Monocular Relative Depth Perception with Web Stereo Data Supervision

Ke Xian[†], Chunhua Shen[‡], Zhiguo Cao[†*], Hao Lu[†], Yang Xiao[†], Ruibo Li[†], Zhenbo Luo[◇]
[†]School of Automation, Huazhong University of Science and Technology, China
[‡]The University of Adelaide, Australia [◇]Samsung Research Beijing, China
e-mail: kexian@hust.edu.cn

Abstract

In this paper we study the problem of monocular relative depth perception in the wild. We introduce a simple yet effective method to automatically generate dense relative depth annotations from web stereo images, and propose a new dataset that consists of diverse images as well as corresponding dense relative depth maps. Further, an improved ranking loss is introduced to deal with imbalanced ordinal relations, enforcing the network to focus on a set of hard pairs. Experimental results demonstrate that our proposed approach not only achieves state-of-the-art accuracy of relative depth perception in the wild, but also benefits other dense per-pixel prediction tasks, e.g.*, metric depth estimation and semantic segmentation.*

1. Introduction

Monocular depth estimation is a long-standing task in Computer Vision, which benefits many applications, such as 2D-to-3D conversion, 3D modeling, and robotics. Although significant progress [1, 2, 3, 4, 5] has been witnessed in recent years due to the success of deep convolutional networks (ConvNets), depth estimation from monocular images still remains challenging, especially for images *in the wild*. Most state-of-the-art methods trained on one dataset often perform worse on a different one. For example, models trained on an indoor dataset (*e.g.*, NYUDv2) fail to predict satisfactory depth in outdoor scenes. Our goal is thus to use one single model to predict relative depth in general scenes, which happens to agree with the spirit of Robust Vision Challenge 2018[1].

Actually, many applications only need relative depth, *e.g.*, 2D-to-3D conversion [7] and depth-of-field [8]. To recover relative depth for monocular images in the wild, Chen *et al.* [6] proposed a "Depth in the Wild" (DIW) dataset consisting of 495k web images, where each image was *manually* annotated with two points of ordinal relation (closer '<' and further '>'). However, training with only one pair of ordinal relation is not sufficient to get satisfactory predictions (see Figure 1). Based on above observations, a question arises: *how to cheaply get diverse images as well as corresponding dense relative depth maps?*

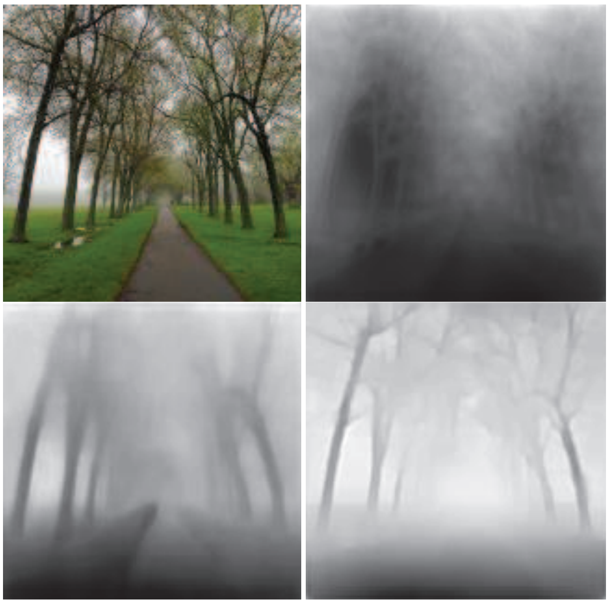


Figure 1. Learning relative depth with one pair of ordinal relation is prone to yield confused predictions [6] (top right). Pre-training a ConvNet on the NYUDv2 dataset with multiple pairs of supervision helps make better predictions [6] (bottom left). We train a ConvNet on our proposed ReDWeb dataset with multiple pairs of ordinal relations to perceive relative depth *in the wild* and achieve state-of-the-art performance (bottom right). For a relative depth map, the darker the pixel is, the closer it should be, and vice versa.

Since a disparity map represents the relative depth of a scene, in this paper, we introduce an effective method to *automatically* produce disparity maps from web stereo images. Considering that web stereo image pairs are not always well-calibrated and the horizontal component of a correspondence map can be seen as a disparity map, we opt to compute correspondence maps by a state-of-the-art optical

*Part of the work was carried out when K. Xian was visiting The University of Adelaide. Z. Cao is the corresponding author.

[1] http://www.robustvision.net/index.php

flow method [9] instead of stereo matching. Therefore, we propose a new dataset termed "Relative Depth from Web" (ReDWeb)[2] that consists of 3600 scene-diverse images as well as corresponding relative depth maps.

Inspired by Chen *et al*. [6], training with multiple pairs of supervision using a ranking loss can achieve promising results. We train a ConvNet to predict relative depth in a similar way. Instead of training with fixed point pairs [6], we resort to explore the diversity of sampled point pairs by online sampling. However, randomly sampling leads to the problem of imbalanced ordinal relations, *i.e*., the number of equal relation is far less than other two relations (closer and further). To improve model capability, we design an improved ranking loss to ease the problem caused by imbalanced ordinal relations. In particular, to avoid the difference of two unequal depth values being too large, we sort the loss of each unequal pair at each iteration, and only sum the loss of hard pairs. Extensive experimental results demonstrate the effectiveness of our approach, and our model pre-trained on the ReDWeb dataset can benefit other dense per-pixel prediction tasks, *e.g*., metric depth estimation and semantic segmentation.

The contributions of this work are as follows:

- We introduce a simple yet effective way to automatically produce dense relative depth annotations from web stereo images, and propose a new dataset "Relative Depth from Web" (ReDWeb) that contains diverse images annotated with dense relative depth maps.
- We deal with imbalanced ordinal relations by introducing an improved ranking loss that enforces our proposed ConvNet to focus on a set of hard pairs.
- We evaluate our approach on the DIW and NYUDv2 datasets, and achieve state-of-the-art performance. Furthermore, our ConvNet that is pre-trained using ordinal relations can benefit other dense per-pixel prediction tasks, *e.g*., metric depth estimation and semantic segmentation.

2. Related Work

RGBD datasets Most existing RGBD datasets are collected by depth sensors, either Kinect [10] or LiDAR [11]. However, Kinect can only be used in indoor scenes, while LiDAR is often used in outdoor scenes. It is difficult to get good results in the wild when training on these datasets due to the diversity of scenes. To address the problem of monocular relative depth perception in the wild, Chen *et al*. [6] propose a DIW dataset which covers a wide range of general scenes. But for each image, only a single pair of ordinal relationship is manually annotated. By contrast, our ReDWeb dataset is cheaply constructed by automatically computing disparity maps from web stereo images. Moreover, our dataset covers a wide range of scenes, at the same time provides a dense relative depth map for each image.

Metric depth estimation Early works on depth estimation from monocular images mainly depend on Markov Random Fields [11, 12, 13] and non-parametric learning methods [7, 14, 15, 16, 17]. Recent works achieve better prediction results by leveraging deep ConvNets [18, 19] and large RGBD datasets. Different network architectures have been tailored to directly regress [1, 2, 3] or classify [20] pixel-wise depth values. To enforce local consistency in the output depth map, Conditional Random Fields (CRFs) are integrated into a layer of ConvNets [4, 21, 22, 23] or used as a post-processing [20, 24]. Inspired by traditional methods that benefited from other vision tasks, such as semantic segmentation [12, 25], surface normal estimation [26] and intrinsic images estimation [15], researchers show greatly improved results using deep learning [21, 22, 24, 27, 28]. Chakrabarti *et al*. [5] predict probability distributions over coefficients using an overcomplete representation that characterizes local geometric structure.

Unlike supervised learning methods trained with a large number of RGBD images, some researches recover depth in an unsupervised learning fashion [29, 30, 31]. They take the idea of image reconstruction [32] to generate depth map based on the fact that stereo image pairs are easily accessible. To construct an end-to-end differentiable system, Taylor approximation [29] and bilinear interpolation [30] are chosen to derive a fully differentiable training loss.

Relative depth perception Since many applications do not need to know exact metric depth, such as 2D-to-3D conversion [7] and depth-of-field [8]. Some recent works [33, 6] focus on perceiving relative depth from single images. Zoran *et al*. [33] first learn a ConvNet to repeatedly classify pairs of points sampled based on superpixel segmentation, and then solve an energy optimization problem to recover global consistent metric depth. Chen *et al*. [6] directly map an input image to metric depth by training a multi-scale network with a ranking loss [34]. Further, the authors of [6] propose a DIW dataset in which each image is manually annotated with two points of ordinal relation. However, a major limitation of [6] is that the DIW dataset only provides one single pair of ordinal relation for each image, which discards important perceptual properties such as continuity, surface orientation [35], *etc*. Moreover, the definition of ranking loss [6] would encourage the difference of depth values to be infinitely large if the sampled two points have different depths. As a result, they fail to obtain satisfactory predictions when trained only with the DIW dataset.

3. Proposed method

Notice that training with more pairs of ordinal relations can boost the performance [6], and a ConvNet can learn effective representations from noisy data [36]. Based on

[2] https://sites.google.com/site/redwebcvpr18.

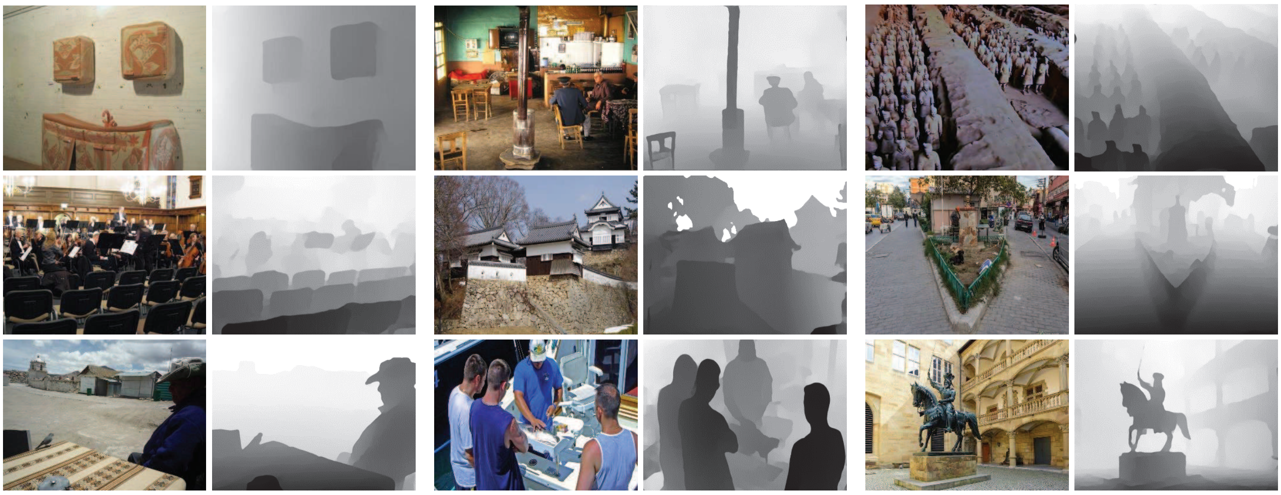
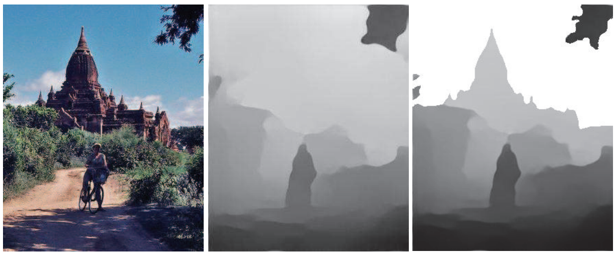


Figure 2. Examples of our ReDWeb dataset which covers a wide range of scenes, including both indoor and outdoor scenes. All relative depth maps are annotated automatically.



(a) Left image (b) Corr w/o seg (c) Corr w/ seg

Figure 3. An example of using semantic segmentation to correct textureless regions (sky). (a) is the left image from a web stereo image pair, (b) is the horizontal component of the correspondence map produced by [9], and (c) is the refined result after using semantic segmentation to mask sky areas as infinity.

above observations, we propose a ReDWeb dataset, and then train a ConvNet with multiple pairs of supervision signals to achieve state-of-the-art performance.

3.1. Our proposed ReDWeb dataset

In this section, we detail our proposed ReDWeb dataset. We first describe how to produce correspondence maps from web stereo data, and then discuss how to postprocess these coarse correspondence maps. Finally, we provide summary statistics about our ReDWeb dataset. Figure 2 demonstrates some examples of the ReDWeb dataset.

Data preprocessing The key idea behind this paper is that training a ConvNet with more pairs of ordinal relations would generate better results than only a single pair as in [6]. Instead of manually labeling ordinal relations, we use web stereo images to generate dense correspondence maps automatically, which can provide more pairs of ordinal relations. To this end, we use some keywords (*e.g.*, stereoscopic) to crawl about 40k stereo images from Flickr. Since web stereo images are not always rectified, directly using stereo matching methods, such as SGM [37] and MC-CNN [38], would produce massive noisy correspondence maps. Although uncalibrated epipolar rectification [39] can be used to rectify raw web stereo images before using stereo matching, the produced correspondence maps are usually still in poor quality. Therefore, we alternatively utilize the current state-of-the-art optical flow algorithm [9] to generate correspondence maps. We regard horizontal component of correspondence map as disparity d. For each pixel p in a left image I_1, we can find its correspondence $p + d_p$ in the corresponding right image I_2. However, web stereo images are not always in a side-by-side (left-right) format, optical flow methods sometimes fail to generate reasonable correspondence maps. Hence, postprocess is essential.

Data postprocessing Since there exists some stereo images in other formats, *e.g.*, anaglyph and left-center-right, flownet2.0 [9] would produce cluttered correspondences on these images. As a result, training on these images would confuse the ConvNet and leads to poor performance. Therefore, We manually exclude some poor coarse correspondence maps with user interactions, and unify the disparity to the same criterion that the darker the pixel is, the closer it should be. However, we find that the remaining correspondence maps are still noisy in textureless regions, especially in sky areas. As shown in Figure 3, textureless regions are difficult to estimate well, *e.g.*, sky. We propose to use semantic segmentation, which is effective to deal with textureless regions, to correct coarse correspondence maps. More specifically, we use RefineNet [40] trained on ADE20K [41] to segment sky areas, and then further optimize boundaries by using a fully connected CRF [42]. To generate final relative depth maps, we identify sky areas in the refined semantic segmentation results, and mask these areas in the

Dataset	Indoor	Outdoor	Annotation	# Images
NYUDv2 [10]	√		dense	1449
SUN3D [43]	√		dense	2.5M
Make3D [11]		√	sparse	534
KITTI [44]		√	sparse	93K
DIW [6]	√	√	single pair	495K
Ours	√	√	dense	3600

Table 1. Comparison of different RGBD datasets.

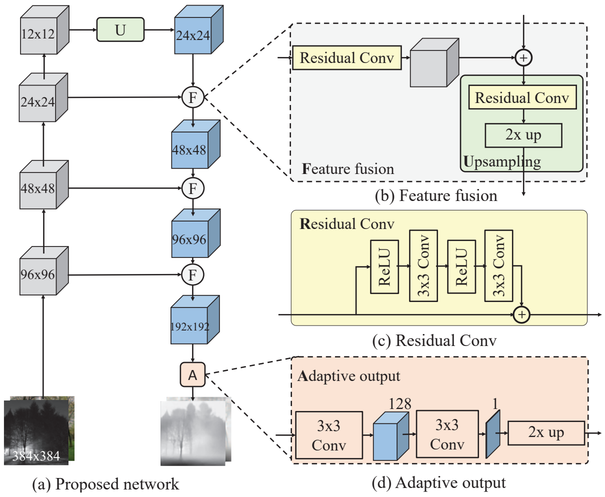
correspondence maps to be at infinity. Note that we also crop borders of left images and their correspondence maps, and keep them well aligned.

Dataset statistics Our ReDWeb dataset consists of 3600 images, which covers a wide range of scenes, such as street, office, hills, park, farm, night scenes, *etc*. To analyze the differences among existing depth datasets, we report some properties in Table 1. Different from other metric depth datasets, *e.g*., NYUDv2 [10] and SUN3D [43] for indoor scenes, Make3D [11] and KITTI [44] for outdoor scenes, our proposed dataset covers both indoor and outdoor scenes. DIW [6] is a relative depth dataset consists of more than 495k images. Nevertheless, it only provides one single pair of ordinal relationship, which is time-consuming to train relative depth models. It has been verified that training with multiple pairs of ordinal relations is beneficial for learning relative depth [6]. Therefore, our dataset which provides dense relative depth maps in the wild is of great value and will be useful for researchers in this community.

3.2. Learning relative depth

This section presents our method for learning relative depth from monocular images. As shown in Figure 4, we formulate monocular relative depth perception as a regression task. Given a batch of input images I, we learn a non-linear function $z = f(I, \theta)$ parameterized by θ in an end-to-end fashion to regress pixel-wise relative depth. To learn with diverse point pairs of annotations, we adopt online mini-batch sampling, and train these sampled point pairs with an improved ranking loss. In the following we first describe our network architecture, and then discuss efficient mini-batch sampling. Finally, we introduce the loss function that we adopt.

Network architecture Similar to recent works [2, 45], we also use pre-trained ResNet as the backbone. Since ResNet comprises a sequence of convolution (stride is 2) and pooling operations, the receptive field of convolutions is increased to capture more contextual information, while the resolution of output feature maps is decreased. Typically, the size of final feature map is 1/32 of the input image. Thus, a coarse prediction would be generated if directly up-sampling or deconvolution/unpooling on these feature maps. Two alternatives can effectively obtain a finer prediction, one is dilated convolution [46] (or atrous convolution), and the other is multi-scale feature fusion [40, 47]. The former is a common technique to avoid reducing the resolution of feature maps while to retain a large receptive field. However, it costs too much memory and is prone to produce checkerboard artifacts [48]. The latter can save memory and is still able to produce high-resolution predictions. In many applications, *e.g*. 2D-to-3D conversion, visual smoothness is of equal importance to metric measurements. Based on the above observations, we choose the latter one to build our network architecture.



(a) Proposed network

(b) Feature fusion

(c) Residual Conv

(d) Adaptive output

Figure 4. Illustrations of proposed network architecture (a). Our proposed network is based on a feedforward ResNet architecture [49], which generates multi-scale feature maps. To obtain finer predictions, we use a progressive refinement strategy to fuse multi-scale features. (b) shows the process of multi-scale feature fusion, and (c) is a Residual Convolution module. (d) is an Adaptive Convolution module that adjusts the channels of feature maps and the final output.

Our proposed network is illustrated in Figure 4. To apply ResNet to dense per-pixel prediction tasks, we first remove the last pooling layer, fully-connected layer and softmax layer. The feedforward ResNet generates a sequence of feature maps at different scales that have different semantics. Since employing only high-level semantic features would result in coarse predictions, we use a progressive refinement strategy, that merge high level semantic features and low level edge-sensitive features, to get more accurate predictions. Generally, we divide the ResNet into 4 different building blocks according to the resolution of feature maps. In each building block, many feature maps are of the same scale. We choose the output of the last layers of individual building blocks as one input to our multi-scale feature fusion modules. Multi-scale feature fusion modules take two groups of feature maps as input. One is obtained from ResNet, and the other is generated by training from scratch. To conduct progressive refinement operations, we begin with an upsampling operation on the last group of feature maps that generated by ResNet. Following [40], we employ residual convolution blocks so that gradients from high-level layers can be efficiently propagated to low-level layers through short-range and long-range residual connections. For each feature fusion module, we first use a residual convolution block to transfer feature maps from specific layers of pre-trained ResNet for our task, and then merge with fused feature maps that produced by last feature fusion module via *summation*. Finally, an upsampling operation is applied to generate feature maps of the same resolution as the next input. Note that, before each residual convolution block, a transitional 3 × 3 convolution layer is applied to adjust the channel number of feature maps. More specifically, the channel number of each transitional layer is set to 256 in our experiments. To produce final output, we stack an adaptive output module that consists of two convolution layers and a bilinear interpolation layer. In particular, the channels of the stacked convolution layers are 128 and 1, respectively.

Mini-batch sampling Instead of training with fixed point pairs from each image [6], we explore the diversity of samples by online sampling, *i.e.*, we resort to sample pairs online within each mini-batch. For each input image I, we randomly sample N point pairs (i, j), where N is the total number of point pairs, i and j represent the location of the first and second points, respectively. To label ordinal relation ℓ_{ij} between each point pair, we first obtain depth values (g_i, g_j) from corresponding ground-truth depth map, and then define the ground-truth ordinal relation ℓ_{ij} as follows:

$$\ell_{ij} = \begin{cases} +1, & \frac{g_i}{g_j} > 1 + \sigma, \\ -1, & \frac{g_j}{g_i} > 1 + \sigma, \\ 0, & otherwise. \end{cases} \tag{1}$$

where σ is an empirical threshold, and we set it to 0.02 following [33]. Thus, our ground-truth relative depth can be denoted by $G = \{i_k, j_k, \ell_k\}, k = 1, 2, ...N$, where i_k and j_k respectively represent the location of the first and the second point in the k-th pair, and $\ell_k \in \{+1, -1, 0\}$ is the corresponding ground-truth ordinal relationship between i_k and j_k that indicates further (+1), closer (-1), and equal (0). Note that there exists the problem of imbalanced ordinal relations, *i.e.*, the number of equal relation is far less than other two relations.

Loss function To enable our ConvNet to be trained with imbalanced ordinal relations, an appropriate loss function is needed. In this paper, we design an improved ranking loss $L(I, G, z)$, which can be formulated as follows:

$$L(I, G, z) = \sum_{k=1}^{N} \omega_k \phi(I, i_k, j_k, \ell_k, z), \tag{2}$$

where z is the estimated relative depth map, ω_k and $\phi(I, i_k, j_k, \ell_k, z)$ are the weight and loss of the k-th point pair, respectively. Note that ω_k can only be 0 or 1 in our experiments. $\phi(I, i_k, j_k, \ell_k, z)$ takes the form:

$$\phi = \begin{cases} \log(1 + \exp[(-z_{ik} + z_{jk})\ell_k]), & \ell_k \neq 0, \\ (z_{ik} - z_{jk})^2, & \ell_k = 0. \end{cases} \tag{3}$$

We initial all ω_k as 1, then the loss can be seen as a ranking loss [6]. To avoid the difference of two unequal depth values being too large and ease the problem of imbalanced ordinal relations, we first sort the loss of unequal pairs at each iteration, and then ignore the smallest part by setting corresponding ω_k to 0. More specifically, we empirically set the smallest 25% of ω_k to 0. Therefore, the ratio of equal relation would be increased so that the problem of imbalanced ordinal relations can be alleviated. In addition, the ConvNet is thus enforced to focus on a set of hard pairs during training.

4. Experiments

To demonstrate the effectiveness of our proposed method, we conduct extensive experiments on the DIW, NYUDv2, as well as the VOC 2012 dataset. We organize the experiments into three parts: 1) investigating the key components that affect the performance of relative depth prediction, 2) comparing our results with other state-of-the-art approaches on two RGBD datasets, and 3) applying ConvNet pretrained on our ReDWeb dataset to other dense per-pixel prediction tasks (*e.g.*, metric depth estimation and semantic segmentation) to test its generalizability. For the task of relative depth prediction and metric depth estimation, we use the following metrics:

- root mean squared error (rms): $\sqrt{\frac{1}{T}\sum_p (g_p - z_p)^2}$
- average relative error (rel): $\frac{1}{T}\sum_p \frac{|g_p - z_p|}{g_p}$
- average $\log_{10}$ error (log10): $\frac{1}{T}\sum_p |\log_{10} g_p - \log_{10} z_p|$
- accuracy with threshold thr: percentage (%) of z_p s.t. $\max(\frac{g_p}{z_p}, \frac{z_p}{g_p}) = \delta < thr$
- Weighted Human Disagreement Rate [33] (WHDR): $\frac{\sum_{ij} \omega_{ij} \mathbf{1}(\ell_{ij} \neq \tilde{\ell}_{ij,\tau})}{\sum_{ij} \omega_{ij}}$

where τ is the threshold that defines the equality relation between two points. Similar to [6], we decide the equality relation if the difference between two predicted depth values is smaller than τ. ω_{ij} is the human confidence weight for the ij-th pair, ℓ and $\tilde{\ell}$ are the ground-truth human annotations and estimated ordinal relations, respectively. Following [6], we set ω_{ij} to 1. Similarly, Weighted Kinect Disagreement Rate (WKDR) can be computed the same way as WHDR [33]. For conventional metric depth evaluation, g_p and z_p represent the ground-truth and predicted depth of

Network	Baseline	UpProj	Dilation	Ours
WHDR	17.59%	16.60%	16.42%	**15.74%**

Table 2. Results on the validation set of the DIW dataset with different network designs.

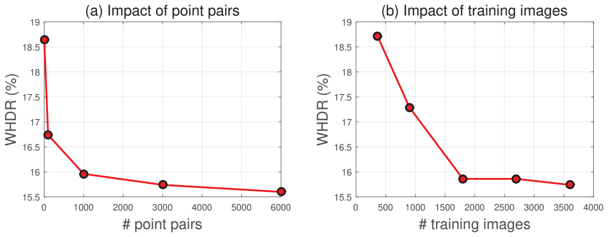


Figure 5. Impact of the amount of point pairs (a) and training images (b).

pixel p, respectively, and T is the total number of pixels in all evaluated images.

We implement our system based on `MatConvNet` [50]. We use the ResNet-50 pre-trained on ImageNet as the backbone, and initial convolutional layers are initialized "from scratch" with simple random Gaussian initialization. During training, data augmentation is performed on-the-fly. Specifically, random cropping and horizontal flipping are applied, and the size of input images fed to the network are 384×384. The stochastic gradient descent (SGD) is used to optimize the network with a mini-batch size of 4 on a Nvidia GTX 1070 GPU. To make use of pre-trained weights, we set the initial learning rate 10 times smaller than the one used for scratch training.

4.1. Ablation Study

We explore four components that affect the performance of relative depth prediction: 1) network architecture, 2) the number of point pairs, 3) the number of training images, and 4) loss function. We train models on our ReDWeb dataset, and report WHDR scores on the validation set of DIW dataset. For the DIW dataset, as in common practice, we use 1.4K images for validation.

Network comparison To analyze the impact of network architecture, we evaluate four types of models: i) a baseline model trained on ResNet-50 (*Baseline*), ii) a model proposed by Laina [2] (*UpProj*), iii) a model that change the dilation rate from 1 to 2 and 4, respectively, in the last two building blocks of ResNet-50 (*Dilation*), and iv) our proposed model that fuses multi-scale features of ResNet-50 (*Ours*). For a fair comparison, we train four models with 3K point pairs of ordinal relations sampled from each input image. Results are shown in Table 2. We observe that our proposed model achieves the best performance. Further, since *Dilation* uses dilated convolution, which introduces zeros in the convolutional kernel, we observe that it is prone to generate checkerboard artifacts.

Number of point pairs To justify the effectiveness of the number of point pairs, we use different amount of point pairs sampled from each image to train a ConvNet. More specifically, we randomly sample 1, 100, 1K, 3K and 6K point pairs per image and train the ConvNet with 3.6K images. Note that, sampling only 1 point pair is the same sampling strategy as in [6]. Figure 5(a) shows the results of our method trained with different number of point pairs. We observe that training with more pairs of ordinal relations improves the performance. Since our method is trained with mini-batch sampling, the diversity of samples would not be a key factor when the sampled point pairs reach a certain number, the performance thus does not increase significantly anymore.

Loss	ℓ_1	ℓ_2	ranking loss	Ours
WHDR	22.27%	21.49%	16.67%	**15.74%**

Table 3. Results on the validation set of the DIW dataset with different loss functions.

Number of training images To study how the number of training images affects the performance of relative depth prediction, we randomly sample a subset of $\{360, 900, 1800, 2700, 3600\}$ images from our ReDWeb dataset. In this experiment, we train our model with 3K point pairs per image. In Figure 5(b), we report the WHDR scores with different ReDWeb subsets. We observe that the WHDR score decreases as the number of training data increases, indicating more training data matter. Note that only 1410 point pairs are used for validation, thus the decrease of WHDR score becomes not that significant.

Loss function We train models on our dataset and validate on the DIW with different loss functions. From Table 3, we find that our improved ranking loss outperforms other loss functions. Per-pixel regression losses (*e.g.*, ℓ_1 or ℓ_2) are effective for metric depth regression but not for ordinal prediction.

4.2. Comparison with state-of-the-art

We compare our method against other state-of-the-art approaches on the DIW and NYUDv2 datasets, respectively. During pre-training, 3.6K images are used from our ReDWeb dataset.

DIW The DIW dataset contains 74K images for testing. For each image, one single pair of points, which only has two possible ordinal relations (further or closer), are used for evaluation. We report the WHDR scores of ten models in Table 4: 1) *Baseline*: a prior that judges ordinal relations by the coordinate of the query points (label the lower point to be closer or randomly guess if the two points are at the same height); 2) *Chen_NYU*: a model trained by Chen *et al.* [6] on the raw NYUDv2 dataset with all available pairs; 3) *Ours_NYU*: our model trained on the raw NYUDv2 dataset with 800 point pairs per image; 4) *Eigen* et al.: a model trained by Eigen *et al.* [27] on the raw NYUDv2

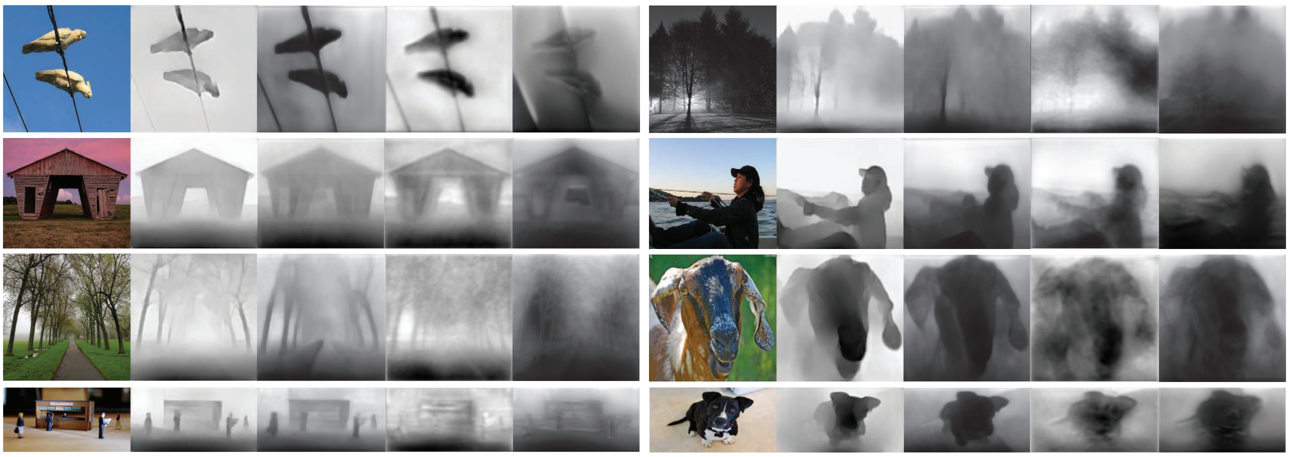


Figure 6. Qualitative comparisons with state-of-the-art results on the DIW dataset. To demonstrate the effectiveness of our ReDWeb dataset, we show some results of our method trained on the ReDWeb dataset only and directly tested on the DIW dataset. Compared to other approaches, our method yields visually more clear and consistent predictions.

Method	WHDR
Baseline	31.37%
Chen_NYU [6]	31.31%
Ours_NYU	27.70%
Eigen *et al.* [27]	25.70%
Chen_DIW [6]	22.14%
Ours_WEB_1	19.01%
Ours_DIW	14.98%
Chen_NYU_DIW [6]	14.39%
Ours_WEB	14.33%
Ours_WEB_DIW	**11.37%**

Table 4. Comparison with state-of-the-art results on the DIW dataset.

dataset; 5) *Chen_DIW*: a model trained by Chen *et al.* [6] on the DIW dataset; 6) *Ours_WEB_1*: our model trained on our ReDWeb dataset with only one point pair per image; 7) *Ours_DIW*: our model trained on the DIW dataset; 8) *Chen_NYU_DIW*: a model by Chen *et al.* [6] pre-trained on the raw NYUDv2 dataset with all available pairs and fine-tuned on the DIW dataset; 9) *Ours_WEB*: our model trained on our ReDWeb dataset with 3K point pairs per image; 10) *Ours_WEB_DIW*: Our model pre-trained on our ReDWeb dataset and fine-tuned on the DIW dataset.

From Table 4, we find that *Ours_WEB_DIW* achieves state-of-the-art performance on the DIW dataset. Even training only on our ReDWeb dataset that contains 3.6K training images so far, our method still outperforms other state-of-the-art approaches. We believe that the performance can be further improved if we keep extending our dataset. We also conclude that training with the NYUDv2 dataset (*Chen_NYU* and *Eigen* et al.), which only contains indoor scenes, do not work well in the wild. By comparing *Ours_NYU* with *Ours_WEB*, we find that our ReDWeb dataset indeed helps relative depth perception in the wild. Some qualitative comparisons with other methods [6] on the DIW dataset are shown in Figure 6, we observe that our predicted relative depth maps are visually more clear and consistent, especially around object boundaries and textureless regions (*e.g.*, sky). Moreover, training a ConvNet with one point pair of ordinal relation (*Ours_WEB_1* and *Chen_DIW*) is prone to yield scattered or edge-conflicting relative depth maps, while training with multiple pairs of ordinal relations leads to smoother and more accurate results. This justifies our motivation that, given multiple pairs of ordinal relations, a ConvNet trained with an appropriate loss (*e.g.* ranking loss or improved ranking loss) gives satisfactory predictions.

NYUDv2 We evaluate ordinal error and metric depth error on the NYUDv2 dataset. For a fair comparison, the same test data are used as Chen *et al.* [6], i.e., around 3K point pairs for each test image are used for ordinal error evaluation. We compare the WKDR scores of different methods: *Zoran* et al., *Chen* et al., *Ours_ranking* and *Ours* are trained on standard NYUDv2 subset with 795 images; *Chen_220K* and *Ours_100K* are trained on the raw NYUDv2 dataset with 220K and 100K images, respectively; *Eigen* et al. is also trained on the raw NYUDv2 dataset with 220K images but designed for metric depth estimation. For metric depth evaluation, we follow [6] to normalize each estimated relative depth map according to the mean and standard deviation of the training set. From Table 5 and Table 6, we find that *Ours* trained with improved ranking loss outperforms other methods on the NYUDv2 subset, and *Ours_100K* matches the state-of-the-art methods using less training images. Figure 7 shows some results of our method compared with Chen *et al.* [6].

Method	WKDR	$\text{WKDR}^{=}$	$\text{WKDR}^{\neq}$
Zoran *et al.* [33]	43.5%	44.2%	41.2%
Ours_ranking	35.8%	36.0%	36.5%
Chen *et al.* [6]	35.6%	36.1%	36.5%
Ours	**33.7%**	**34.6%**	**34.1%**
Chen_220K [6]	**28.3%**	30.6%	**28.6%**
Ours_100K	29.1%	**29.5%**	29.7%
Eigen *et al.* [27]	34.0%	43.3%	29.6%

Table 5. Ordinal error measures on the NYUDv2 dataset (lower is better). Our improved ranking loss outperforms ranking loss over 2% when trained on the NYUDv2 subset.

Method	RMSE	RMSE (log)	RMSE (s.inv)	absrel	sqrrel
Zoran *et al.* [33]	1.20	0.42	-	0.40	0.54
Chen *et al.* [6]	1.13	0.39	0.26	0.36	0.46
Ours_ranking	1.10	0.38	0.23	0.34	0.42
Ours	**1.09**	**0.37**	**0.23**	**0.34**	**0.41**
Chen_220K [6]	1.10	0.38	0.24	0.34	0.42
Ours_100K	**1.07**	**0.36**	**0.22**	**0.33**	**0.39**
Eigen *et al.* [27]	0.64	0.21	0.17	0.16	0.12

Table 6. Metric depth error measures on the NYUDv2 dataset. Details for each metric can be found in [1].

	Accuracy			Error		
	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	rel	log10	rms
Baseline	71.9%	93.2%	98.5%	0.177	0.076	0.733
Wang *et al.* [24]	60.5%	89.0%	97.0%	0.210	0.094	0.745
Liu *et al.* [4]	65.0%	90.6%	97.6%	0.213	0.087	0.759
Ours	**78.1%**	**95.0%**	**98.7%**	**0.155**	**0.066**	**0.660**
Eigen *et al.* [27]	76.9%	95.0%	**98.8%**	0.158	-	0.641
Laina *et al.* [2]	**81.1%**	**95.3%**	**98.8%**	**0.127**	**0.055**	**0.573**

Table 7. Metric depth error measures on the NYUDv2 dataset. All models are trained with ground-truth depth supervision. Note that, the last two models are trained on the raw NYUDv2 dataset.

4.3. Generalizability

We further show the generalizability of our ConvNet. It is first pre-trained on the ReDWeb dataset and then is applied to two other dense per-pixel prediction tasks, i.e., metric depth estimation and semantic segmentation.

Metric depth estimation on NYUDv2 To verify that our ReDWeb-pretrained ConvNet can benefit the task of monocular metric depth estimation, we train a ResNet50-based ConvNet with and without pre-training on the ReDWeb dataset (*i.e.*, *Ours* and *Baseline*), and finetune the ConvNets on the NYUDv2 subset. To reduce overfitting, we use offline data augmentation to generate about 10K images. We formulate metric depth estimation as a regression task, and utilize a robust regression loss (*i.e.*, Tukey's biweight loss). From Table 7, we can see that our ReDWeb-pretrained ConvNet (*i.e.*, *Ours*) can boost the performance of metric depth estimation.

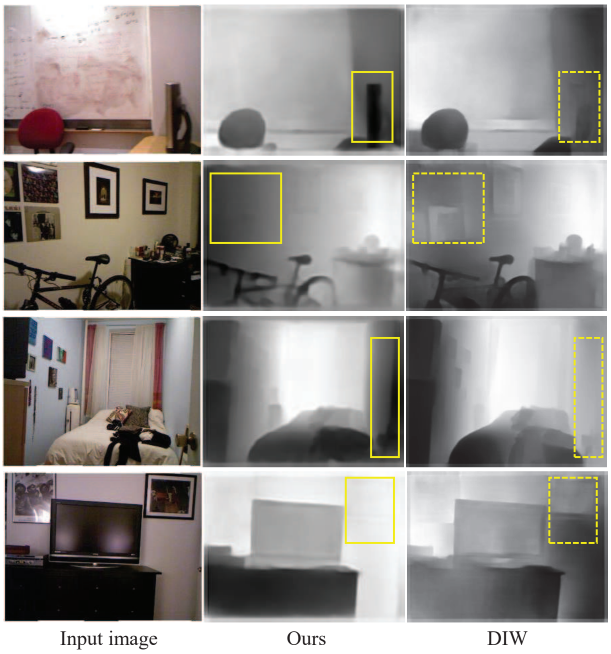


Figure 7. Qualitative comparisons with state-of-the-art results on the NYUDv2 dataset. Details processed by our method (marked in yellow boxes) are visually consistent with input images.

Semantic segmentation on VOC 2012 To explore the applicability of ReDWeb-pretrained ConvNet to the task of semantic segmentation, we finetune the ConvNet on the VOC 2012 dataset. Specifically, we train a ResNet101-based ConvNet with and without pre-training on the ReDWeb dataset, and tune the ConvNets on the union set of VOC 2012 and the Berkeley's extended annotations [51]. According to the results, our ReDWeb-pretrained ConvNet performs better than the one directly finetuned on the VOC 2012 dataset (mIU: 70.82 vs. 69.67).

5. Conclusion

In this paper, we have introduced a method to automatically produce dense relative depth annotations from web stereo images, and proposed a new dataset "ReDWeb" that consists of 3.6K scene-diverse images as well as corresponding dense relative depth maps. To recover relative depth from monocular images, we trained our ConvNet with an improved ranking loss to regress per-pixel relative depth. Experimental results show that our ReDWeb dataset not only helps monocular relative depth estimation in the wild, but also benefits other dense per-pixel prediction tasks. We are still working on extending our dataset.

Acknowledgements This work was supported in part by the National High-tech R&D Program of China (863 Program) under Grant No. 2015AA015904 and in part by the National Natural Science Foundation of China under Grant No. 61502187.

References

[1] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014. 1, 2, 8

[2] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *Proc. IEEE Int. Conf. 3D Vision*, 2016. 1, 2, 4, 6, 8

[3] A. Roy and S. Todorovic, "Monocular depth estimation using neural regression forest," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2016. 1, 2

[4] 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.*, 2016. 1, 2, 8

[5] A. Chakrabarti, J. Shao, and G. Shakhnarovich, "Depth from a single image by harmonizing overcomplete local network predictions," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016. 1, 2

[6] W. Chen, Z. Fu, D. Yang, and J. Deng, "Single-image depth perception in the wild," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016. 1, 2, 3, 4, 5, 6, 7, 8

[7] K. Karsch, C. Liu, and S. B. Kang, "Depthtransfer: Depth extraction from video using non-parametric sampling," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014. 1, 2

[8] J. Shi, Q. Yan, L. Xu, and J. Jia, "Break ames room illusion: Depth from general single images," in *Ann. ACM SIGIR Asia Conf.*, 2015. 1, 2

[9] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, Jul. 2017. 2, 3

[10] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *Proc. Eur. Conf. Comp. Vis.*, 2012. 2, 4

[11] A. Saxena, M. Sun, and A. Y. Ng, "Make3d: Learning 3d scene structure from a single still image," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2009. 2, 4

[12] B. Liu, S. Gould, and D. Koller, "Single image depth estimation from predicted semantic labels," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2010. 2

[13] A. Saxena, A. Ng, and S. Chung, "Learning depth from single monocular images," in *Proc. Adv. Neural Inf. Process. Syst.*, 2005. 2

[14] S. Choi, D. Min, B. Ham, Y. Kim, C. Oh, and K. Sohn, "Depth analogy: Data-driven approach for single image depth estimation using gradient samples," *IEEE Trans. Image Proc.*, 2015. 2

[15] N. Kong and M. J. Black, "Intrinsic depth: Improving depth transfer with intrinsic images," in *Proc. IEEE Int. Conf. Comp. Vis.*, 2015, pp. 3514–3522. 2

[16] J. Konrad, M. Wang, P. Ishwar, C. Wu, and D. Mukherjee, "Learning-based, automatic 2d-to-3d image and video conversion," *IEEE Trans. Image Proc.*, 2013. 2

[17] M. Liu, M. Salzmann, and X. He, "Discrete-continuous depth estimation from a single image," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2014. 2

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105. 2

[19] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015. 2

[20] Y. Cao, Z. Wu, and C. Shen, "Estimating depth from monocular images as classification using deep fully convolutional residual networks," *IEEE Trans. Circuits Syst. Video Technol.*, 2017. 2

[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 *Proc. Eur. Conf. Comp. Vis.*, 2016. 2

[22] 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 *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, June 2015. 2

[23] D. Xu, E. Ricci, W. Ouyang, X. Wang, and N. Sebe, "Multi-scale continuous crfs as sequential deep networks for monocular depth estimation," 2017. 2

[24] 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 *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, June 2015. 2, 8

[25] L. Ladicky, J. Shi, and M. Pollefeys, "Pulling things out of perspective," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2014. 2

[26] C. Hane, L. Ladicky, and M. Pollefeys, "Direction matters: Depth estimation with a surface normal classifier," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015, pp. 381–389. 2

[27] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proc. IEEE Int. Conf. Comp. Vis.*, 2015. 2, 6, 7, 8

[28] A. Mousavian, H. Pirsiavash, and J. Kosecka, "Joint semantic segmentation and depth estimation with deep convolutional networks," *arXiv preprint arXiv:1604.07480*, 2016. 2

[29] R. Garg and I. Reid, "Unsupervised cnn for single view depth estimation: Geometry to the rescue," in *Proc. Eur. Conf. Comp. Vis.*, 2016. 2

[30] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 2

[31] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 2

[32] J. Flynn, I. Neulander, J. Philbin, and N. Snavely, "Deepstereo: Learning to predict new views from the world's imagery," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2016. 2

[33] D. Zoran, P. Isola, D. Krishnan, and W. T. Freeman, "Learning ordinal relationships for mid-level vision," in *Proc. IEEE Int. Conf. Comp. Vis.*, 2015. 2, 5, 8

[34] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li, "Learning to rank: from pairwise approach to listwise approach," in *Proc. Int. Conf. Mach. Learn.*, 2007. 2

[35] W. Chen, D. Xiang, and D. Jia, "Surface normals in the wild," in *Proc. IEEE Int. Conf. Comp. Vis.*, 2017. 2

[36] D. Pathak, R. Girshick, P. Dollár, T. Darrell, and B. Hariharan, "Learning features by watching objects move," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 2

[37] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 328–341, 2008. 3

[38] J. Zbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *J. Mach. Learn. Res.*, vol. 17, 2016. 3

[39] A. Fusiello and L. Irsara, "Quasi-euclidean uncalibrated epipolar rectification," in *Proc. IEEE Int. Conf. Patt. Recogn.* IEEE, 2008, pp. 1–4. 3

[40] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, Jul. 2017. 3, 4, 5

[41] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ade20k dataset," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, Jul. 2017. 3

[42] P. Krähenbühl and V. Koltun, "Parameter learning and convergent inference for dense random fields," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 513–521. 3

[43] J. Xiao, A. Owens, and A. Torralba, "Sun3d: A database of big spaces reconstructed using sfm and object labels," in *Proc. IEEE Int. Conf. Comp. Vis.*, 2013. 4

[44] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger, "Sparsity invariant cnns," in *3*. 4

[45] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 4

[46] 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," *arXiv preprint arXiv:1606.00915*, 2016. 4

[47] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 4

[48] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell, "Understanding convolution for semantic segmentation," *arXiv preprint arXiv:1702.08502*, 2017. 4

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2016. 4

[50] A. Vedaldi and K. Lenc, "Matconvnet – convolutional neural networks for matlab," in *Proc. ACM Int. Conf. Multimedia.*, 2015. 6

[51] B. Hariharan, P. Arbelaez, L. Bourdev, S. Maji, and J. Malik, "Semantic contours from inverse detectors," in *Proc. IEEE Int. Conf. Comp. Vis.*, 2011, pp. 991–998. 8