

[42] Wei Yin, Yifan Liu, Chunhua Shen, and Youliang Yan. Enforcing geometric constraints of virtual normal for depth prediction. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 1, 3

[43] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (ECCV)*, 2014. 3

[44] Quanshi Zhang, Ruiming Cao, Feng Shi, Ying Nian Wu, and Song-Chun Zhu. Interpreting CNN knowledge via an explanatory graph. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *AAAI Conference on Artificial Intelligence (AAAI)*, 2018. 2

[45] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 3

[46] Chaoqiang Zhao, Qiyu Sun, Chongzhen Zhang, Yang Tang, and Feng Qian. Monocular depth estimation based on deep learning: An overview. *Science China Technological Sciences*, 2020. 1

[47] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene cnns. In Yoshua Bengio and Yann LeCun, editors, *International Conference on Learning Representations (ICLR)*, 2015. 3

[48] Shengjie Zhu, Garrick Brazil, and Xiaoming Liu. The edge of depth: Explicit constraints between segmentation and depth. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3