Improving Monocular Depth Estimation by Leveraging Structural Awareness and Complementary Datasets

Tian Chen[⋆], Shijie An[⋆], Yuan Zhang, Chongyang Ma,
Huayan Wang, Xiaoyan Guo, and Wen Zheng

Y-tech, Kuaishou Technology

Abstract. Monocular depth estimation plays a crucial role in 3D recognition and understanding. One key limitation of existing approaches lies in their lack of structural information exploitation, which leads to inaccurate spatial layout, discontinuous surface, and ambiguous boundaries. In this paper, we tackle this problem in three aspects. First, to exploit the spatial relationship of visual features, we propose a structure-aware neural network with spatial attention blocks. These blocks guide the network attention to global structures or local details across different feature layers. Second, we introduce a global focal relative loss for uniform point pairs to enhance spatial constraint in the prediction, and explicitly increase the penalty on errors in depth-wise discontinuous regions, which helps preserve the sharpness of estimation results. Finally, based on analysis of failure cases for prior methods, we collect a new Hard Case (HC) Depth dataset of challenging scenes, such as special lighting conditions, dynamic objects, and tilted camera angles. The new dataset is leveraged by an informed learning curriculum that mixes training examples incrementally to handle diverse data distributions. Experimental results show that our method outperforms state-of-the-art approaches by a large margin in terms of both prediction accuracy on NYUDv2 dataset and generalization performance on unseen datasets.

1 Introduction

Recovering 3D information from 2D images is one of the most fundamental tasks in computer vision with many practical usage scenarios, such as object localization, scene understanding, and augmented reality. Effective depth estimation for a single image is usually desirable or even required when no additional signal (e.g., camera motion and depth sensor) is available. However, monocular depth estimation (MDE) is well known to be ill-posed due to the many-to-one mapping from 3D to 2D. To address this inherent ambiguity, one possibility is to leverage auxiliary prior information, such as texture cues, object sizes and locations, as well as occlusive and perspective clues [40,21,27].

[⋆] Joint first authors

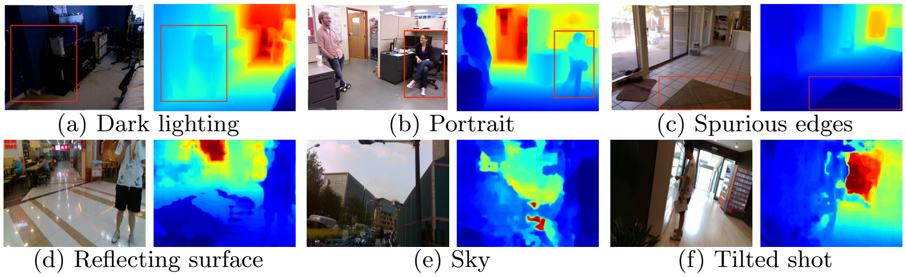


Fig. 1: Six typical hard cases for existing monocular depth estimation methods. (a), (c), and (e) show results of Alhashim *et al.* [1], while (b), (d), and (f) are based on Fu *et al.* [11]. Red boxes highlight inaccurate regions in the results.

More recently, advances in deep convolutional neural network (CNN) have demonstrated superior performance for MDE by capturing these priors implicitly and learning from large-scale dataset [10,9,28,52,36,19]. CNNs often formulate MDE as classification or regression from pixels values without explicitly accounting for global structure. That leads to loss of precision in many cases. To this end, we focus on improving structure awareness in monocular depth estimation.

Specifically, we propose a new network module, named *spatial attention block*, which extracts features via blending cross-channel information. We sequentially adopt this module at different scales in the decoding stage (as shown in Fig. 2a) to generate spatial attention maps which correspond to different levels of detail. We also add a novel loss term, named *global focal relative loss* (GFRL), to ensure sampled point pairs are ordered correctly in depth. Although existing methods attempt to improve the visual consistency between predicted depth and the RGB input, they typically lack the ability to boost performance in border areas, which leads to a large portion of quantitative error and inaccurate qualitative details. We demonstrate that simply assigning larger weights to edge areas in the loss function can address this issue effectively.

Furthermore, MDE through CNNs usually cannot generalize well to unseen scenarios [8]. We find six types of common failure cases as shown in Fig. 1 and note that the primary reason for these failures is the lack of training data, even if we train our network on five commonly used MDE datasets combined. To this end, we collect a new dataset, named *HC Depth Dataset*, to better cover these difficult cases. We also show that an incremental dataset mixing strategy inspired by curriculum learning can improve the convergence of training when we use data following diverse distributions.

To sum up, our main contributions include:

- A novel spatial attention block in the network architecture.
- A new loss term (GFRL) and an edge-aware consistency scheme.
- A new MDE dataset featuring hard cases that are missing or insufficient in existing datasets, and a data mixing strategy for network training.

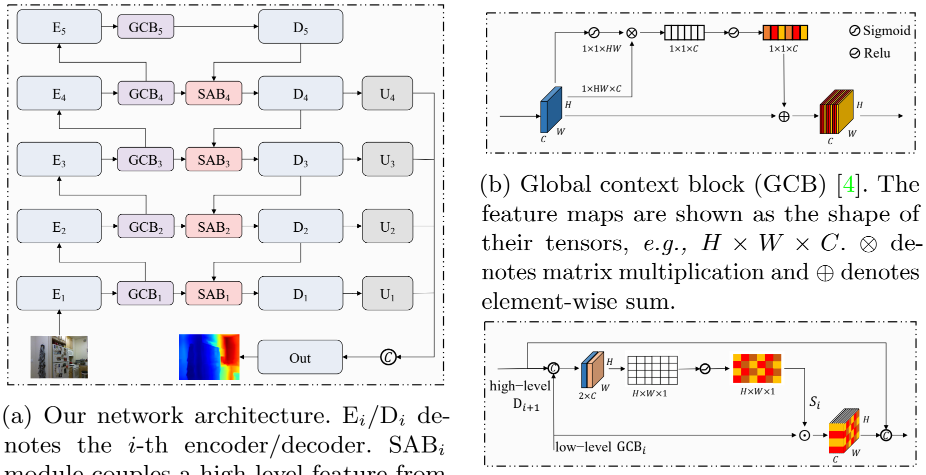
2 Related Work

Monocular depth estimation. Depth estimation from 2D images is an essential step for 3D reconstruction, recognition, and understanding. Early methods for depth estimation are dominated by geometry-based algorithms which build feature correspondences between input images and reconstruct 3D points via triangulation [17,23]. Recently CNN-based approaches for pixel-wise depth prediction [24,9,52] present promising results from a single RGB input, based on supervision with ground-truth training data collected from depth sensors such as LiDAR and Microsoft Kinect camera. By leveraging multi-level contextual and structural information from neural network, depth estimation has achieved very encouraging results [12,25,28,54]. The major limitation of this kind of methods is that repeated pooling operations in deep feature extractors quickly decrease the spatial resolution of feature maps. To incorporate long-range cues which are lost in downsampling operations, a variety of approaches adopt skip connections to fuse low-level depth maps in encoder layers with high-level ones in decoder layers [25,11,52].

Instead of solely estimating depth, several recent multi-task techniques [9,36,19] predict depth map together with other information from a single image. These methods have shown that the depth, normal, and class label information can be jointly and consistently transformed with each other in local areas. However, most of these approaches only consider local geometric properties, while ignoring global constraints on the spatial layout and the relationship between individual objects. The most relevant prior methods to ours are weakly-supervised approaches which consider global relative constraint and use pair-wise ranking information to estimate and compare depth values [5,51,6].

Attention mechanism. Attention mechanisms has been successfully applied to various high-level tasks, such as generative modeling, visual recognition, and object detection [55,47,18]. In addition, attention maps are very useful in pixel-wise tasks. NLNet [49] adopts self-attention mechanism to model the pixel-level pairwise relationship. CCNet [20] accelerates NLNet by stacking two criss-cross blocks, which extract contextual information of the surrounding pixels. Yin *et al.* [53] leverage multi-scale structured attention model which automatically regulates information transferred between corresponding features.

Cross-dataset knowledge transfer. A model trained on one specific dataset generally does not perform well on others due to dataset bias [45]. For MDE, solving different cases, *e.g.* indoor, outdoor, and wild scenes, usually requires explicitly training on diverse datasets [31,6,13,29]. When training on mixed datasets, curriculum learning [3] is needed to avoid the local minimum problem by training the model on *easier* datasets first. Moreover, when the datasets are imbalanced, resampling [15,16,35] is often performed to reshape the data distribution.



(a) Our network architecture. $\mathrm{E}_i/\mathrm{D}_i$ denotes the i-th encoder/decoder. SAB_i module couples a high-level feature from D_{i+1} with a low-layer feature from GCB_i. Ⓒ is the concatenate operation for the output of upsampling blocks $\{\mathrm{U}_i\}$.

(b) Global context block (GCB) [4]. The feature maps are shown as the shape of their tensors, *e.g.*, $H \times W \times C$. $\otimes$ denotes matrix multiplication and $\oplus$ denotes element-wise sum.

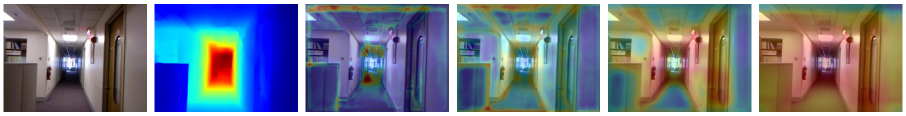
(c) Our spatial attention block (SAB). S_i represents the attention map of SAB_i and $\odot$ denotes element-wise dot product.

Fig. 2: Illustration of our network architecture and the proposed SAB.

3 Our Method

3.1 Network Architecture

We illustrate our network architecture in Fig. 2a. Based on a U-shaped network with an encoder-decoder architecture [33], we add skip connections [48] from encoders to decoders for multi-level feature maps. We observe that encoder mainly extracts semantic features, while decoder pays more attention to spatial information. A light-weight attention based global context block (GCB) [4] (Fig. 2b) is sequentially applied to each residual block in the encoding stage to recalibrate channel-wise features. The recalibrated vectors with global context are then combined with high-level features as input for our spatial attention blocks (SAB) through skip connections. Then, the recalibrated features and the high-level features are fused to make the network focus on either the global spatial information or the detail structure at different stages. From a spatial point of view, the channel attention modules are applied to emphasize semantic information *globally*, while the spatial attention modules focus on where to emphasize or suppress *locally*. With skip connections, these two types of attention blocks build a 3D attention map to guide feature selection. The output features of the four upsampling blocks are upsampled by a factor of 2, 4, 8, and 16, respectively, and then fed to the refinement module to obtain the final depth map which has the same size as the original image. The main purpose of these multi-scale output layers is to fuse information at multiple scales together. The low resolution output retains information with finer global layout, while the high resolution output is used to restore details lost after the downsampling operations.



(a) Input (b) Output (c) S_1 (d) S_2 (e) S_3 (f) S_4

Fig. 3: Spatial attention maps at different scales visualized as heat maps overlaid on the input image. S_i is the ith attention map from the top layer in the decoder (see Fig. 2c). Red color indicates larger value. Our network attends to different levels of structural detail at different scales.

Spatial attention block. Different from channel-attention mechanism which selects semantic feature in network derivation, our SAB is designed to optimize geometric spatial layout in pixel-wise regression tasks. In SAB, we perform a squeeze operation on concatenate features via 1 × 1 convolution to aggregate spatial context across their channel dimensions. Then we activate local attention to get a 2D attention map which encodes pixel-wise depth information over all spatial locations. The low-level features are multiplied by this 2D attention map for subsequent fusion to deliver the spatial context from a higher-level layer. Therefore, our SAB generates attention maps with richer spatial information to recalibrate the semantic features from GCB. The SAB shown in Fig. 2(c) can be formulated as

$$\mathrm{D}_i = f\Big(\sigma\big(W_1 * f(\mathrm{D}_{i+1}, \mathrm{GCB}_i)\big) \odot \mathrm{GCB}_i, \mathrm{D}_{i+1}\Big), \tag{1}$$

where f is a fusion function (e.g. element-wise sum, element-wise dot product, or concatenation), while $*$ denotes 1×1 or 3×3 convolution and $\odot$ denotes element-wise dot product. Since a depth maps has a wide range of positive values, we use ReLU as the activation function $\sigma(x)$.

As shown in Fig. 3, attention feature maps obtained using our SAB help the network focus on the specific information of relative depth across different levels. Specifically, the attention map S_4 contains low-level features which depict the semantic hierarchy and capture the overall near-and-far structure in 3D space. The closer a spatial attention feature map is to the top layer S_1, the more local details are focused for the prediction output.

3.2 Network Training

The loss function to train our network contains four terms, i.e., Berhu loss $\mathcal{L}_B$, scale-invariant gradient loss $\mathcal{L}_g$, normal loss $\mathcal{L}_n$, and global focal relative loss $\mathcal{L}_r$. We describe each loss term in detail as follows.

BerHu loss. The BerHu loss refers to the reversed Huber penalty [58] of depth residuals, which provides a reasonable trade-off between L1 norm and L2 norm in regression tasks when the errors present a heavy-tailed distribution [39,28].

Therefore, the BerHu loss $\mathcal{L}_B$ is commonly used as the basic error metric for MDE and is defined as:

$$\mathcal{L}_B = \sum_{i,j} |d_{i,j} - \hat{d}_{i,j}|_b, \ |x|_b = \begin{cases} |x| & |x| \leq c, \\ \frac{x^2+c^2}{2c} & |x| > c \end{cases}, \tag{2}$$

where $d_{i,j}$ and $\hat{d}_{i,j}$ are the ground-truth and predicted depth values at the pixel location (i, j), respectively. We set c to be $0.2 \max_p(|\hat{d}_p - d_p|)$, where $\{p\}$ indicate all the pixels in one batch.

Scale-invariant gradient loss. We use scale-invariant gradient loss [46] to emphasize depth discontinuities at object boundaries and to improve smoothness in homogeneous regions. To cover gradients at different scales in our model, we use 5 different spacings $\{s = 1, 2, 4, 8, 16\}$ for this loss term $\mathcal{L}_g$:

$$\begin{aligned} \mathcal{L}_g &= \sum_s \sum_{i,j} |\mathbf{g}_s(i,j) - \hat{\mathbf{g}}_s(i,j)|^2, \\ \mathbf{g}_s(i,j) &= \left(\frac{d_{i+s,j} - d_{i,j}}{|d_{i+s,j} + d_{i,j}|}, \frac{d_{i,j+s} - d_{i,j}}{|d_{i,j+s} + d_{i,j}|} \right)^\top, \end{aligned} \tag{3}$$

Normal loss. To deal with small-scale structures and to further improve high-frequency details in the predicted depth, we also use a normal loss term $\mathcal{L}_n$:

$$\mathcal{L}_n = \sum_{i,j} \left(1 - \frac{\langle \mathbf{n}_{i,j}, \hat{\mathbf{n}}_{i,j} \rangle}{\sqrt{\langle \mathbf{n}_{i,j}, \mathbf{n}_{i,j} \rangle} \cdot \sqrt{\langle \hat{\mathbf{n}}_{i,j}, \hat{\mathbf{n}}_{i,j} \rangle}} \right), \tag{4}$$

in which $\langle \cdot, \cdot \rangle$ denotes the inner product of two vectors. The surface normal is denoted as $\mathbf{n}_{i,j} = [-\nabla_x, -\nabla_y, 1]^\top$, where ∇_x and ∇_y are gradient vectors along the x and y-axis in the depth map, respectively.

Global focal relative loss. Relative loss (RL) [5,51,30,6] is used to make depth-wise ordinal relations between sample pairs predicted by the network consistent with the ground-truth. Inspired by the focal loss defined in Lin *et al.* [32], we propose an improved version of relative loss, named *global focal relative loss* (GFRL), to put more weight on sample pairs of incorrect ordinal relationships in the prediction. To ensure uniform selection of point pairs, we subdivide the image into 16×16 blocks of the same size and randomly sample one point from each block. Each point is compared with all the other points from the same image when training the network. Our loss term of n pairs is formally defined as $\mathcal{L}_r = \sum_k^n \mathcal{L}_{r,k}$. The k-th pair loss term $\mathcal{L}_{r,k}$ is

$$\begin{aligned} \mathcal{L}_{r,k} &= \begin{cases} w_k^\gamma \log\Big(1 + \exp\big(-r_k\,(d_{1,k} - d_{2,k})\big)\Big), & r_k \neq 0 \\ (d_{1,k} - d_{2,k})^2, & r_k = 0 \end{cases} \\ w_k &= 1 - 1/\big(1 + \exp\big(-r_k\,(d_{1,k} - d_{2,k})\big)\big), \end{aligned} \tag{5}$$

where r_k is the ground-truth ordinal relationship and is set to -1, 0, or 1, if the first point has a smaller, equal, or a larger depth value compared to the second point. The equality holds if and only if the depth difference ratio is smaller than a threshold of 0.02.

In Eqn. 5, our key idea is to introduce a modulating factor w_k^{γ} to scale the relative loss. Intuitively, when a pair of pixels have incorrect ordinal relationship in the prediction, the loss is unaffected since w_k is close to 1. If the depth ordinal relationship is correct and the depth difference is large enough, the weight w_k on this pair will go to 0. The parameter γ smoothly adjusts the magnitude of weight reduction on easy point pairs. When $\gamma = 0$, our GFRL is equivalent to RL [5]. As γ increases, the impact of the modulating factor becomes larger and we set $\gamma = 2$ in our experiments. It turns out our GFRL outperforms RL under various evaluation metrics (see Sec. 5.2).

Total loss. We train our network in a sequential fashion for better convergence by using different combinations of loss terms in different stages [19]. Our total loss function $\mathcal{L}_{total}$ is defined as:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_B^{\text{I-III}} + \lambda_2 \mathcal{L}_g^{\text{II-III}} + \lambda_3 \mathcal{L}_n^{\text{III}} + \lambda_4 \mathcal{L}_r^{\text{III}}, \tag{6}$$

where $\{\lambda_i\}$ are the weights of different loss terms, and the superscripts I-III denote the stages of using the corresponding terms. BerHu loss is the basic term being used to start the training. After convergence, we first add gradient loss for smooth surface and sharp edges. To further improve the details of predicted depth and refine the spatial structure, we add normal loss and global focal relative loss in the final stage.

Edge-aware consistency. Depth discontinuities typically arise at the boundaries of occluding objects in a scene. Most existing MDE methods cannot recover these edges accurately and tend to generate distorted and blurry object boundaries [19]. Besides, we observe that in these areas, the average prediction error is about 4 times larger than that of other areas.

Based on these observations, we introduce an edge-aware consistency scheme to preserve sharp discontinuities in depth prediction results. Specifically, we first use Canny edge detector [2] to extract edges for the ground-truth depth map, and then dilate these edges with a kernel of 5 to get a boundary mask.We multiply the loss $\mathcal{L}_{ij}$ at the pixel p_{ij} by a weight of 5 if p_{ij} is an edge pixel in the boundary mask. Our edge-aware consistency scheme can be considered as a hard example mining method [41] to explicitly increase the penalty on prediction errors in boundary regions.

4 Datasets

4.1 HC Depth Dataset

In recent years, several RGBD datasets have been proposed to provide collections of images with associated depth maps. In Tab. 1, we list five open source RGBD

Table 1: Properties of different monocular depth estimation datasets. The last column denotes types of hard cases (see Fig. 1) included in these datasets.

ID	Datasets	Type	Annotation	Images	Hard cases
1	NYU	Indoor	Kinect	108,644	a,b,c
2	ScanNet	Indoor	Kinect	99,571	a,c,f
3	SUNCG	Indoor	Synthetic	454,078	-
4	CAD	Portrait	Kinect	145,155	b
5	URFall	Portrait	Kinect	10,764	b
6	HC Depth	Mixed	Kinect & RealSense	120,060	a,b,c,d,e,f

datasets and summarize their properties such as types of content, annotation methods, and number of images. Among them, NYUDv2 [34], ScanNet [7], CAD [37], and URFall [26] are captured from real indoor scenes using Microsoft Kinect [56], while SUNCG [42] is a synthetic dataset collected by rendering manually created 3D virtual scenes. In addition, CAD and URFall contain videos of humans performing activities in indoor environments. These datasets offer a large number of annotated depth images and are widely used to train models for an MDE task. However, each of these RGBD datasets primarily focuses on only one type of scenes and may not cover enough challenging cases. In Fig. 1, we identify and summarize six types of typical failure cases for two state-of-the-art MDE methods [1,11] trained on the NYUDv2 dataset.

To complement existing RGBD datasets and provide sufficient coverage on challenging examples for the MDE task, we design and acquire a new dataset, named *HC Depth dataset*, which contains all the six categories of hard cases shown in Fig. 1. Specifically, we collect 24660 images using Microsoft Kinect [56], which provides dense and accurate depth maps for the corresponding RGB images. Due to the limited effective distance range of Kinect, these images are mainly about indoor scenes of portraits. We also collect 95400 images of both indoor and outdoor scenes using Intel RealSense [22], which is capable of measuring larger depth range in medium precision. In *sky* cases, we assign a predefined maximum depth value to sky regions based on semantic segmentation [57]. We also perform surface smoothing and completion for all the cases using the toolbox proposed by Silberman *et al.* [34]. We show several typical examples of our HC Depth dataset in the supplementary materials.

4.2 Incremental Dataset Mixing Strategy

Training on aforementioned datasets together poses a challenge due to the different distributions of depth data in various scenes. Motivated by curriculum learning [3] in global optimization of non-convex functions, we propose an incremental dataset mixing strategy to accelerate the convergence of network training and improve the generalization performance of trained models.

Curriculum learning is related to boosting algorithms, in which difficult examples are gradually added during the training process. In our case of MDE, we divide all the training examples into four main categories based on the content and difficulty, i.e., indoor (I), synthetic (S), portrait (PT), and hard cases (HC).

First, we train our model on datasets with similar distributions (e.g., I + S) until convergence. Then we add remaining datasets (e.g., PT or HC) one by one and build a new sampler for each batch to ensure a balanced sampling from these imbalanced datasets. Specifically, we count the number of images k_i contained in each dataset, and $K = \sum_i k_i$ is the total number of training images. The probability of sampling an image from the i-th dataset is proportional to K/k_i to effectively balance different datasets.

5 Experiments

5.1 Experimental Setup

To validate each algorithm component, we first train a baseline model $\mathcal{M}_b$ without any module introduced in Sec. 3. We denote the model trained with edge-aware consistency as $\mathcal{M}_e$ and the model trained with both edge-aware consistency and our spatial attention blocks as $\mathcal{M}_{e,SAB}$. The model obtained with all the components is denoted as $\mathcal{M}_{full}$, which is essentially $\mathcal{M}_{e,SAB}$ trained with the addition of GFRL and $\mathcal{M}_{e,\mathcal{L}_r}$ trained with the addition of SAB. We also compare with the option to train $\mathcal{M}_{e,SAB}$ with an additional relative loss $\mathcal{L}_r$ described in Chen *et al.* [5]. We show additional comparisons with variations of existing attention modules, e.g., Spatial Excitation Block (sSE) [38] and Convolutional Block Attention Module (CBAM) [50]. Finally, we train the full model on different combinations of datasets to evaluate the effect of adding more training data.

We implement our method using PyTorch and train the models on four NVIDIA GPUs with 12GB memory. We initialize all the ResNet-101 blocks with pretrained ImageNet weights and randomly initialize other layers. We use RMSProp [44] with a learning rate of 10^{-4} for all the layers and reduce the rate by 10% after every 50 epochs. β_1, β_2 and weight decay are set to 0.9, 0.99, and 0.0001 respectively. The batch size is set to be 48. The weights $\{\lambda_i\}$ of the loss terms described in Sec. 3.2 are set to 1, 1, 1, and 0.5 respectively. We pretrain the baseline model $\mathcal{M}_b$ for 40 epochs on NYUDv2 dataset. The total number of trainable parameters for our full model is about 167M. When training on multiple datasets, we first use our multi-stage algorithm (Sec. 3.2) on the first dataset until convergence of the total loss function, and then include more data via our incremental dataset mixing strategy (Sec. 4.2).

We compare our method with several state-of-the-art MDE algorithms both quantitatively and qualitatively using the following commonly used metrics [10]:

- Average relative error (REL): $\frac{1}{n}\sum_p^n \left|d_p - \hat{d}_p\right| / \hat{d}_p$.
- Root mean squared error (RMSE): $\sqrt{\frac{1}{n}\sum_p^n \left(d_p - \hat{d}_p\right)^2}$.
- Average log10 error: $\frac{1}{n}\sum_p^n \left|\log_{10}\left(d_p\right) - \log_{10}\left(\hat{d}_p\right)\right|$.
- Threshold accuracy: $\{\delta_i\}$ are ratios of pixels $\{d_p\}$ s.t. $\max\left(\frac{d_p}{\hat{d}_p}, \frac{\hat{d}_p}{d_p}\right) < thr_i$ for $thr_i = 1.25, 1.25^2, \text{and } 1.25^3$.

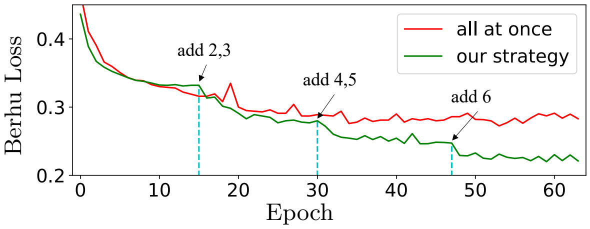


Fig. 4: BerHu loss curves of different training strategies. The red curve denotes the loss curve with all datasets added from the beginning, while the green curve represents the loss curve based on our incremental dataset mixing strategy. The blue dot lines indicate the epochs when new datasets are included. The dataset IDs are defined in Tab. 1.

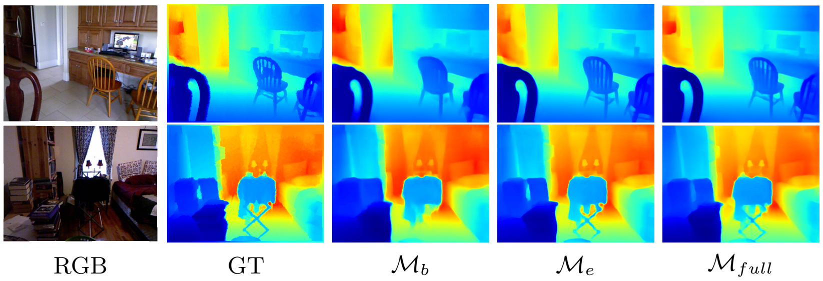


Fig. 5: Evaluation results of our edge-aware consistency module. From left to right: input images, ground-truth depth, results of our baseline model $\mathcal{M}_b$, the model $\mathcal{M}_e$ with our edge-aware consistency module, and our full model $\mathcal{M}_{full}$.

5.2 Experimental Results

Results on NYUDv2 Dataset. The NYUDv2 dataset contains 464 indoor scenes. We follow the same train/test split as previous work, i.e., to use about 50K images from 249 scenes for training and 694 images from 215 scenes for testing. In Fig. 4, we compare the BerHu loss curves on the test set when training on multiple datasets *without* and *with* our incremental dataset mixing strategy (Sec. 4.2), which illustrates that our strategy considerably improves the convergence of network training.

Tab. 2 summarizes the quantitative results of our ablation study on NYUDv2 dataset together with the numbers reported in previous work. Our baseline model $\mathcal{M}_b$ uses ResNet101 as the backbone and couples with GCB modules, combining several widely used loss terms (Berhu loss, scale-invariant gradient loss, and normal loss) of previous work. As can be seen from the table, all of our algorithm components can improve depth estimation results noticeably, including edge-aware consistency, SAB, and GFRL. Furthermore, adding our HC Depth dataset with our dataset mixing strategy (Sec. 4.2) significantly improves the model performance (see $\mathcal{M}_b \rightarrow 1{,}6$ and $\mathcal{M}_{full} \rightarrow 1{,}6$). By comparing $\mathcal{M}_{full}$ with

Table 2: Quantitative results on NYUDv2 dataset using different quantitative metrics. Higher numbers indicate better accuracy, while lower ones represent better results in terms of predication errors. The numbers after the right arrow indicate the IDs of datasets (defined in Tab. 1) to train each model.

Methods	Accuracy ↑			Error ↓		
	δ_1	δ_2	δ_3	RMSE	REL	log10
Eigen *et al.* [9] → 1	0.769	0.950	0.988	0.641	0.158	-
Laina *et al.* [28] → 1	0.811	0.953	0.988	0.573	0.127	0.055
Qi *et al.* [36] → 1	0.834	0.960	0.990	0.569	0.128	0.057
Hao *et al.* [14] → 1	0.841	0.966	0.991	0.555	0.127	0.053
Hu *et al.* [19] → 1	0.866	0.975	0.993	0.530	0.115	0.050
Fu *et al.* [11] → 1	0.828	0.965	0.992	0.509	0.115	0.051
Alhashim *et al.* [1] → 1	0.846	0.974	0.994	0.465	0.123	0.053
Yin *et al.* [54] → 1	0.875	0.976	0.994	0.416	0.108	0.048
$\mathcal{M}_b$ → 1	0.856	0.974	0.994	0.430	0.120	0.051
$\mathcal{M}_e$ → 1	0.860	0.974	0.994	0.426	0.118	0.050
$\mathcal{M}_{e,SAB}$ → 1	0.864	0.971	0.993	0.417	0.113	0.049
$\mathcal{M}_{full}$ → 1	0.876	0.979	0.995	0.407	0.109	0.047
$\mathcal{M}_{e,SAB} + \mathcal{L}_r$ [5] → 1	0.864	0.971	0.993	0.418	0.113	0.049
$\mathcal{M}_{e,\mathcal{L}_r}$+ sSE [38] → 1	0.857	0.968	0.992	0.433	0.117	0.051
$\mathcal{M}_{e,\mathcal{L}_r}$+ CBAM [50] → 1	0.860	0.968	0.992	0.432	0.117	0.050
$\mathcal{M}_b$ → 1,6	0.868	0.976	0.995	0.420	0.115	0.049
$\mathcal{M}_b$ → 1-6	0.874	0.978	0.995	0.414	0.111	0.048
$\mathcal{M}_{full}$ → 1-3	0.888	0.982	**0.996**	0.391	0.104	0.044
$\mathcal{M}_{full}$ → 1-5	0.888	0.981	**0.996**	0.390	0.103	0.044
$\mathcal{M}_{full}$ → 1,6	0.885	0.979	0.994	0.401	0.104	0.045
$\mathcal{M}_{full}$ → 1-6	**0.899**	**0.983**	**0.996**	**0.376**	**0.098**	**0.042**

$\mathcal{M}_{e,SAB} + \mathcal{L}_r$ [5], we can conclude that GFRL leads to better results than the alternative relative loss [5]. We also show that our SAB ($\mathcal{M}_{full}$) brings notable improvement over Spatial Excitation Block ($\mathcal{M}_{e,\mathcal{L}_r}$+ sSE [38]) and Convolutional Block Attention Module ($\mathcal{M}_{e,\mathcal{L}_r}$+ CBAM [50]). Finally, training our full model $\mathcal{M}_{full}$ on multiple datasets can achieve the best results and outperforms training solely on NYUDv2 dataset by a large margin.

Fig. 6 presents qualitative results on three test cases to compare our full model with three prior methods [11,19,1]. The first example shows that our results have more reasonable spatial layout. In the second example, our model provides more accurate estimation than other methods in the regions of color boundaries. In the third example, our results preserve details in the region around the chandelier. Although Fu *et al.* [11] achieves structural patterns similar to our method, their results contain disordered patterns and thus have much larger errors. Fig. 5 shows evaluation results on two more test examples from NYUDv2 to compare our baseline model $\mathcal{M}_b$ with the model $\mathcal{M}_e$ trained with edge-aware consistency and our full model $\mathcal{M}_{full}$. Our edge-aware consistency module leads to much more accurate boundaries in the depth estimation results, such as the back of the chair and legs of the desk.

Results on TUM Dataset. We use the open source benchmark TUM [43] to evaluate the generalization performance of different methods in the setting

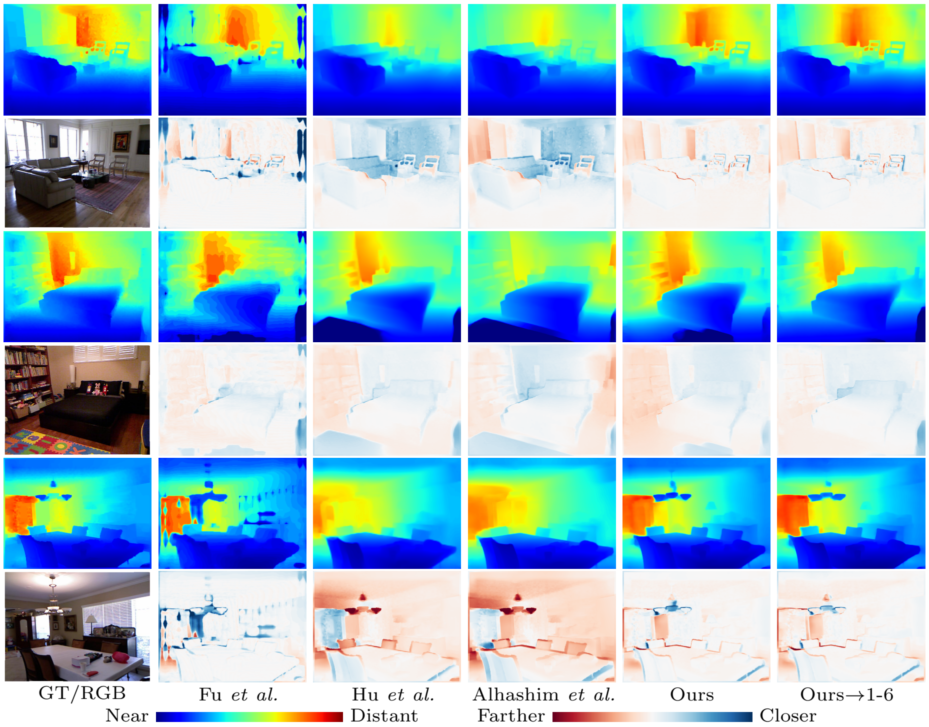


Fig. 6: Qualitative results on four test cases from NYUDv2 Dataset. In each example, the first row shows the ground-truth depth map, results of three prior methods [11,19,1], and the results of our full model trained solely on NYUDv2 and on the combination of all the six datasets listed in Tab. 1, respectively. The second row of each example shows the input image and the error maps of the corresponding results in the first row.

of zero-shot cross-dataset inference. The test set of TUM consists of 1815 high quality images taken in a factory including pipes, computer desks and people, which are never seen when we train our models. Tab. 3 demonstrates that our full model $\mathcal{M}_{full}$ trained solely on NYUDv2 dataset outperforms previous methods. Furthermore, training the full model by adding HC Depth dataset or more datasets using our dataset mixing strategy can significantly improve the generalization performance. As shown in the qualitative results in Fig. 7, our method retains sharp edges of plants in the first example and leads to more accurate spatial layout in the second and third examples.

Results on HC Depth Dataset. To test the performance of different models on hard cases, we use the test split of our HC Depth dataset which contains 328 examples. Tab. 4 summarizes the corresponding quantitative results, from which we can obtain consistent findings with Tab. 3. Fig. 8 illustrates qualitative results

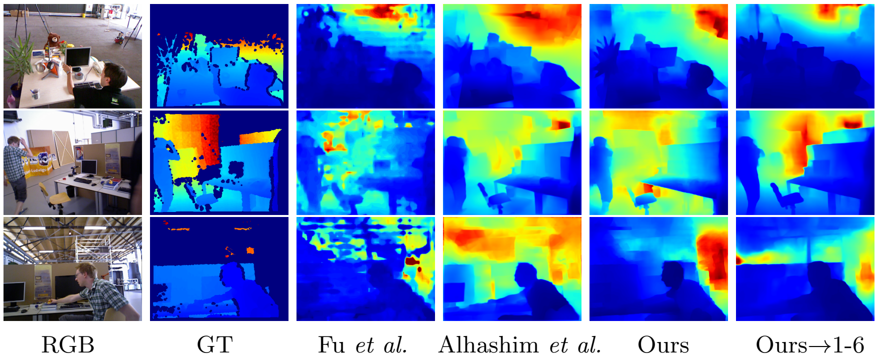


Fig. 7: Qualitative results on TUM dataset.

Table 3: Quantitative results of generalization test on TUM dataset. The experimental configuration is the same as Tab. 2.

Methods	Accuracy ↑			Error ↓		
	δ_1	δ_2	δ_3	RMSE	REL	log10
Hu *et al.* [19] → 1	0.577	0.842	0.932	1.154	0.216	0.111
Fu *et al.* [11] → 1	0.598	0.855	0.934	1.145	0.209	0.110
Alhashim *et al.* [1] → 1	0.567	0.847	0.920	1.250	0.224	0.115
$\mathcal{M}_{full}$ →1	0.606	0.888	0.947	1.109	0.212	0.100
$\mathcal{M}_{full}$ → 1-3	0.665	0.903	0.955	1.031	0.194	0.091
$\mathcal{M}_{full}$ → 1-5	0.710	0.913	0.952	1.013	**0.175**	0.084
$\mathcal{M}_{full}$ →1,6	0.689	0.906	0.950	1.029	0.189	0.086
$\mathcal{M}_{full}$ → 1-6	**0.735**	**0.927**	**0.959**	**0.912**	0.177	**0.081**

of six examples from the test set, which correspond to the six types of hard cases in Fig. 1. As shown in Fig. 8 and Tab. 4, our method based on NYUDv2 dataset already provides more faithful prediction compared to Alhashim *et al.* [1] and the predicted depth can be further improved considerably by adding our HC Depth dataset or using all the datasets through our dataset mixing strategy.

6 Conclusions

In this paper we put together a series of coherent efforts to improve the structural awareness in monocular depth estimation, with the effectiveness and necessity of each component thoroughly verified. We introduce a novel encoder-decoder architecture using the spatial attention mechanism, and boost the network performance by proposing a global focal relative loss and an edge-aware consistency module. We further collect a dataset of hard cases for the task of depth estimation and leverage a data mixing strategy based on curriculum learning for effective network training. We validate each component of our method via comprehensive ablation studies and demonstrate substantial advances over state-of-the-art approaches on benchmark datasets. Our experimental results show that truly generic models for

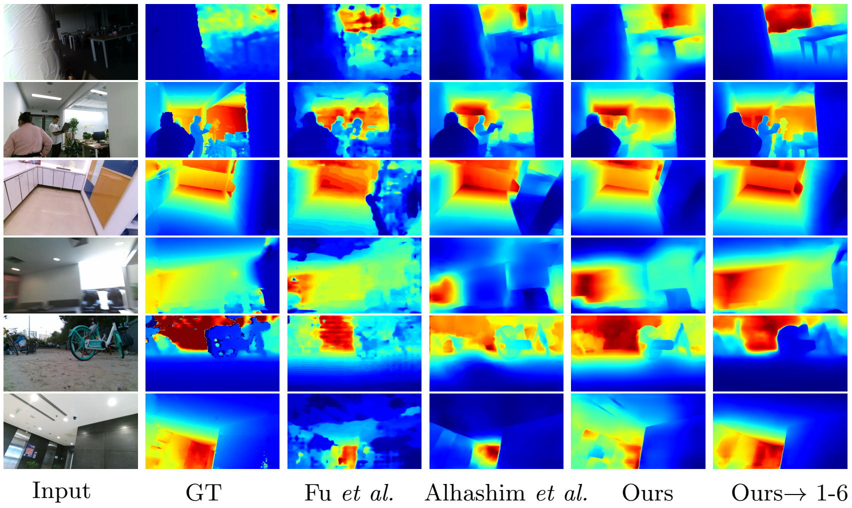


Fig. 8: Qualitative results on our HC Depth dataset. From top to bottom, the examples correspond to the six types of hard cases showed in Fig. 1.

Table 4: Quantitative results on the HC Depth dataset. The experimental configuration is the same as Tab. 2.

Methods	Accuracy ↑			Error ↓		
	δ_1	δ_2	δ_3	RMSE	REL	log10
Hu *et al.* [19] → 1	0.531	0.783	0.898	1.276	0.285	0.128
Fu *et al.* [11] → 1	0.477	0.755	0.866	1.356	0.2962	0.145
Alhashim *et al.* [1] → 1	0.551	0.819	0.918	1.137	0.257	0.118
$\mathcal{M}_{full}$ →1	0.600	0.843	0.930	1.070	0.249	0.109
$\mathcal{M}_{full}$ → 1-3	0.610	0.837	0.921	1.072	0.244	0.108
$\mathcal{M}_{full}$ → 1-5	0.633	0.845	0.924	1.065	0.237	0.107
$\mathcal{M}_{full}$ →1,6	0.825	0.895	0.961	0.715	0.190	0.087
$\mathcal{M}_{full}$ → 1-6	**0.879**	**0.965**	**0.988**	**0.566**	**0.113**	**0.048**

monocular depth estimation require not only innovations in network architecture and training algorithm, but also sufficient data for various scenarios.

Our source code, pretrained models, and the HC Depth dataset will be released to encourage follow-up research. In the future, we plan to capture more diverse scenes and further expand our HC Depth dataset. We would also like to deploy our models on mobile devices for several applications such as augmented reality.

Acknowledgements. We would like to thank the anonymous reviewers for their valuable comments, Jiwen Liu for help on preparing our dataset, and Miao Xuan for help on paper proofreading.

References

1. Alhashim, I., Wonka, P.: High Quality Monocular Depth Estimation via Transfer Learning. arXiv preprint arXiv:1812.11941 (2018)
2. Bao, P., Zhang, L., Wu, X.: Canny edge detection enhancement by scale multiplication. IEEE transactions on pattern analysis and machine intelligence **27**(9), 1485–1490 (2005)
3. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th annual international conference on machine learning. pp. 41–48. ACM (2009)
4. Cao, Y., Xu, J., Lin, S., Wei, F., Hu, H.: GCNet: Non-local networks meet squeeze-excitation networks and beyond. arXiv preprint arXiv:1904.11492 (2019)
5. Chen, W., Fu, Z., Yang, D., Deng, J.: Single-image depth perception in the wild. In: Advances in neural information processing systems. pp. 730–738 (2016)
6. Chen, W., Qian, S., Deng, J.: Learning single-image depth from videos using quality assessment networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5604–5613 (2019)
7. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5828–5839 (2017)
8. Dijk, T.v., Croon, G.d.: How do neural networks see depth in single images? In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2183–2191 (2019)
9. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: Proceedings of the IEEE international conference on computer vision. pp. 2650–2658 (2015)
10. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: Advances in neural information processing systems. pp. 2366–2374 (2014)
11. Fu, H., Gong, M., Wang, C., Batmanghelich, K., Tao, D.: Deep ordinal regression network for monocular depth estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2002–2011 (2018)
12. Garg, R., BG, V.K., Carneiro, G., Reid, I.: Unsupervised cnn for single view depth estimation: Geometry to the rescue. In: European Conference on Computer Vision. pp. 740–756. Springer (2016)
13. Gordon, A., Li, H., Jonschkowski, R., Angelova, A.: Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. arXiv preprint arXiv:1904.04998 (2019)
14. Hao, Z., Li, Y., You, S., Lu, F.: Detail preserving depth estimation from a single image using attention guided networks. In: 2018 International Conference on 3D Vision (3DV). pp. 304–313. IEEE (2018)
15. He, H., Garcia, E.A.: Learning from imbalanced data. IEEE Transactions on knowledge and data engineering **21**(9), 1263–1284 (2009)
16. He, H., Ma, Y.: Imbalanced learning: foundations, algorithms, and applications. John Wiley & Sons (2013)
17. Hirschmuller, H.: Stereo processing by semiglobal matching and mutual information. IEEE Transactions on pattern analysis and machine intelligence **30**(2), 328–341 (2008)
18. Hu, H., Gu, J., Zhang, Z., Dai, J., Wei, Y.: Relation networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3588–3597 (2018)

19. Hu, J., Ozay, M., Zhang, Y., Okatani, T.: Revisiting single image depth estimation: toward higher resolution maps with accurate object boundaries. In: WACV. pp. 1043–1051 (2019)
20. Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., Liu, W.: CCNet: Criss-Cross Attention for Semantic Segmentation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 603–612 (2019)
21. Karsch, K., Liu, C., Kang, S.B.: Depth transfer: Depth extraction from video using non-parametric sampling. IEEE transactions on pattern analysis and machine intelligence **36**(11), 2144–2158 (2014)
22. Keselman, L., Iselin Woodfill, J., Grunnet-Jepsen, A., Bhowmik, A.: Intel realsense stereoscopic depth cameras. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 1–10 (2017)
23. Khamis, S., Fanello, S., Rhemann, C., Kowdle, A., Valentin, J., Izadi, S.: Stereonet: Guided hierarchical refinement for real-time edge-aware depth prediction. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 573–590 (2018)
24. Kong, S., Fowlkes, C.: Pixel-wise attentional gating for scene parsing. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 1024–1033. IEEE (2019)
25. Kuznietsov, Y., Stuckler, J., Leibe, B.: Semi-supervised deep learning for monocular depth map prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6647–6655 (2017)
26. Kwolek, B., Kepski, M.: Human fall detection on embedded platform using depth maps and wireless accelerometer. Computer methods and programs in biomedicine **117**(3), 489–501 (2014)
27. Ladicky, L., Shi, J., Pollefeys, M.: Pulling things out of perspective. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 89–96 (2014)
28. Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., Navab, N.: Deeper depth prediction with fully convolutional residual networks. In: 3D Vision (3DV), 2016 Fourth International Conference on. pp. 239–248. IEEE (2016)
29. Lasinger, K., Ranftl, R., Schindler, K., Koltun, V.: Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-Shot Cross-Dataset Transfer. arXiv:1907.01341 (2019)
30. Lee, J.H., Kim, C.S.: Monocular depth estimation using relative depth maps. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9729–9738 (2019)
31. Li, Z., Dekel, T., Cole, F., Tucker, R., Snavely, N., Liu, C., Freeman, W.T.: Learning the depths of moving people by watching frozen people. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4521–4530 (2019)
32. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
33. Mayer, N., Ilg, E., Hausser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T.: A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4040–4048 (2016)
34. Nathan Silberman, Derek Hoiem, P.K., Fergus, R.: Indoor Segmentation and Support Inference from RGBD Images. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 746–760 (2012)

35. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Learning and transferring mid-level image representations using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1717–1724 (2014)
36. Qi, X., Liao, R., Liu, Z., Urtasun, R., Jia, J.: GeoNet: Geometric Neural Network for Joint Depth and Surface Normal Estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 283–291 (2018)
37. Robot Learning Lab at Cornell University: Cornell Activity Datasets: CAD-60 & CAD-120 (2019), http://pr.cs.cornell.edu/humanactivities/data.php
38. Roy, A.G., Navab, N., Wachinger, C.: Concurrent spatial and channel 'squeeze & excitation'in fully convolutional networks. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 421–429. Springer (2018)
39. Roy, A., Todorovic, S.: Monocular depth estimation using neural regression forest. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5506–5514 (2016)
40. Saxena, A., Chung, S.H., Ng, A.Y.: Learning depth from single monocular images. In: Advances in neural information processing systems. pp. 1161–1168 (2006)
41. Shrivastava, A., Gupta, A., Girshick, R.: Training region-based object detectors with online hard example mining. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 761–769 (2016)
42. Song, S., Yu, F., Zeng, A., Chang, A.X., Savva, M., Funkhouser, T.: Semantic scene completion from a single depth image. Proceedings of 30th IEEE Conference on Computer Vision and Pattern Recognition (2017)
43. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of RGB-D SLAM systems. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 573–580. IEEE (2012)
44. Tieleman, T., Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning **4**(2), 26–31 (2012)
45. Torralba, A., Efros, A.A., et al.: Unbiased look at dataset bias. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1521–1528 (2011)
46. Ummenhofer, B., Zhou, H., Uhrig, J., Mayer, N., Ilg, E., Dosovitskiy, A., Brox, T.: Demon: Depth and motion network for learning monocular stereo. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5038–5047 (2017)
47. Wang, F., Jiang, M., Qian, C., Yang, S., Li, C., Zhang, H., Wang, X., Tang, X.: Residual attention network for image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3156–3164 (2017)
48. Wang, P., Shen, X., Lin, Z., Cohen, S., Price, B., Yuille, A.L.: Towards unified depth and semantic prediction from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2800–2809 (2015)
49. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7794–7803 (2018)
50. Woo, S., Park, J., Lee, J.Y., So Kweon, I.: Cbam: Convolutional block attention module. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 3–19 (2018)

51. Xian, K., Shen, C., Cao, Z., Lu, H., Xiao, Y., Li, R., Luo, Z.: Monocular relative depth perception with web stereo data supervision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 311–320 (2018)
52. Xu, D., Ricci, E., Ouyang, W., Wang, X., Sebe, N.: Multi-scale continuous CRFs as sequential deep networks for monocular depth estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5354–5362 (2017)
53. Xu, D., Wang, W., Tang, H., Liu, H., Sebe, N., Ricci, E.: Structured attention guided convolutional neural fields for monocular depth estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3917–3925 (2018)
54. Yin, W., Liu, Y., Shen, C., Yan, Y.: Enforcing geometric constraints of virtual normal for depth prediction. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 5684–5693 (2019)
55. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. arXiv preprint arXiv:1805.08318 (2018)
56. Zhang, Z.: Microsoft kinect sensor and its effect. IEEE multimedia **19**(2), 4–10 (2012)
57. Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., Torralba, A.: Semantic understanding of scenes through the ade20k dataset. International Journal on Computer Vision (2018)
58. Zwald, L., Lambert-Lacroix, S.: The Berhu penalty and the grouped effect. arXiv preprint arXiv:1207.6868 (2012)