Detail Preserving Depth Estimation from a Single Image Using Attention Guided Networks

利用注意力导向网络保持单目深度估计细节

Zhixiang Hao[1] Yu Li[3] Shaodi You[4,5] Feng Lu[1,2,*]

[1]State key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing, China
[2]Beijing Advanced Innovation Center for Big Data-Based Precision Medicine
Beihang University, Beijing China
[3]Advanced Digital Sciences Center, Singapore [4]Data61-CSIRO [5]Australian National University

{haozx, lufeng}@buaa.edu.cn yul@illinois.edu shaodi.you@data61.csiro.au

Abstract

Convolutional Neural Networks have demonstrated superior performance on single image depth estimation in recent years. These works usually use stacked spatial pooling or strided convolution to get high-level information which are common practices in classification task. However, depth estimation is a dense prediction problem and low-resolution feature maps usually generate blurred depth map which is undesirable in application. In order to produce high quality depth map, say clean and accurate, we propose a network consists of a Dense Feature Extractor (DFE) and a Depth Map Generator (DMG). The DFE combines ResNet and dilated convolutions. It extracts multi-scale information from input image while keeping the feature maps dense. As for DMG, we use attention mechanism to fuse multi-scale features produced in DFE. Our Network is trained end-to-end and does not need any post-processing. Hence, it runs fast and can predict depth map in about 15 fps. Experiment results show that our method is competitive with the state-of-the-art in quantitative evaluation, but can preserve better structural details of the scene depth.

密集特征提取器（ResNet+带孔卷积）和深度图生成器（用注意力机制融合多尺度特征）

1. Introduction

Single image depth estimation is an important task as it enables a variety of computer vision and graphics applications, such as 3D scene reconstructions [42], depth-aware image re-rendering [4], and image refocus [2] *etc*. It is given only a single RGB image as the input, however, aiming to estimate a depth map as output. In particular, indoor scenes have complex textures and structural variations which make it more difficult.

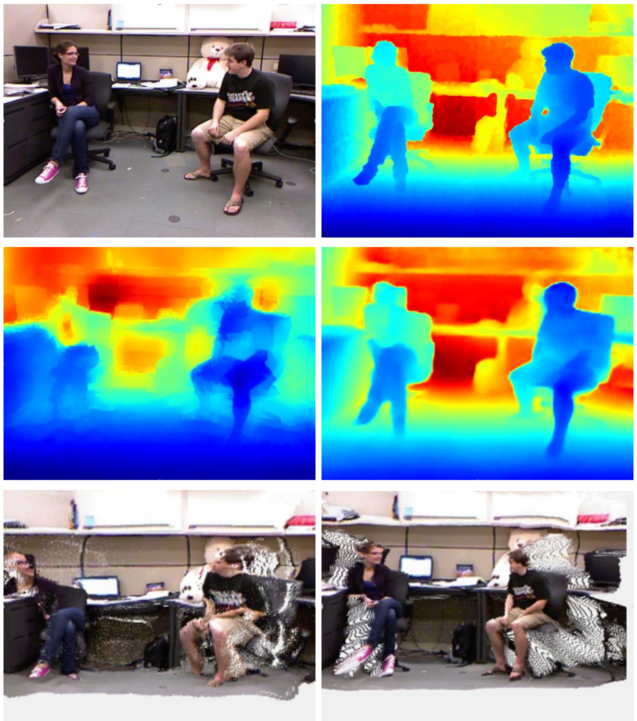


Figure 1. (**Top left**) input RGB image. (**Top right**) ground truth depth. (**Middle left**) depth estimated by Xu *et al.* [44] (relative error: 0.092). (**Middle right**) depth estimated by our method (relative error: 0.085). (**Bottom left**) point cloud from Xu *et al.* depth. (**Bottom right**) point cloud from ours depth. While the accuracy is comparable, our detail-preserving result performs better in depth related applications. 可以看出3D点云的效果比Xu要好

Traditional methods use hand-crafted priors such as Karsch *et al.* [19] who use a depth transfer based on

[*] Corresponding Author: Feng Lu

SIFT Flow [28] and Liu *et al.* [27] who combined predicted semantic information with depth features. Recently, deep learning based methods have shown superior performance by directly learning from a large amount of data [9, 10, 29, 24, 44].

A high-quality depth map is expected to be both accurate and detailed. Existing deep neural network based methods have mostly focused on accuracy, but do not pay much attention to the important details [11, 21]. An example is shown in Fig. 1, while both Xu *et al.* and our method have comparable accuracy; only our method preserves the structural detail and produces reasonable results in 3D reconstruction. Other existing methods suffer from the same problem [24, 10, 5]. The loss of the detail comes from the fact that CNN use an intensive number of strided convolutions and spatial poolings; it reduces the resolution and loses the accurate location information. Leading to a smoothed and low-resolution output. Long *et al.* [32] and Ronneberger *et al.* [35] have proposed methods to improve the resolution of feature maps. However, the detailed edges still fail to align with the image.

To preserve structural details and produce a sharp output, we propose two network modules. The first is Dense Feature Extractor (DFE) which combines ResNet [15] with dilated convolution. DFE uses dilated convolution to extract multi-scale features from an image while keeping the feature maps dense. We also introduce a Depth Map Generator (DMG) module which uses attention mechanism to regress the feature maps to depth in which we can allocate compute resources toward the most informative parts of an input signal according to the context. By combining the proposed DFE and DMG, our network can extract multi-scaled dense while informative features and fuse them effectively. Quantitative experiments on NYU Depth V2 dataset [34] show our method is competitive with the state-of-the-art. Moreover, projected to point cloud and bokeh generation show that our method can preserve better structural details of the scene depth.

We summarize our contributions as follows:

- We propose a novel approach for predicting depth map from a single image which integrates a Dense Feature Extractor and attention mechanism.
- We propose a Fully Convolutional Network which can predict accurate depth map with errors competitive to the state-of-the-art on benchmark dataset, moreover depth map produced by proposed method preserves significantly more structural details that benefit various applications.

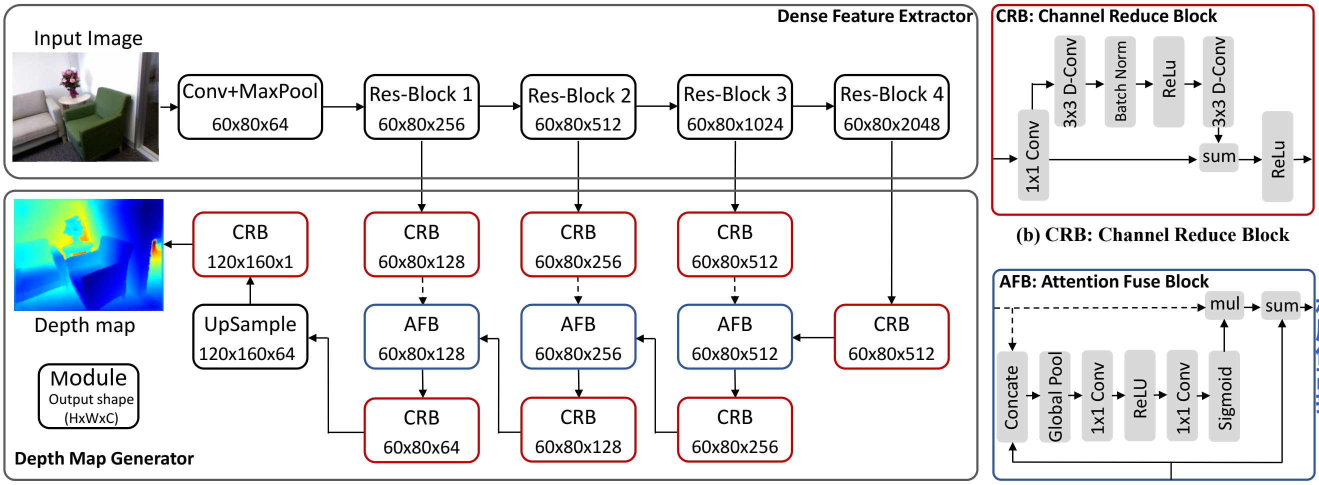
2. Related Work

Single image depth estimation is an important problem in computer vision and an active field of research. We focus on reviewing the monocular methods.

Classic methods for depth estimation from a single image mainly relied on hand-crafted features or graphical models. Saxena *et al.* [37] use a multi-scale Markov Random Field (MRF) and linear regression to predict depth information from features extracted from a single image. Later, they extend their work into Make3D [38]. Inspired by this work, Liu *et al.* [27] show that depth estimation can benefit from combining semantic information with depth features, where predicted labels are used as additional constraints to help the optimization task. Ladicky *et al.* [23] predict semantic labels and depth maps jointly in a classification approach. They train a classifier considering both the predicted semantic labels loss and depth loss. Karsch *et al.* [19] propose a depth transfer, a non-parametric method based on SIFT Flow [28], to reconstruct depth by transferring depth of image in the same dataset. Liu *et al.* [31] treat depth estimation as an optimization problem which is formulated as a Conditional Random Field (CRF) with continuous and discrete variable potentials.

Recently, most of the state-of-the-art for many problems in computer vision is set by deep neural networks, especially CNNs [15, 14, 41]. Not surprisingly, deep learning is highly effective for depth estimation. Eigen *et al.* [10] is the first to use CNNs for estimating depth from a single image. They propose a two-scale network in where one predicts coarse depth from the entire image, and the other refine the coarse prediction locally. This work is extended to predict normal and semantic labels using a three-scale network based on VGG [9, 39]. Liu *et al.* [30] propose combining CNNs and CRF to learn the unary and pairwise potentials with CNNs. They predict depth at a superpixel level and train the CNNs with CRF loss, while Li *et al.* [25] and Wang [43] use hierarchical CRFs to refine patch-wise CNNs prediction from superpixel down to pixel level. Roy [36] proposes the neural regression forest (NRF), which combines random forests and CNNs. Laina *et al.* [24] propose leveraging the power of pre-trained CNNs such as ResNet [15] to predict dense depth maps. Combined with fast up-projection, they archive state-of-the-art results on single image depth estimation.

More recently, Li *et al.* [26] use two-streamed CNNs that predicts both depth and depth gradients, and then fuse them together to produce an accurate depth map. Xu *et al.* [44] fuse complementary information derived from multiple CNNs side outputs by using continuous CRFs. Later, they combine this work with attention model [45]. Fu *et al.* [11] introduce a spacing-increasing discretization (SID) strategy to discretize depth and translate depth estimation from a regression to a classification task. There are also some methods focus on unsupervised or semi-supervised learning by recovering a right view with a left view [12, 22].



(a) Network Architecture

(b) CRB: Channel Reduce Block

(c) AFB: Attention Fuse Block

实际上就是ResBlock，带孔卷积就是为了快速获得高层特征。

相比于旷视的高层特征监督选择低层特征 此处用高层+低层信息选择低层信息

Figure 2. **Network Architecture Overview.** (a) Network architecture. (b) Inner structure of Channel Reduce Block (CRB). (c) Inner structure of Attention Fuse Block (AFB). The dotted line and solid line distinguish the two different inputs in AFB, the D-Conv in CRB represents Dilated Convolution. The input image in the diagram has resolution of 240x320.

120x160算是最小的输入，192x256中等，而240x320正合适

3. Method

In this section, we first detailedly describe the proposed Dense Feature Extractor (DFE) and Depth Map Generator (DMG). Then, by combining the DFE and DMG, we introduce the complete architecture of our proposed network.

3.1. Dense Feature Extractor

Many previous depth estimation methods directly use networks designed for image classification tasks [24, 9, 44], in which, networks output low-resolution feature maps. Hence, these methods suffer from low-resolution or over-smoothed output. To recover the high-resolution spatial detail from the low-resolution feature maps, previous methods [24, 10, 45] usually use the repeated combination of deconvolutional layers (or transposed convolution) which is time-consuming and complicate network training.

反卷积比较复杂和浪费时间

Instead, we leverage the using of 'dilated convolution' (or atrous convolution) to extract higher contextual information while keeping the spatial resolution of feature maps unchanged. Such technology is inspired by recent advance in semantic segmentation [47, 6]. Dilated convolution introduces a parameter called dilation rate. By adjusting dilation rate, it is convenient to control the resolution of feature maps and adjust the convolution kernel's field-of-view in order to extract multi-scale information. For instance, a higher dilation rate such 4 or 8 in our DFE performs convolution with a bigger filed-of-view which extracts higher-level information. In the case of two-dimensional signals such as image, for each location i on the output y and a filter w, dilated convolution is applied over the feature maps x can be formalized as follows [7]:

用带孔卷积提取高层特征并保持空间分辨率

带孔卷积公式：
$$y[i] = \sum_{k} x[i + r \cdot k] w[k], \quad (1)$$

where r is dilation rate. When $r = 1$ dilated convolution is equal to standard convolution.

In our proposed network, we combine dilated convolution and ResNet-101 [15] to form DFE. As shown in Fig. 2, different from the original ResNet, we remove the global average pooling and fully connected layer at the end of ResNet and replace all standard convolution layers with dilated convolution layers in Res-Block 1 to Res-Block 4. Like the original ResNet, our DFE can be divided into five stages according to the size of filled-of-view. By introducing different dilation rate in different Res-Block, our DFE can extract multi-scale information as side outputs in each Res-Block. In the lower stage, *i.e.* small dilation rate, the feature maps contain more low-level spatial information such as location and edge, however, it has poor semantic information due to its small field-of-view. While in the higher stage, *i.e.* big dilation rate, it has bigger field-of-view, hence, more semantic information. Overall, our DFE produces multi-scale feature maps which can represent different information extracted from an input image.

移除全局平均池化层和全连接层正常卷积全部替换为带孔卷积

5个带孔卷积

在低层用小膨胀率 高层用大膨胀率

3.2. Depth Map Generator

To combine the multi-scale information produced by DFE, we introduce the DMG. DMG consists of Attention Fuse Block (AFB) and Channel Reduce Block (CRB). Fig. 2 illustrate the structure of our DMG.

Attention Fuse Block. Attention mechanism allows us to allocate available compute resources towards the most

注意力机制允许我们将有限的计算资源分配给最重要的部分

important parts of an input signal according to the context [33, 17, 16, 46]. Our AFB is designed to change the weights of different channels in feature maps according to the global context and fuse the adjacent stages information similar to [16, 46]. To achieve this goal, we need the weights which will be used to reweight different channels could respond to different scene context. The previous Encoder-Decoder networks, such as SegNet [3] and U-Net [35], usually ignore the global context and just summed up or concatenate the feature maps of adjacent stages. The convolution operates applied to a feature maps which just summed over all channels can be formed as follows:

$$\boldsymbol{y} = \boldsymbol{W} * \boldsymbol{X} = \sum_{s=1}^{C} \boldsymbol{w}_s * \boldsymbol{x}_s, \tag{2}$$

将所有通道信息加起来

where $*$ denotes convolution. $\boldsymbol{y}$ is the output of convolution. $\boldsymbol{X} = [\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_c}]$ represents input feature maps and $\boldsymbol{w_s} \in \boldsymbol{W}$ is a 2D spatial kernel which acts on the corresponding channel of $\boldsymbol{X}$. Note that when the training ends, the weights in convolution kernel $\boldsymbol{W}$ are fixed, therefore, they cannot be changed with input in order to give informative features more weight.

As shown in Fig. 2 (c), the feature maps produced by current Res-Block (represented by dotted line) and previous AFB (represented by solid line) is fused in current AFB. First, the two inputs are concatenated, then feed into a global average pooling layer followed by two 1x1 convolution layers. It produces a 1x1xC tensor in which C is the number of channels in input represented by dotted line. The value in this 1x1xC tensor represents the weight in the corresponding channel of feature maps and it varies according to the context in a different scene. Next, the feature map produced by current Res-Block is weighed by element-wise multiplication with the 1x1xC tensor. Finally, the weighted feature maps is added to the feature maps produced by the previous AFB.

SE作用是原来是固定的1x1卷积 现在换成自适应的1x1卷积同理 guide滤波也可以实现

Channel Reduce Block. As shown in Fig. 2 (b), the DFE produces feature maps with huge channels, such as 2048 in Res-Block 4. In order to reduce the number of channels and refine the feature maps, we introduce the Channel Reduce Block. The first component of the block is a 1x1 convolution layer in which the input's channel reduced. Then, in order to refine the reduced feature maps, the following is a basic residual block inspired by ResNet [15]. In CRB, we use dilated convolution with the same dilation rate as corresponding Res-Block instead of the standard 3x3 convolution.

1x1卷积减少通道 res block优化map

3.3. Network Architecture

By combining DFE and DMG, we propose our complete network for predicting depth from a single image as illustrated in Fig. 2 (a).

We use ResNet [15] pre-trained on ImageNet [8] as a backbone of our DFE. As shown in the diagram, there are four different Res-Block in our DFE according to the dilation rate $r \in \{1, 2, 4, 8\}$ respectively. The final feature maps in first three Res-Block is feed into a CRB, then, an AFB followed by another CRB. Due to lacking followed Res-Block, the last Res-Block is feed into only a CRB. The feature maps in all Res-Block have the same spatial resolution ($1/4$ of the input image) because we use dilated convolution instead of standard strided convolution. Therefore, we apply only one 2x bilinearly upsample to the last CRB output in order to get a denser depth map.

In training, we use $\mathcal{L}_1$ Loss in log space defined between ground truth depth map D and predicted depth map $\hat{D}$ as:

$$L(D, \hat{D}) = \frac{1}{n} \sum_{p}^{n} |\log_e(D_p + 1) - \log_e(\hat{D}_p + 1)|, \tag{3}$$

log距离

where n is the number of pixels in depth map and p is a pixel index. Converting depth to log space can down-weight contribution of regions with large depth value. It benefits for training because the regions with larger depth value have less rich information for depth estimation.

3.4. Implementation Details

We use TensorFlow [1] framework to implement the proposed model and train on a single NVIDIA TITAN Xp GPU with 12GB memory. We initialize the DFE of our proposed network with ResNet-101 weights which pre-trained for ILSVRC image classification task and the weights in DMG are randomly initialized with Xavier initializer [13]. After initialization, we fixed all parameters in the first two stages in ResNet-101 because the first few layers extract general low-level information, and it could be beneficial to prevent overfitting. Besides, the batch normalization parameters are fixed during training time. The batch size is set to 3. Training is performed for 20 epochs, which consists of $50K$ steps per epoch. Similar to [7], we use a "poly" learning rate decay policy with Adam [20] optimizer in which the learning rate l is defined as:

$$l = (l_{init} - l_{end}) * (1 - \frac{global\_step}{decay\_steps})^{power} + l_{end}, \tag{4}$$

学习率更新公式 指数衰减

where l_{init} is initial learning rate and l_{end} is end learning rate, $global\_step = \min(global\_step, decay\_steps)$. Specifically, we set $l_{init} = 10^{-5}$ and $l_{end} = 10^{-7}$ in DFE, $l_{init} = 10^{-4}$ and $l_{end} = 10^{-6}$ in DMG. Since the parameters in DFE are pre-trained on ImageNet, they need more fine scale change. The $decay\_steps$ is set to 16 epochs and $power$ is set to 1.0 in both DFE and DMG. Our training time is about 50 hours in our sampled training set.

Table 1. Depth prediction comparison with other methods on NYU Depth V2 dataset.

Method				Base network	Error (lower is better)			Accuracy (higher is better)		
					rel	rms	log10	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Karsch *et al.* [19]				-	0.374	1.120	0.131	-	-	-
Ladicky *et al.* [23]				-	-	-	-	0.542	0.829	0.941
Liu *et al.* [31]				-	0.355	1.060	0.127	-	-	-
Li *et al.* [25]				-	0.232	0.821	0.094	0.624	0.886	0.968
Liu *et al.* [29]				-	0.230	0.824	0.095	0.614	0.883	0.971
Roy [36]				-	0.187	0.744	0.078	-	-	-
Eigen [9]				VGG-16	0.158	0.641	-	0.769	0.950	0.988
Chakrabarti *et al.* [5]				VGG-19	0.149	0.620	-	0.806	0.958	0.987
Laina *et al.* [24]				ResNet-50	**0.127**	**0.573**	0.055	0.811	0.953	0.988
Li *et al.* [26]				VGG-16	0.152	0.611	0.064	0.789	0.955	0.988
MS-CRF [44]				ResNet-50	**0.121**	0.586	**0.052**	0.811	0.954	0.987
Xu *et al.* [45]				ResNet-50	**0.125**	0.593	0.057	0.806	0.952	0.986
	DFE	AFB	Dilated CRB							
Ours				ResNet-50	0.153	0.663	0.070	0.754	0.953	**0.990**
	✓			ResNet-101	0.134	0.583	0.056	**0.826**	**0.962**	**0.989**
	✓	✓		ResNet-101	0.129	**0.568**	**0.054**	**0.836**	**0.965**	**0.991**
	✓	✓	✓	ResNet-101	**0.127**	**0.555**	**0.053**	**0.841**	**0.966**	**0.991**

4. Experiments

4.1. Dataset and Evaluation Metrics

Dataset. We use one of the largest indoor scene dataset, NYU Depth V2, for training and evaluation. The NYU Depth V2 dataset contains two sub-datasets: labeled dataset and raw dataset. The raw dataset consists of 249 training scenes and 215 test scenes captured with Microsoft Kinect. The labeled dataset has 1449 aligned RGB-D pairs which be officially split into two parts: 795 pairs for training and 654 for testing. For training our depth estimation network, we extract $\sim 13K$ pairs from 249 training scenes in raw dataset. The images in the dataset which have the resolution in 480×640 are down-sampled to 312×416 and then cropped to 240×320 in order to remove blank boundaries. We also perform randomly online data augmentation for the training pairs as follows.

1. The images RGB value are added with a random $\Delta \in [-0.2, 0.2]$ to adjust the image brightness.
2. The images contrast are adjusted with a factor $\delta \in [0.8, 1.2]$.
3. Both RGB images and depth maps are horizontally flipped with 0.5 probability.

Similar to previous works [9, 25, 31, 44], we evaluate our predicted depth map using mean relative error (rel), root mean squared error (rms), mean $\log_{10}$ error and accuracy with threshold as evaluation metrics.

4.2. Performance on NYU Depth V2

Comparison with the State-of-the-Art. In Table 1 we compare the accuracy of depth predicted by the proposed network with the state-of-the-art. The numerical results are taken from the corresponding original paper. The best, second and third result on each metric is colored in red, green and blue respectively. Our method outperforms all other methods in accuracy metrics. In error metrics, we obtain the best rms and competitive rel and log10 results. We would like to point out that MS-CRF [44] which achieve the best rel and log10 error is trained on a $95K$ subset of the NYU Depth V2 dataset. While our train set, consists of $13K$ samples, is radically smaller than MS-CRF.

Ablation Study. We use a simple encoder-decoder architecture consists of an original ResNet-50 removed the global pooling and fully connected layer in the end as encoder and stacked deconvolution layers as decoder as the baseline. In the baseline network, we also introduce skip connection [35] between the same spatial resolution feature maps in encoder and in decoder. The performance of the baseline network is reported in the first row of our methods in Table 1. Comparing the baseline with the proposed network, it clear that leveraging DFE and DMG improves the performance by a large margin. To evaluate the effect of our proposed modules, we add DFN, AFB and dilated CFB to the baseline network gradually. The results show that each of our proposed modular can improve the performance of the corresponding network.

Qualitative Comparison. In Fig. 3 we show some selected visual results of our best model and compare with the publicly available predictions of Eigen [9], Laina *et al.* [24] and MS-CRF [44]. One can clearly see that depth maps produced by our method have noteworthy visual quality especially edge quality and rich details, for example, in the (c) row, the bike in our depth map is as clear as in ground

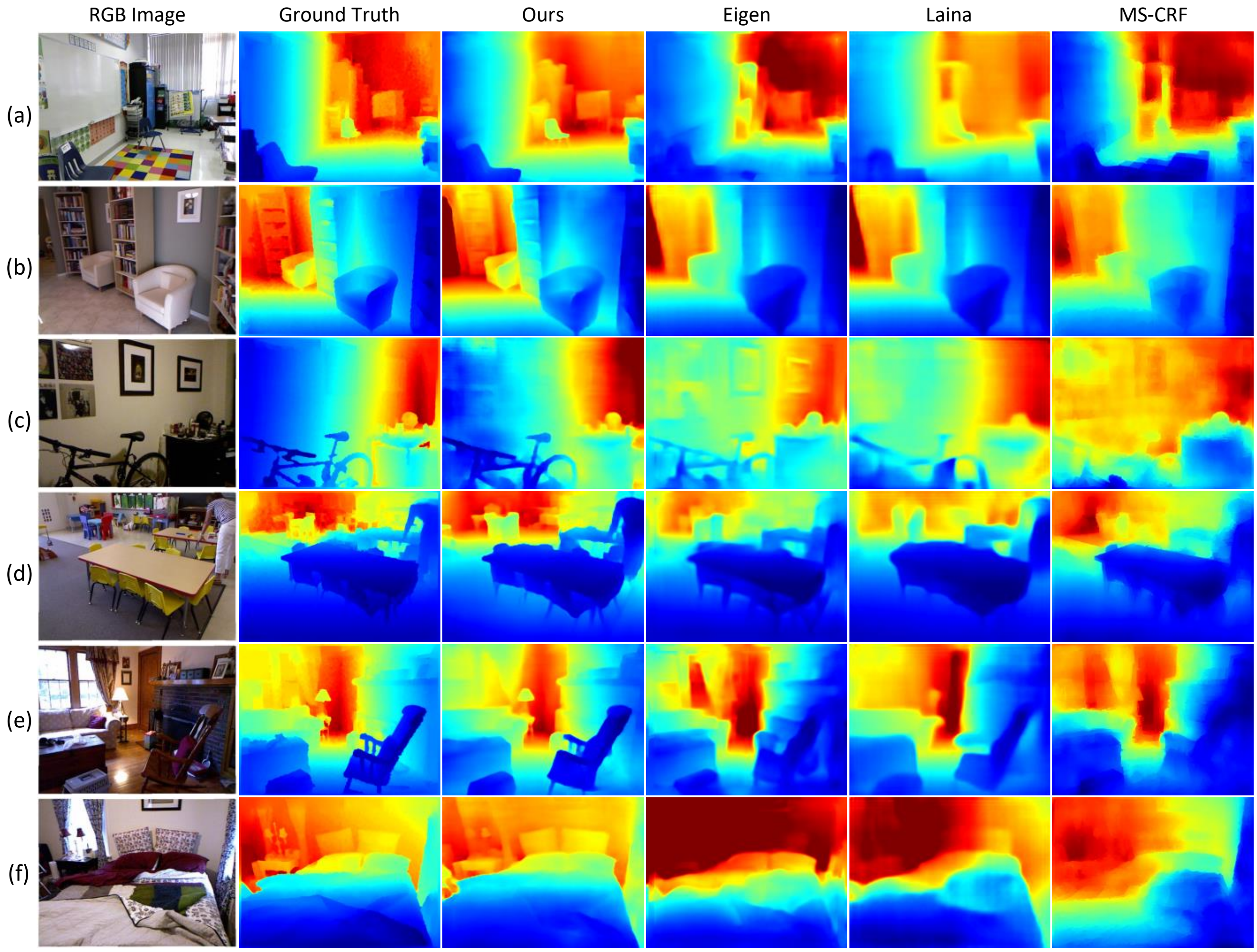


Figure 3. **Qualitative Comparisons on NYU Depth.** Our predictions are clearer and sharper than Eigen [9], Laina [24] and MS-CRF [44].

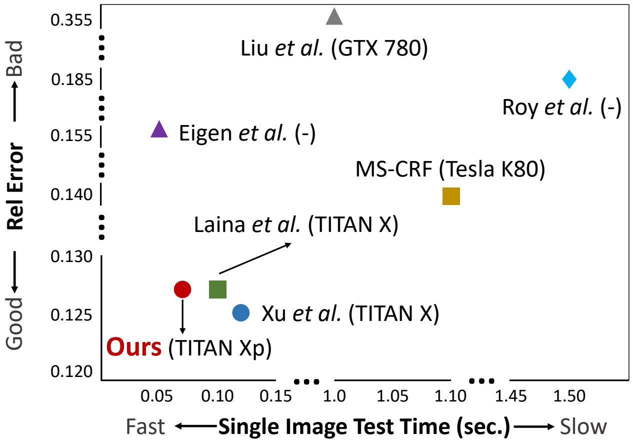


Figure 4. **Accuracy and Speed Comparison.** Comparison with previous method: rel error versus a single image prediction time. Numbers in char are taken from [45].

truth. Moreover, in the (d) row, the person in our depth map is sharp, while, cannot be able to tell in other methods. In the (a) row, our depth map is the only one which the chair in far is able to tell from the background. It is also worth noting that our method produces depth map end-to-end, thus, do not need any additional post-processing such as CRF.

Efficiency. Robotics is one of the major application areas that indoor scene depth estimation can be used for. In which prediction speed is critical. Since we use DFE, there is only one 2x upsampling in our DMG which allow our framework to work at $\sim 15 fps$ for inferencing. We compare the proposed method with previous methods in terms of computational cost in test phase in Fig. 4. Our approach outperforms Roy *et al.* [36], Liu *et al.* [31] and MS-CRF (small training set) [44] both in terms of accuracy and of running time. Furthermore, in comparison to Eigen *et al.* [9], Laina *et al.* [24] and Xu *et al.* [45], the proposed framework obtains an impressive trade-off between accuracy and speed.

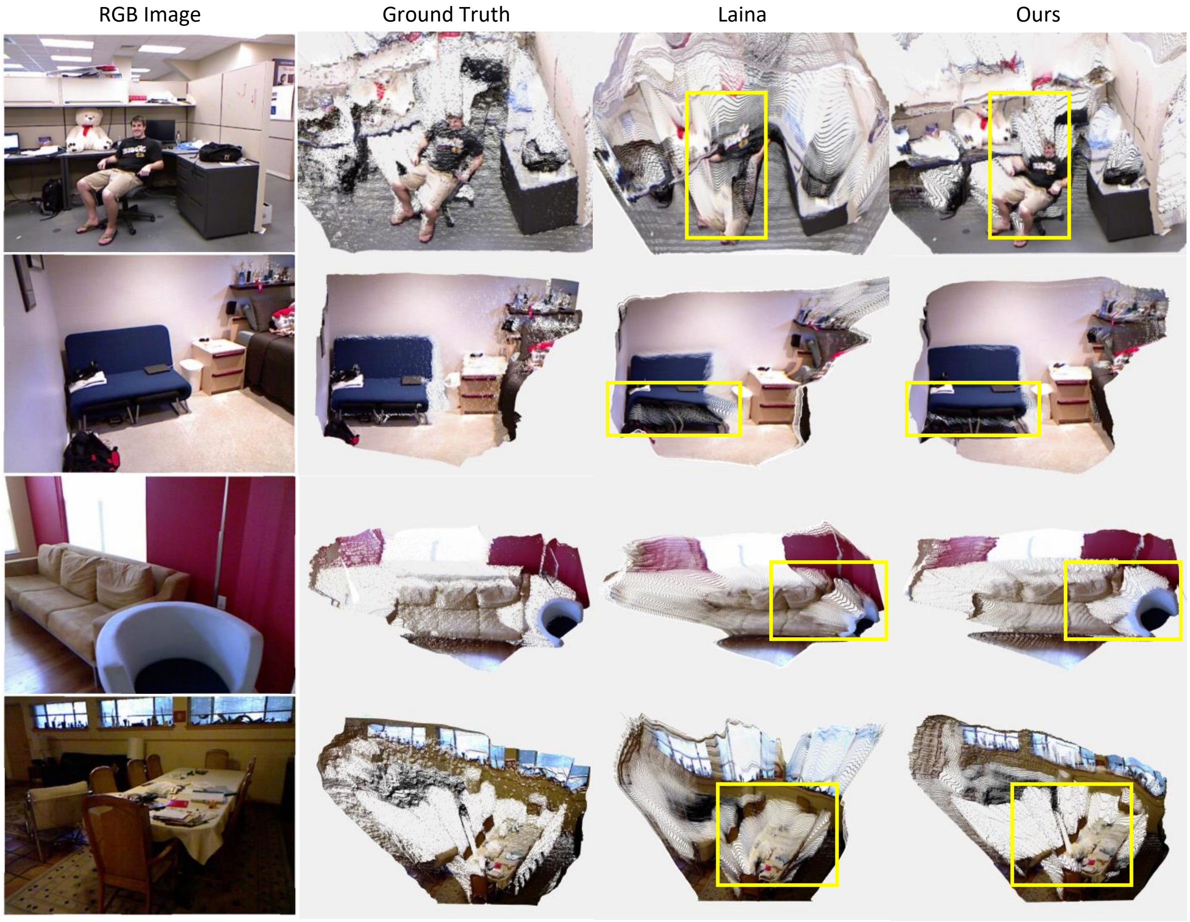


Figure 5. **Projecting Depth Map to 3D Point Cloud Comparisons.** Our results recover better 3D information.

4.3. Applications: point cloud and bokeh

To further illustrate the differences between our method and the previous methods, and emphasize the importance of sharp edge, we compare our result with Laina *et al.* [24] in (1) 3D point cloud, and (2) application of bokeh.

Point Cloud. We make a qualitative comparison in $3D$ by projecting the predicted $2D$ depth map into $3D$ point cloud. The coordinate of the point is computed by a script provided by NYU Depth V2 toolbox. To visualize the quality of produced point cloud, we render the point cloud in a different view from the original RGB image. Some selected results are shown in Fig. 5. It is easy to see that our depth map produces a better $3D$ representation of the scene. For example, in the first row, our method predicts the person in a relative right place while Laina *et al.* [24] predict the head far away from the body. It also worth noting that Laina reports a similar evaluate score to ours, one can see that a numerical metrics such as rel not always reveal the predicted depth quality.

Bokeh. Bokeh effect is usually obtained with DSLR camera and requires special photography skill. It is a

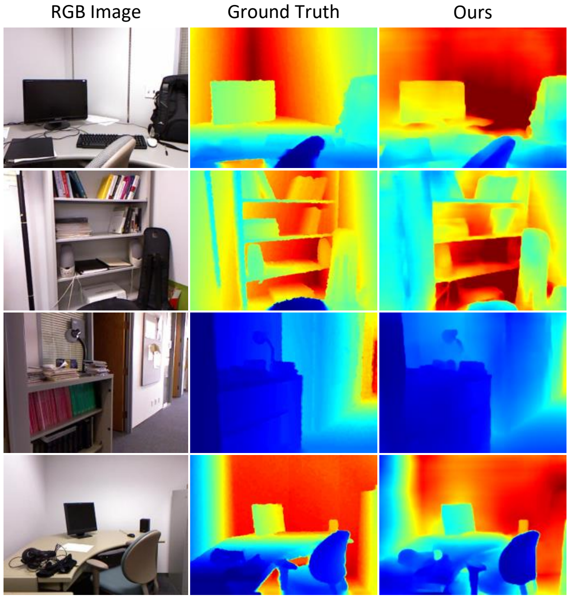


Figure 6. Representative results on SUN-RGBD dataset.

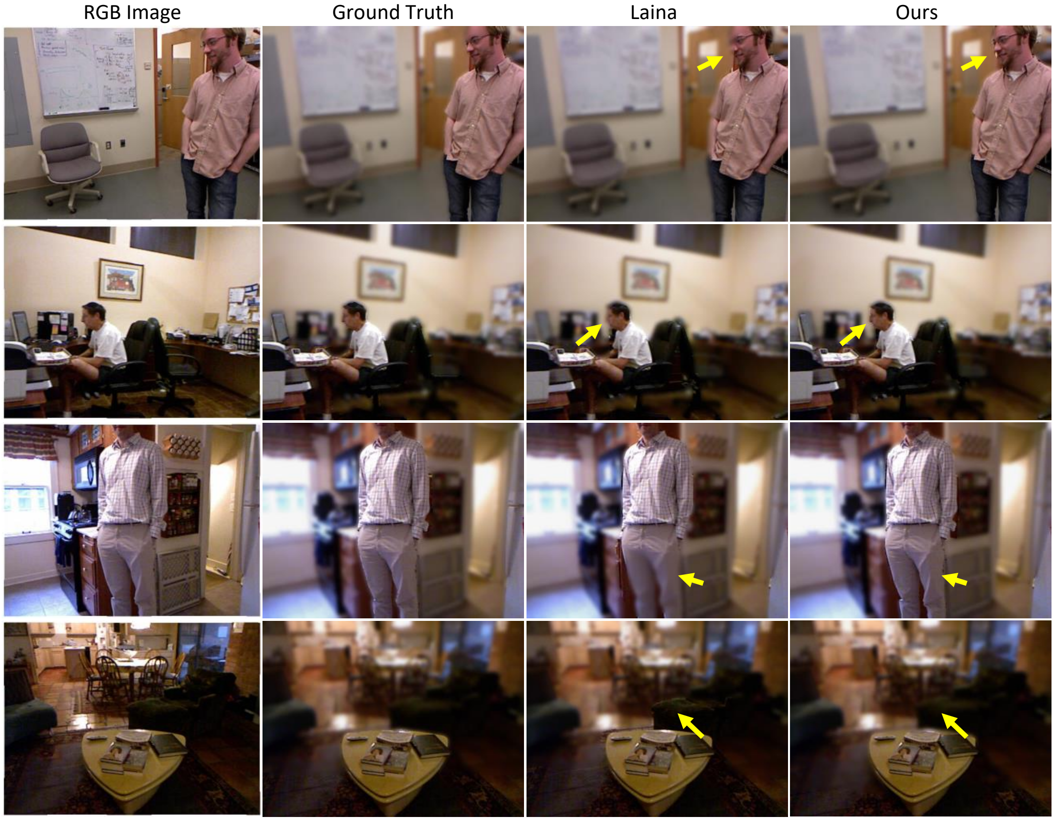


Figure 7. **Bokeh Generation Comparisons.** Depth maps estimated by our method produce better blur effect.

highly desired feature in consumer level like photography using a smart phone. This is now feasible with the scene depth information. Some high-end phones equip with additional sensors for capturing the depth (e.g. iPhone's portrait mode). We show that based on our depth prediction method we can render realistic bokeh effect just with a single color image. With the estimated depth map, we can choose to either keep sharp contents ('in-focus region') or blur the scenes ('out-of-focus region') according to their depth planes. Fig. 7 shows four examples. It is obvious that the results using our depth prediction method is closer to the ones using ground truth depth which keep the edge of object sharper. Specifically, in first to third rows, our depth map makes the person stand out from the background while Laina *et al*. [24] fuses partial body with the background (legs in third rows, face in the first and second row).

4.4. Generalization

To show the generalization of the proposed network, we test it on B3DO [18] which is contained in SUN-RGBD dataset [40]. The image and depth in B3DO are captured by a Microsoft Kinect which has similar camera parameters to the camera used in NYU Depth V2. Note that our depth estimate network is trained on data only contains samples in NYU Depth V2 and never see images in B3DO. Fig. 6 shows some result samples that our network predicts in B3DO. One can see that our network obtains a quite good result especially sharp edge structures.

5. Conclusions

In this paper, we have proposed a novel approach for single image depth estimation. Our method is motivated by obtaining a detailed depth map. Unlike previous works that typically reduce spatial resolution of the feature maps, our method introduces DFE to extract features from the image while keeping the feature maps dense. Moreover, attention mechanism is integrated into DMG to fuse the multi-scale information produced in DFE. We evaluate the proposed method both numerically and qualitatively on the benchmark dataset. By performing applications such as bokeh generation and $3D$ point cloud, it indicates that our result has more fine-scaled details and align with depth boundary better. In the future, we want to extend our method to estimate depth in monocular video with temporal consistency.

Acknowledgements

This work was supported by NSFC under Grant 61602020 and Grant 61732016. The GPU is granted by DRIVE PX Grant Program.

References

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016. 4

[2] S. Anwar, Z. Hayder, and F. Porikli. Depth estimation and blur removal from a single out-of-focus image. In *BMVC*, 2017. 1

[3] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *TPAMI*, 39(12):2481–2495, 2017. 4

[4] T. D. Basha, Y. Moses, and S. Avidan. Stereo seam carving a geometrically consistent approach. *TPAMI*, 35(10):2513–2525, 2013. 1

[5] A. Chakrabarti, J. Shao, and G. Shakhnarovich. Depth from a single image by harmonizing overcomplete local network predictions. In *NIPS*, 2016. 2, 5

[6] 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. *TPAMI*, 40(4):834–848, 2018. 3

[7] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. *arXiv preprint arXiv:1802.02611*, 2018. 3, 4

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 4

[9] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015. 2, 3, 5, 6

[10] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NIPS*, 2014. 2, 3

[11] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, 2018. 2

[12] R. Garg, V. K. BG, G. Carneiro, and I. Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *ECCV*, 2016. 2

[13] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, 2010. 4

[14] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *ICCV*, 2017. 2

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2, 3, 4

[16] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 4

[17] L. Itti and C. Koch. Computational modelling of visual attention. *Nature reviews neuroscience*, 2(3):194, 2001. 4

[18] A. Janoch, S. Karayev, Y. Jia, J. T. Barron, M. Fritz, K. Saenko, and T. Darrell. A category-level 3d object dataset: Putting the kinect to work. In *Consumer Depth Cameras for Computer Vision*, pages 141–165. Springer, 2013. 8

[19] K. Karsch, C. Liu, and S. B. Kang. Depth extraction from video using non-parametric sampling. In *ECCV*, 2012. 1, 2, 5

[20] D. Kinga and J. B. Adam. Adam: A method for stochastic optimization. In *ICLR*, 2015. 4

[21] J. N. Kundu, P. K. Uppala, A. Pahuja, and R. V. Babu. Adadepth: Unsupervised content congruent adaptation for depth estimation. In *CVPR*, 2018. 2

[22] Y. Kuznietsov, J. Stückler, and B. Leibe. Semi-supervised deep learning for monocular depth map prediction. In *CVPR*, 2017. 2

[23] L. Ladicky, J. Shi, and M. Pollefeys. Pulling things out of perspective. In *CVPR*, 2014. 2, 5

[24] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*, 2016. 2, 3, 5, 6, 7, 8

[25] 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 *CVPR*, 2015. 2, 5

[26] J. Li, R. Klein, and A. Yao. A two-streamed network for estimating fine-scaled depth maps from single rgb images. In *ICCV*, 2017. 2, 5

[27] B. Liu, S. Gould, and D. Koller. Single image depth estimation from predicted semantic labels. In *CVPR*, 2010. 2

[28] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *TPAMI*, 5(33):978–994, 2011. 2

[29] F. Liu, C. Shen, and G. Lin. Deep convolutional neural fields for depth estimation from a single image. In *CVPR*, 2015. 2, 5

[30] F. Liu, C. Shen, G. Lin, and I. Reid. Learning depth from single monocular images using deep convolutional neural fields. *TPAMI*, 38(10):2024–2039, 2016. 2

[31] M. Liu, M. Salzmann, and X. He. Discrete-continuous depth estimation from a single image. In *CVPR*, 2014. 2, 5, 6

[32] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 2

[33] V. Mnih, N. Heess, A. Graves, et al. Recurrent models of visual attention. In *NIPS*, 2014. 4

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

[35] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 2, 4, 5

[36] A. Roy and S. Todorovic. Monocular depth estimation using neural regression forest. In *CVPR*, 2016. 2, 5, 6

[37] A. Saxena, S. H. Chung, and A. Y. Ng. Learning depth from single monocular images. In *NIPS*, 2006. 2

[38] A. Saxena, M. Sun, and A. Y. Ng. Make3d: Learning 3d scene structure from a single still image. *TPAMI*, 31(5):824–840, 2009. 2

[39] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2014. 2

[40] S. Song, S. P. Lichtenberg, and J. Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *CVPR*, 2015. 8

[41] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016. 2

[42] K. Tateno, F. Tombari, I. Laina, and N. Navab. Cnn-slam: Real-time dense monocular slam with learned depth prediction. In *CVPR*, 2017. 1

[43] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. L. Yuille. Towards unified depth and semantic prediction from a single image. In *CVPR*, 2015. 2

[44] D. Xu, E. Ricci, W. Ouyang, X. Wang, and N. Sebe. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In *CVPR*, 2017. 1, 2, 3, 5, 6

[45] D. Xu, W. Wang, H. Tang, H. Liu, N. Sebe, and E. Ricci. Structured attention guided convolutional neural fields for monocular depth estimation. In *CVPR*, 2018. 2, 3, 5, 6

[46] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang. Learning a discriminative feature network for semantic segmentation. In *CVPR*, 2018. 4

[47] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 3