SDC-Depth: Semantic Divide-and-Conquer Network for Monocular Depth Estimation

Lijun Wang[1], Jianming Zhang[2], Oliver Wang[2], Zhe Lin[2], and Huchuan Lu[1,3] *
[1]Dalian University of Technology, [2]Adobe Research, [3]Peng Cheng Lab
{ljwang,lhchuan}@dlut.edu.cn, {jianmzha,owang,zlin}@adobe.com

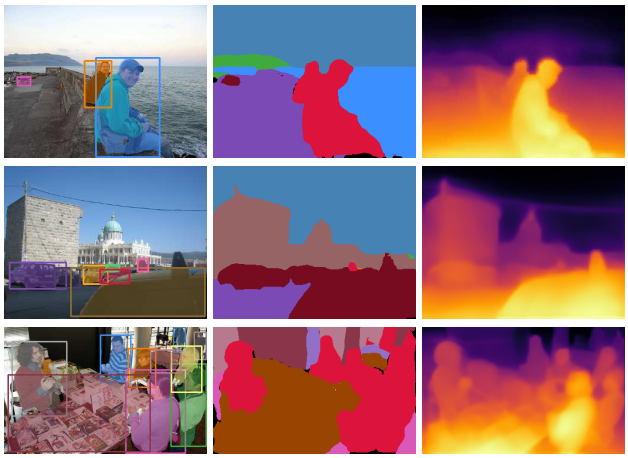
Abstract

Monocular depth estimation is an ill-posed problem, and as such critically relies on scene priors and semantics. Due to its complexity, we propose a deep neural network model based on a semantic divide-and-conquer approach. Our model decomposes a scene into semantic segments, such as object instances and background stuff classes, and then predicts a scale and shift invariant depth map for each semantic segment in a canonical space. Semantic segments of the same category share the same depth decoder, so the global depth prediction task is decomposed into a series of category-specific ones, which are simpler to learn and easier to generalize to new scene types. Finally, our model stitches each local depth segment by predicting its scale and shift based on the global context of the image. The model is trained end-to-end using a multi-task loss for panoptic segmentation and depth prediction, and is therefore able to leverage large-scale panoptic segmentation datasets to boost its semantic understanding. We validate the effectiveness of our approach and show state-of-the-art performance on three benchmark datasets.

1. Introduction

Depth estimation is an important component of 3D perception. Compared to reconstruction techniques based on active sensors or multi-view geometry, monocular depth estimation is significantly more ill-posed, and is therefore critically reliant on learning strong scene priors and semantics.

Recent works studying this problem [4, 14, 39] have achieved significant progresses using deep convolutional neural networks (CNNs) supervised by depth data, showing that they are able to capture complex high-level scene semantics. In addition, some works [39, 28] further feed semantic segmentation labels to their models to boost depth estimation accuracy in some specific domains. However, monocular depth estimation in the wild remains challenging due to the diversity of real world scenes.



(a) (b) (c)

Figure 1: Our depth prediction method jointly decomposes an input image into both instance (a) and category segments (b). It then independently predicts depth in a canonical space for each segment and recomposes them into a final globally coherent depth map (c). Note that the depth maps are generated by our SDC-Depth Net trained with sparse point-level depth order supervision [2].

We propose a Semantic Divide-and-Conquer Network (SDC-Depth Net) for monocular depth estimation. We decompose a natural image into a number of semantic segments, and then predict, for each segment, a normalized depth map in the range $[0, 1]$. We refer to this normalized depth map as the *canonical* depth for a given segment. This decomposition simplifies the depth prediction problem, as semantic categories have much consistent depth structures when viewed in isolation, and training category-specific depth decoders makes it easier to learn these priors. For example, the sky region is always infinitely far away and depth in the ground region often varies smoothly along the vertical direction. For object categories like people, instance-level depth maps also have a high degree of similarity to each

*Corresponding author.

other. Once we predict the canonical depth for each semantic segment, we assemble the results together based on a global context derived from the input image. Specifically, our model predicts the scale and shift for each segment's depth using the global context. The model is trained fully end-to-end using a multi-task loss for segmentation and depth estimation, for which we can use separate datasets to increase the diversity of our supervision. Figure 1 demonstrates sampled results of our approach.

Our approach is inspired by the classical divide-and-conquer algorithm [17], but relies on semantic and instance segmentation to divide the problem. Luckily, diverse panoptic segmentation annotations are relatively easy to collect compared to depth supervision, and we can leverage existing large-scale panoptic segmentation datasets such as COCO Panoptic Segmentation dataset [23] to complement the limited depth supervision. As an auxiliary task, semantic and instance segmentation not only helps split objects and categories for local depth prediction, but also necessitates the model's understanding of shape and contour regions. Thus, it can improve the model's generalization ability, and is also useful in cases where only sparse or low res depth annotations (*e.g.*, depth order of point pairs as in [2, 37], or Kinect data as in [32]) are available.

We show experiments on three benchmark datasets, demonstrating that our method can significantly improve the performance of depth estimation. Particularly, on the challenging "Depth in the Wild" (DIW) dataset [2], we achieve a new state-of-the-art error rate of 11.21% improving upon the previous best result of 13.02% [39].

In summary, we present a novel framework for monocular depth estimation based on a semantic divide-and-conquer strategy. We present an implementation of this high-level framework through SDC-Depth Net, a carefully designed end-to-end trainable architecture. Experimental validation of our approach shows consistent improvements over the state-of-the-art on three benchmark datasets.

2. Related Work

Single Image Depth Prediction There has been a long history of methods that have attempted to predict depth from a single image [11, 31, 24, 26]. Recently monocular depth estimation has gained popularity due to the ability of CNNs to learn strong priors from images corresponding to geometric layout. Among others, Laina *et al*. [14] propose a fully convolutional architecture with up-projection blocks to handle high-dimensional depth regression. In [19], a two-stream convolutional network is proposed, which simultaneously predicts depth and depth gradients to preserve more depth details. Besides using deep networks alone, recent works have shown that the combination of deep networks and shallow models [18, 25, 36, 40, 30] can also deliver superior depth estimation performance. Meanwhile, different forms of supervision and learning techniques have also been explored in recent works to improve the generalization ability of depth estimation models, including self-supervised learning with photometric losses from stereo images [6, 8] or multiple views [43, 34, 7], transfer learning using synthetic images [42, 42, 1], and those using sparse [2, 37] or dense [21, 35, 33, 20] relative depth as supervisions.

Augmenting Depth with Semantic Segmentation Some recent works [41, 39, 28] propose to improve monocular depth estimation with semantic segmentation annotations. For instance, Liu *et al*. [24] propose to guide single image depth estimation with semantic labels using Markov random fields. Xu *et al*. [39] develop a multi-modal distillation module, which can leverage intermediate depth and segmentation predictions to refine the final output. In [12] a synergy network together with an attention-driven loss is proposed to better propagate semantic information to depth prediction. In comparison, [41] presents a task-recursive learning strategy, which can refine both depth and segmentation predictions through task-level interactions. Another related work to ours is [28], where depth estimation is learned in a unsupervised manner by integrating both semantic segmentation and instance edges as input.

Although improvement has been achieved, these approaches have their own drawbacks. For one, existing works estimate depth for different categories with a single model. We argue that the depth values of different categories may exhibit different properties and subject to a variety of data distributions. Covering all these variations with one model may be sub-optimal. In addition, besides semantic categories, object instance information may also play a crucial role in depth estimation. However, compared to semantic segmentation, instance detection/segmentation is less explored in monocular depth estimation. Compared to these existing works, our proposed method performs depth estimation for each segment independently by investigating their semantic and instance information. We disentangle canonical depth estimation and depth scale inference leading to more accurate depth prediction results.

3. Sematic Divide-and-Conquer Network for Monocular Depth Estimation

We present SDC-Depth Net, an end-to-end trainable depth prediction network based on the aforementioned Semantic Divide-and-Conquer strategy. Our SDC-Depth Net consists of four parts: a backbone network, a segmentation module, a depth prediction module, and a depth aggregation module. Figure 2 overviews the proposed method.

The backbone network, shared by the segmentation and the depth prediction modules, extracts features of an input image. The segmentation module performs semantic and instance segmentation to divide the image into semantic

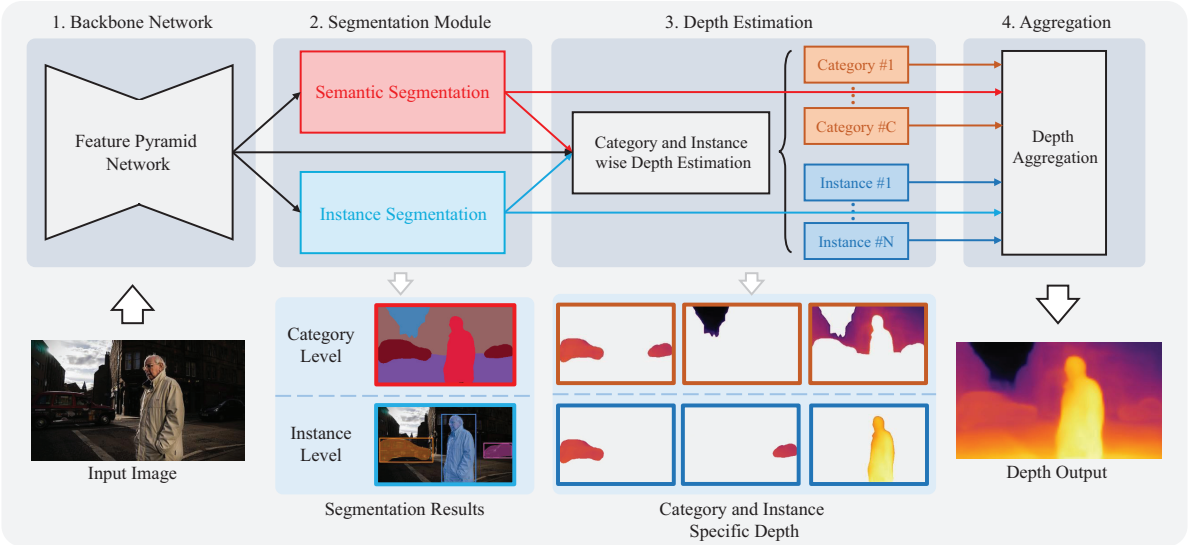


Figure 2: Overview of the proposed SDC-Depth Net for depth prediction. Our method decomposes the input images into category and instance segments, predicts depth maps for each individual segments specifically, and stitches the segment-level depth into the final output.

segments. For each semantic segment, the depth estimation module infers a category-specific depth map in a canonical space, as well as scale and shift parameters based on the global context. The aggregation module then stitches and aggregates the per-segment depth maps to generate a globally consistent depth map.

In our experiments, we adopt a feature pyramid network (FPN) [22] with ResNet-50 [10] as the backbone network. We use a fully convolutional network (FCN) [22, 38] and a Mask R-CNN model [9] for semantic and instance segmentation respectively. The FCN network performs semantic segmentation for C categories, where the first K categories are object classes (*e.g.* person, car) and the rest belong to stuff classes (*e.g.* road, grass). The Mask R-CNN network detects object instance masks for the K object classes. We now discuss each part in detail.

3.1. Per-Segment Depth Estimation

Given a semantic segment, such as a category mask or an instance mask, the depth prediction module predicts a segment-centered canonical depth map, as well as a transformation to convert the canonical depth to the global depth space. In this way, we decompose depth prediction into local segment depth prediction and global transformation estimation, which we will show to be beneficial compared to the direct prediction baseline.

We use two depth prediction streams to handle semantic category segments and instance segments respectively. The category segment stream operates in the category-level by predicting depth for each entire category jointly, whereas for countable object classes, the absolute depth of an instance can vary a lot depending on its position in the scene. Therefore, the instance-wise depth stream is further designed to improve the depth map on a *per instance* basis.

Category-wise Depth Estimation. Given a semantic category, we use a two-branch architecture to predict its canonical depth and global transformation. As shown in Figure 3, the local branch consists of a stack of convolutional layers, which takes as input the backbone image feature pyramid and predicts the canonical depth for each semantic category. We use the sigmoid function to normalize the output depth into the canonical space. The global depth decoding branch contains a Global Average Pooling (GAP) layer and a stack of fully connected layers. It maps the input feature pyramid to a vector characterizing the global context of the input image, which is used to infer the global transformation $\mathcal{T}_c(\cdot)$ for the c-th semantic category. Then, the global depth for the c-th category is computed as $\boldsymbol{D}_c = \mathcal{T}_c(\tilde{\boldsymbol{D}}_c)$. In our experiments, we adopt an affine transformation $\mathcal{T}_c(\tilde{\boldsymbol{D}}_c) = w_c \cdot \tilde{\boldsymbol{D}}_c + b_c$ for simplicity.

Instance-wise Depth Estimation. For object classes, such as human, car, *etc*., we can borrow the ROIAlign technique from Mask R-CNN [9] to extract features per object instance, and map these to a depth map. However, the resolution of the default ROIAlign features is too low (28×28) for accurate depth prediction, especially for larger objects.

To address this issue, we propose a new network architecture for high-resolution instance depth estimation (*c.f.* Figure 3). The instance stream consists of two branches, a

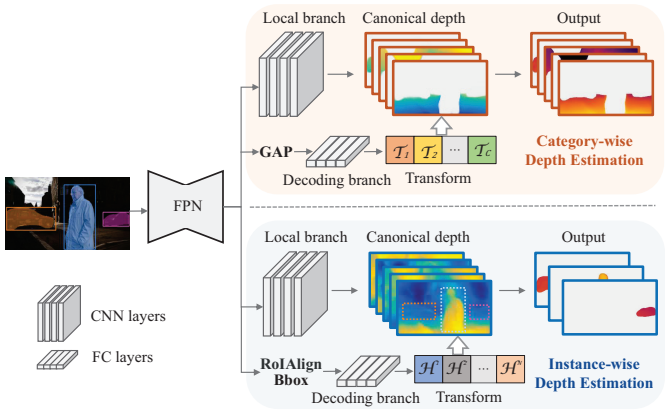


Figure 3: Our two-stream depth prediction module for category and instance-wise depth estimation. Each stream contains a local depth representation branch to infer canonical depth (normalized to $[0, 1]$) and a global decoding branch to estimate a scale-and-shift transformation based on global context (either GAP output or the combination of RoIAligned instance features and box positions).

fully convolutional local branch and an instance depth decoding branch. The local branch operates in a fully convolutional manner, which takes the backbone image feature pyramid as input and predicts a category-agnostic depth representation map $\tilde{\boldsymbol{F}}$ of size $H \times W \times Z$ for the entire input image (Z is set to 32 in our experiments). Given the bounding box location of the i-th object instance, its instance level depth map representation $\tilde{\boldsymbol{F}}^i \in \mathbb{R}^{H_i \times W_i \times Z}$ can be computed by cropping from $\tilde{\boldsymbol{F}}$, where $H_i \times W_i$ is the spatial size of its bounding box.

To predict the depth for the i-th instance, the depth decoding branch extracts a fixed-length feature vector from the instance region using ROIAlign[1] on the backbone feature. Given the category $c_i \in \{1, 2, \ldots, K\}$ of the instance inferred by the segmentation module, the depth decoding branch then takes as input the ROIAlign feature vector as well as the normalized bounding box coordinates of the instance, and predicts a linear depth decoding function $\mathcal{H}^i(\cdot) = \mathcal{G}^i \circ \mathcal{C}^i(\cdot)$ corresponding to the c_i-th object category. The function $\mathcal{C}^i$ is a 1×1 convolutional layer, which linearly combines the Z channels of the instance depth representation map $\tilde{\boldsymbol{F}}^i$ into a instance-centric canonical depth map. The function $\mathcal{G}^i$ is an affine transformation, which further transforms the canonical depth into the global depth $\boldsymbol{F}_i \in \mathbb{R}^{H_i \times W_i}$ by adjusting its scale and shift. The parameters of the two functions for different object categories are produced by the depth decoding branch in a category-specific manner. Assume the total number of transformation parameters for each category is n. The depth decoding branch predicts K sets of parameters through an output vector of length $n \times K$. We select the c_i-th set of parameters for the i-th instance.

[1]We use the same implementation of ROIAlign in [9].

3.2. Segmentation Guided Depth Aggregation

Now we have produced the category depth maps $\{\boldsymbol{D}_c \in \mathbb{R}^{H \times W} | c = 1, \ldots, C\}$ and a set of object instance depth maps $\{\boldsymbol{F}_i \in \mathbb{R}^{H_i \times W_i} | i = 1, 2, \ldots, N\}$ for a total number of N object instances within the input image. To make the final depth prediction, a depth aggregation module combines per-segment depth maps based on the semantic segmentation and the instance segmentation results.

Our depth aggregation module proceeds in two steps. Given instance depth maps $\boldsymbol{F}_i$ and their category labels, the first step performs local updates to the region of each instance in its corresponding category depth map $\{\boldsymbol{D}_c | c = 1, 2, \ldots, K\}$. To this end, we associate each object category depth map $\boldsymbol{D}_c$ with an normalization mask $\boldsymbol{M}_c$ with the same spatial size $H \times W$, whose elements are all initialized to constant value 1. The normalization masks are used to record the update from each instance depth map, and normalize the final depth map accordingly. Given the category c_i and bounding box location of the i-th instance, we denote the instance region on the corresponding depth map $\boldsymbol{D}_{c_i}$ and the normalization mask $\boldsymbol{M}_{c_i}$, as $\boldsymbol{D}^i_{c_i} \in \mathbb{R}^{H_i \times W_i}$ and $\boldsymbol{M}^i_{c_i} \in \mathbb{R}^{H_i \times W_i}$, respectively. The depth map and the normalization mask can then be locally updated as follows:

$$\begin{aligned} \boldsymbol{D}^i_{c_i} &\leftarrow \boldsymbol{D}^i_{c_i} + v \times p^i \odot \boldsymbol{S}^i \odot \boldsymbol{F}^i, \\ \boldsymbol{M}^i_{c_i} &\leftarrow \boldsymbol{M}^i_{c_i} + v \times p^i \odot \boldsymbol{S}^i, \end{aligned} \tag{1}$$

where v is a hyper-parameter to balance the weight of instance depth maps (v is set to 10 in our experiments); $\odot$ indicates the element-wise multiplication; p^i denotes the probability of the i-th instance belonging to category c_i, and $\boldsymbol{S}^i$ of spatial size $H_i \times W_i$ represents the upsampled segmentation mask of the i-th instance. Both p^i and $\boldsymbol{S}^i$ are generated by the instance segmentation model of our segmentation module, and are used to measure the reliability of the i-th instance prediction.

After all instance regions have been updated, each category depth map $\boldsymbol{D}_c$ is computed:

$$\boldsymbol{D}_c \leftarrow \boldsymbol{D}_c / \boldsymbol{M}_c \tag{2}$$

where the division is performed element-wisely. More details of the first step is summarized in Algorithm 1.

The second step aggregates all of the updated category depth maps $\boldsymbol{D}_c$ according to the semantic segmentation results. This can be performed through the following weighted combination:

$$\boldsymbol{D} = \sum_{c=1}^{C} \boldsymbol{P}_c \odot \boldsymbol{D}_c, \tag{3}$$

Algorithm 1 Update category depth maps with instance depth maps.

Input: Category-specific depth maps $\{\boldsymbol{D}_c|c = 1, 2, \ldots, K\}$, instance-specific depth maps $\boldsymbol{F}^i$, instance segmentation masks $\boldsymbol{S}^i$, instance category c_i, instance classification probability $p_i, i = 1, 2, \ldots, N$.

Output: Updated category-specific depth maps $\{\boldsymbol{D}_c|c = 1, 2, \ldots, K\}$

1: Initialize normalization mask $\boldsymbol{M}_c$ for each category c
2: **for** $i = 1, 2, \ldots, N$ **do**
3: Locate instance regions $\boldsymbol{D}^i_{c_i}$ and $\boldsymbol{M}^i_{c_i}$ on depth map and normalization masks.
4: Locally update depth map by $\boldsymbol{D}^i_{c_i} \leftarrow \boldsymbol{D}^i_{c_i} + v \times p^i \odot \boldsymbol{S}^i \odot \boldsymbol{F}^i$
5: Locally update energy mask by $\boldsymbol{M}^i_{c_i} \leftarrow \boldsymbol{M}^i_{c_i} + v \times p^i \odot \boldsymbol{S}^i$
6: **end for**
7: Normalize each category depth map $\boldsymbol{D}_c \leftarrow \boldsymbol{D}_c / \boldsymbol{M}_c$

where $\boldsymbol{D}$ represents the final depth map. $\boldsymbol{P}_c$ is the per-class, per-pixel segmentation result predicted by the semantic segmentation module, where for a class c, an element located at (x, y) represents the probability of the corresponding pixel belonging to that class.

3.3. Network Training

The whole system can be trained in an end-to-end manner using the following loss function:

$$L = \alpha_I L_I + \alpha_S L_S + \alpha_D L_D, \tag{4}$$

where we use standard implementations of the instance segmentation loss L_I [9], and semantic segmentation losses L_S[27]. The depth prediction loss L_D varies depending on the depth supervision. On training datasets with dense depth annotations (*e.g.*, NYU-Depth V2 [32] and Cityscapes [3]) we use a standard L_1 loss, and on datasets with relative depth annotations between pairs of random points (*e.g.*, DIW dataset [2]), we use the ranking loss proposed in [2]. More details of the loss functions and their loss weights $\alpha_I, \alpha_S, \alpha_D$ are present in the supplementary materials. All four modules of our method can then be jointly trained by minimizing the overall loss function in (4).

4. Experiments

4.1. Implementation

We adopt ResNet50 pre-trained on the ImageNet classification task to initialize our backbone network. Detailed architecture design for our depth prediction module can be found in the supplementary materials. We resize each input image to have a minimum side of 256 pixels while maintaining its aspect ratio. Data augmentation techniques including random flipping, scaling and color jitter have also been employed to avoid over-fitting. Our network is trained using Adam optimizer [13] with a mini-batch of 4 input images. Our whole network has 50.4 M parameters and runs at 10.23 FPS for 19 semantic categories on one NVIDIA GTX 1080 TI GPU. Source code is available at https://bit.ly/39oty26.

We evaluate our method on three depth datasets, including Cityscapes [3], DIW [2] and NYU-Depth V2 [32], which involve either dense or sparse depth annotations, and contain diverse scenes. The performance of compared methods are measured by: RMSE in both linear and log space, absolute and squared relative error (Abs Rel and Sq Rel), depth accuracy (with thresholds 1.25, 1.25^2 and 1.25^3), and weighted human disagreement rate (WHDR) [4, 2]. We adopt the evaluation code of [5] to calculate the above metrics.

4.2. Cityscapes Results

Cityscapes [3] is a large dataset for urban scene understanding, containing both depth and panoptic segmentation annotations of 20 semantic categories. We train our model for 25 epochs on the training set of 2975 images with the initial learning rate of 5e-3. We evaluate the trained model on the validation (500 images) and test (1525 images) sets, and compare to 3 state-of-the-art methods including Laina *et al*. [14], Xu *et al*. [39], and Zhang *et al*. [41]. Among them, Xu *et al*. [39], and Zhang *et al*. [41] train their model on both depth estimation and semantic segmentation in a multi-task manner. Table 1 reports the results. Our method achieves higher performance than compared methods, particularly in terms of RMSE and depth accuracy. This should be mainly attributed to the fact that our method predicts each category and instance depth independently with specific depth decoders. Since Xu *et al*. and Zhang *et al*. also leverage semantic segmentation annotation, their performance is superior than Laina *et al*. Qualitative results are shown in Figure 4.

4.3. DIW Results

DIW [2] is a large-scale dataset containing images of diverse scenes in the wild, where each image is manually annotated with the relative depth order (either closer or further away from the camera) between one randomly sampled point pair. The whole dataset is split into 421K training images and 74K test images. Since DIW dataset does not contain segmentation annotations and the COCO panoptic segmentation dataset [23] also contains images of unconstrained scenes, we simultaneously train our model on DIW and COCO for relative depth estimation and segmentation, respectively. In order to reduce computational complexity, we adopt the super-class annotation of COCO dataset to train our segmentation module, containing 15 stuff and 12 object classes. During training, we sequentially feed training images from both datasets to the network in each itera-

Method	Error				Accuracy		
	RMSE	RMSE (log)	Abs Rel	Sq Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Laina *et al*. [14]	7.273	0.448	0.257	4.238	0.765	0.893	0.940
Xu *et al*. [39]	7.117	0.428	0.246	4.060	0.786	0.905	0.945
Zhang *et al*. [41]	7.104	0.416	0.234	**3.776**	0.776	0.903	0.949
Ours	**6.917**	**0.414**	**0.227**	3.800	**0.801**	**0.913**	**0.950**

Table 1: Comparison with state-of-the-art methods on Cityscapes test set [3]. Best results are in **bold** font, second best are underlined.

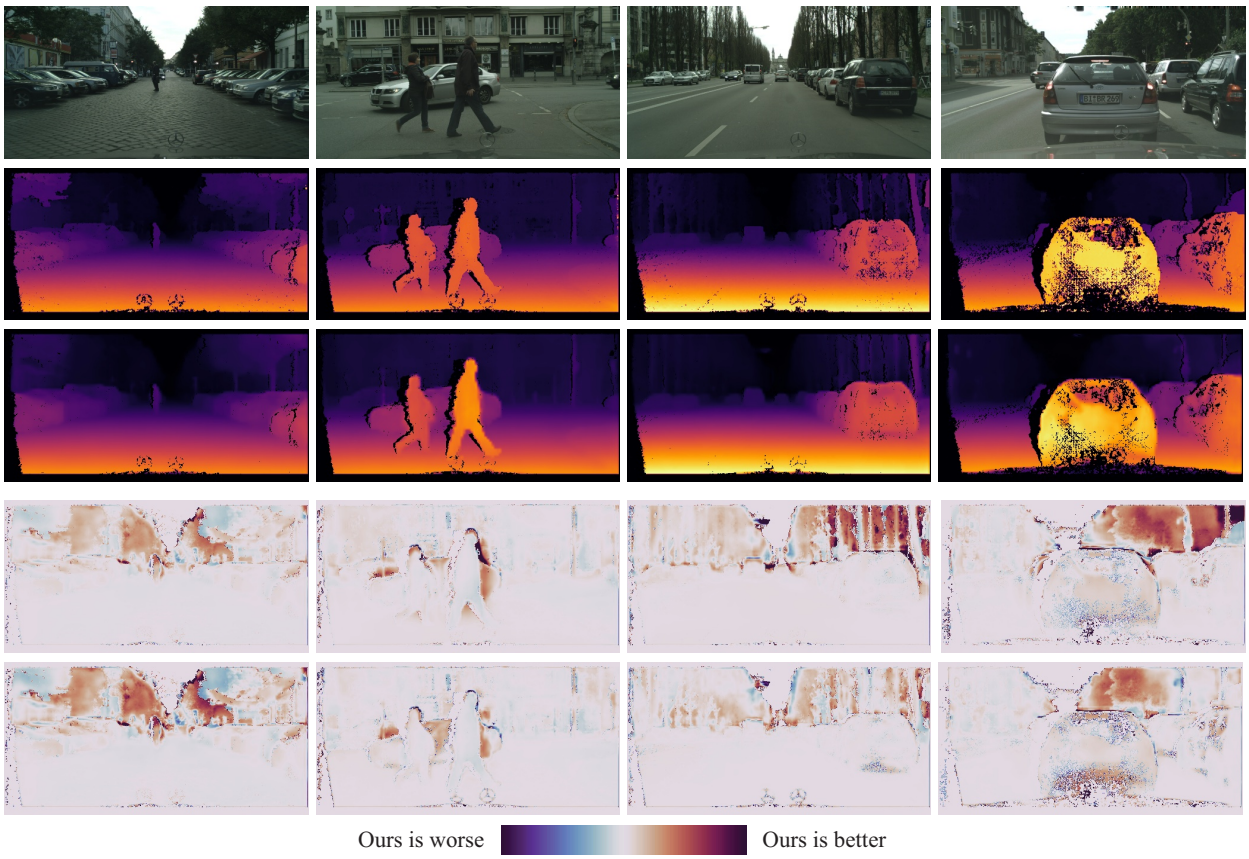


Figure 4: Qualitative results on Cityscapes dataset. The first three rows are input images, ground truth, and our predicted depth maps, respectively. The last two rows are error map comparisons of our method against Xu *et al*. [39] and Zhang *et al*. [41], respectively, where dark red indicates our method achieves lower error and dark blue is the opposite.

Method	Chen *et al*. [2]	Xian *et al*. [37]	Xu *et al*. [39]	Ours
WHDR	22.14%	14.98%	13.02%	**11.21%**

Table 2: Comparison with state-of-the-art methods on DIW dataset [2]. The best result is in **bold** font.

tion, and update network parameters using the accumulated gradients. Network training starts with an initial learning rate of 1e-3 and converges at around 45K iterations.

We compare our method against three state-of-the-art approaches, including Chen *et al*. [2], Xian *et al*. [37], and Xu *et al*. [39], where Xu *et al*. [39] is trained on both DIW and COCO dataset using the same training strategy as ours. Table 2 shows the comparison results in terms of WHDR. Xian *et al*. achieves lower WHDR than Chen *et al*. Meanwhile, Xu *et al*. outperforms Xian *et al*. by exploring additional segmentation data. In comparison, our proposed method adopt a divide-and-conquer strategy to estimate depth for each segments independently, thus achieves the best performance. Figure 5 compares the predicted depth maps of Xu *et al*. [39] and our proposed method.

Method	Error			Accuracy		
	RMSE	RMSE (log)	Abs Rel	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Laina *et al*. [14]	0.584	0.198	0.136	0.822	0.956	0.989
Xu *et al*. [40]	0.593	-	0.125	0.806	0.952	0.986
Qi *et al*. [29]	0.569	-	0.128	0.834	0.960	0.990
Lee *et al*. [15]	0.572	0.193	0.139	0.815	0.963	0.991
Fu *et al*. [5]	0.509	0.188	**0.116**	0.828	0.965	**0.992**
Zhang *et al*. [41]	0.501	0.181	0.144	0.815	0.962	**0.992**
Xu *et al*. [39]	0.582	-	0.120	0.817	0.954	0.987
Ours	**0.497**	**0.174**	0.128	**0.845**	**0.966**	0.990

Table 3: Comparison with state-of-the-art methods on NYU-Depth V2 dataset [32].

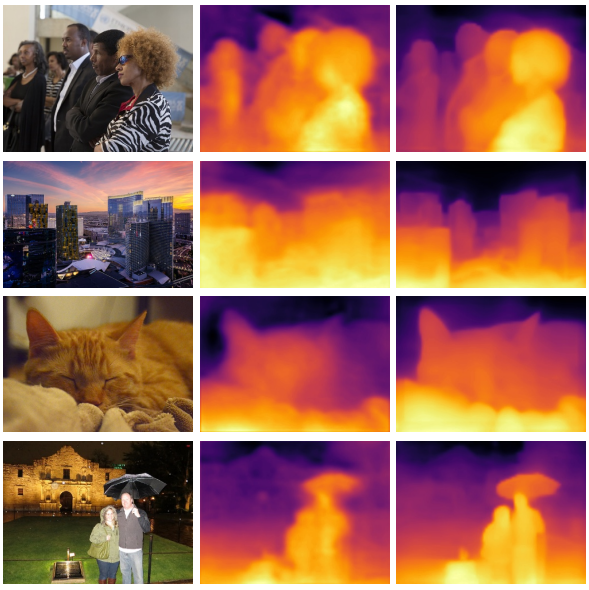


Image Xu *et al*. [39] Ours

Figure 5: Qualitative results on DIW test set [2]. All methods are trained on sparse point-level supervison.

4.4. NYU-Depth V2 Results

The NYU-Depth V2 dataset contains 464 indoor scenes, where 249 of them are for training and the rest for testing. We train our networks by randomly sampling 40K images from the 249 training scenes and following the multi-task training strategy introduced in Section 4.3. We adopt an initial learning rate of 1e-3 and train the network for 15 epochs.

We compare our method to seven state-of-the-art methods. Among them, Lee *et al*. [16] and Fu *et al*. [5] use all the 120K training images, and Xu *et al*. [39] and Zhang *et al*. [39] also use the available segmentation supervision. As shown in Table 3, the proposed method performs favorably against state-of-the-art approaches, particularly in terms of depth accuracy by using a limited amount of segmentation annotations. We believe our performance can be further improved by using more segmentation data from indoor scenes.

		SDC-A	SDC-B	SDC-C	SDC-D
Design Choice	Cat.	✗	✓	✓	✓
	Ins.	✗	✗	✓	✓
	DEnt.	✗	✗	✗	✓
Err.	RMSE	7.203	6.962	6.958	**6.917**
	Abs Rel	0.276	0.236	0.234	**0.227**
Acc.	$\delta < 1.25$	0.767	0.794	0.797	**0.801**
	$\delta < 1.25^2$	0.895	0.911	0.911	**0.913**
	$\delta < 1.25^3$	0.941	0.949	0.951	**0.950**

Table 4: Ablation study on Cityscapes dataset [3]. Components tested are category (Cat.) and instance (Ins.) depth estimation, and disentangling canonical depth and scale inference (DEnt). The best results are in **bold** font.

4.5. Ablation Study

To achieve more comprehensive understanding of our method, we perform ablative study on Cityscapes [3] and DIW [2] datasets by adjusting different modules of our method. Unless otherwise stated, we follow the same experimental setup as described in Section 4.1.

Effects of semantic divide-and-conquer. The proposed SDC-Depth Net learns *category* and *instance* aware depth estimation with *disentangled* canonical depth and scale inference mechanism. To investigate the impact of the above design choices, we compare the performance of baselines (SDC-A to SDC-D) containing different subsets of these choices on the Cityscapes dataset as shown in Table 4. It can be observed that category aware depth estimation plays a very important role in improving the depth accuracy. Instance aware depth estimation and disentangled depth prediction also yield a considerable performance gain. To further verify their effectiveness, Figure 6 compares the depth accuracy of baseline methods with respect to different categories on Cityscapes validation set. The performance gain

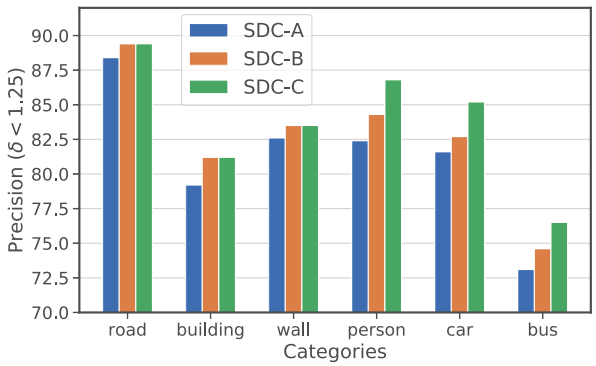


Figure 6: Depth accuracy of our variants across semantic categories on Cityscapes validation set [3].

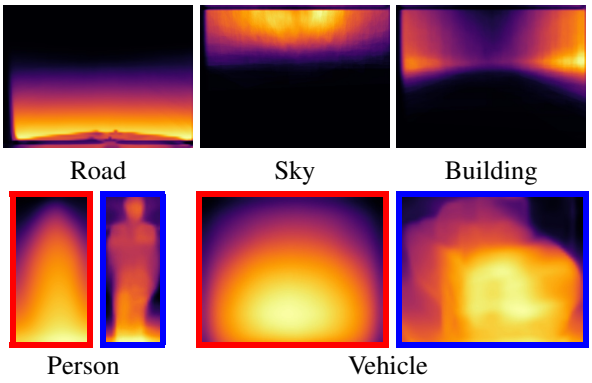


Figure 7: Average canonical depth maps of stuff (top) and object (bottom) categories predicted by our SDC-Depth Net. For each object category (bottom), we present both its averaged depth map (left, red border) as well as one random instance depth map of that category (right, blue border).

of the category-aware depth estimation is consistent cross all categories while instance-aware depth estimation is more effective for object categories.

We also visualize the average canonical depth maps, for a number of different segments (Figure 7). We can see how by splitting the depth prediction at a segment level, the network can learn simpler *category-specific* depth priors.

Benefits of segmentation annotation. To evaluate how much our method benefits from additional segmentation annotations, we train 4 instances of our SDC-Depth Net on DIW and COCO datasets using different portions (25%, 50%, 75%, and 100%, respectively) of the COCO dataset for training. In addition to this, we also train two baseline networks that are encoder-decoder architectures that directly predict depth maps with a similar parameter count to SDC-Depth Net. One of the baseline is trained only on DIW, named as BNet-DIW. The other one is trained on both

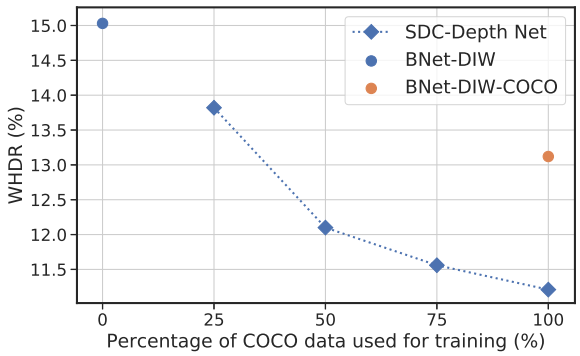


Figure 8: WHDR on DIW test set [2] of SDC-Depth Net and baseline methods trained on different portion of COCO training data. As our method has access to more and more segmentation labels, we see that quality improves beyond that of baseline approaches (BNet is a standard U-Net [2] with similar parameter count).

DIW and COCO datasets in a multi-task learning manner, named as BNet-DIW-COCO, as in [35]. The comparison results in terms of depth accuracy on DIW test set are shown in Figure 8. It can be observed that the performance can be consistently improved by using more segmentation training data, and when using all the COCO training data, the proposed method outperforms BNet-DIW-COCO with a significant margin.

5. Conclusion

We present a semantic divide-and-conquer strategy to reduce monocular depth estimation into that of individual semantic segments. Based on this idea, the SDC-Depth Net is designed, which decomposes an input images into segments of different categories and instances, and infers the canonical depth as well as the scale-and-shift transformation for each segment using specifically trained parameters. An aggregation method is also developed to stitch the per-segment depth into the final depth map. The whole network can be trained fully end-to-end by leveraging additional segmentation annotations. Experiments on three popular benchmarks demonstrates the effectiveness of our method.

Acknowledgments. This work is supported by National Key R&D Program of China (2018AAA0102001), National Natural Science Foundation of China (61725202, U1903215, 61829102, 91538201, 61751212, 61906031), Fundamental Research Funds for the Central Universities (DUT19GJ201), China Postdoctoral Science Foundation (2019M661095), National Postdoctoral Program for Innovative Talent (BX20190055), and Adobe Research.

References

[1] Amir Atapour-Abarghouei and Toby P Breckon. Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2800–2810, 2018. 2

[2] Weifeng Chen, Zhao Fu, Dawei Yang, and Jia Deng. Single-image depth perception in the wild. In *Advances in neural information processing systems*, pages 730–738, 2016. 1, 2, 5, 6, 7, 8

[3] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 5, 6, 7, 8

[4] 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. 1, 5

[5] 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. 5, 7

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

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

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

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 3, 4, 5

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[11] Derek Hoiem, Alexei A Efros, and Martial Hebert. Geometric context from a single image. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 654–661. IEEE, 2005. 2

[12] Jianbo Jiao, Ying Cao, Yibing Song, and Rynson Lau. Look deeper into depth: Monocular depth estimation with semantic booster and attention-driven loss. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 53–69, 2018. 2

[13] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[14] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth international conference on 3D vision (3DV)*, pages 239–248. IEEE, 2016. 1, 2, 5, 6, 7

[15] Jae-Han Lee, Minhyeok Heo, Kyung-Rae Kim, and Chang-Su 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. 7

[16] Jae-Han Lee and Chang-Su Kim. Monocular depth estimation using relative depth maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2019. 7

[17] Charles Eric Leiserson, Ronald L Rivest, Thomas H Cormen, and Clifford Stein. *Introduction to algorithms*, volume 6. MIT press Cambridge, MA, 2001. 2

[18] Bo Li, Chunhua Shen, Yuchao Dai, Anton Van Den Hengel, and Mingyi 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

[19] Jun Li, Reinhard Klein, and Angela Yao. A two-streamed network for estimating fine-scaled depth maps from single rgb images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3372–3380, 2017. 2

[20] Zhengqi Li, Tali Dekel, Forrester Cole, Richard Tucker, Noah Snavely, Ce Liu, and William T Freeman. Learning the depths of moving people by watching frozen people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4521–4530, 2019. 2

[21] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018. 2

[22] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 3

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2, 5

[24] Beyang Liu, Stephen Gould, and Daphne Koller. Single image depth estimation from predicted semantic labels. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1253–1260. IEEE, 2010. 2

[25] Fayao Liu, Chunhua Shen, and Guosheng Lin. Deep convolutional neural fields for depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5162–5170, 2015. 2

[26] Miaomiao Liu, Mathieu Salzmann, and Xuming He. Discrete-continuous depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–723, 2014. 2

[27] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 5

[28] Yue Meng, Yongxi Lu, Aman Raj, Samuel Sunarjo, Rui Guo, Tara Javidi, Gaurav Bansal, and Dinesh Bharadia. Signet: Semantic instance aided unsupervised 3d geometry perception. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9810–9820, 2019. 1, 2

[29] 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. 7

[30] Anirban Roy and Sinisa Todorovic. Monocular depth estimation using neural regression forest. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5506–5514, 2016. 2

[31] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3d: Learning 3d scene structure from a single still image. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):824–840, 2008. 2

[32] 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. 2, 5, 7

[33] Chaoyang Wang, Simon Lucey, Federico Perazzi, and Oliver Wang. Web stereo video supervision for depth prediction from dynamic scenes, 2019. 2

[34] Chaoyang Wang, José Miguel Buenaposada, Rui Zhu, and Simon Lucey. Learning depth from monocular videos using direct methods. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2022–2030, 2018. 2

[35] Lijun Wang, Xiaohui Shen, Jianming Zhang, Oliver Wang, Zhe L. Lin, Chih-Yao Hsieh, Sarah Kong, and Huchuan Lu. Deeplens: shallow depth of field from a single image. *ACM Trans. Graph.*, 37(6):245:1–245:11, 2018. 2, 8

[36] Peng Wang, Xiaohui Shen, Zhe Lin, Scott Cohen, Brian Price, and Alan 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. 2

[37] Ke Xian, Chunhua Shen, Zhiguo Cao, Hao Lu, Yang Xiao, Ruibo Li, and Zhenbo Luo. Monocular relative depth perception with web stereo data supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 311–320, 2018. 2, 6

[38] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8818–8826, 2019. 3

[39] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 675–684, 2018. 1, 2, 5, 6, 7

[40] Dan Xu, Wei Wang, Hao Tang, Hong Liu, Nicu Sebe, and Elisa Ricci. Structured attention guided convolutional neural fields for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3917–3925, 2018. 2, 7

[41] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Zequn Jie, Xiang Li, and Jian Yang. Joint task-recursive learning for semantic segmentation and depth estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 235–251, 2018. 2, 5, 6, 7

[42] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. T2net: Synthetic-to-realistic translation for solving single-image depth estimation tasks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 767–783, 2018. 2

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