Pattern Recognition 109 (2021) 107578

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

DPNet: Detail-preserving network for high quality monocular depth estimation

Xinchen Ye[a,b,*], Shude Chen[a], Rui Xu[a,b]

[a] DUT-RU International School of Information Science & Engineering Dalian University of Technology, China
[b] Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, China

ARTICLE INFO

Article history:
Received 31 August 2019
Revised 15 June 2020
Accepted 4 August 2020
Available online 25 August 2020

Keywords:
Depth estimation
Detail-preserving
Spatial
Attention
Depth map

ABSTRACT

Existing monocular depth estimation methods are unsatisfactory due to the inaccurate inference of depth details and the loss of spatial information. In this paper, we present a novel detail-preserving network (DPNet), i.e., a dual-branch network architecture that fully addresses the above problems and facilitates the depth map inference. Specifically, in contextual branch (CB), we propose an effective and efficient nonlocal spatial attention module by introducing non-local filtering strategy to explicitly exploit the pixel relationship in spatial domain, which can bring significant promotion on depth details inference. Meanwhile, we design a spatial branch (SB) to preserve the spatial information and generate high-resolution features from input color image. A refinement module (RM) is then proposed to fuse the heterogeneous features from both spatial and contextual branches to obtain a high quality depth map. Experimental results show that the proposed method outperforms SOTA methods on benchmark RGB-D datasets.

1. Introduction

Depth estimation is one of the most widely studied topics in the field of computer vision. An accurate depth map has been demonstrated to provide essential 3D information for many computer vision tasks, including semantic labeling [1,2], robotics navigation [3], 3D reconstruction [4,5] and so on. While high quality texture information can be easily captured by popular color cameras, the acquisition of depth information [6] is still remaining a challenging task in real conditions.

Hence, depth estimation from single camera becomes alternate choice by exploiting monocular cues from a given color image. However, for lacking geometry constraints in color images, estimating depth from a generic scene is an ill-posed problem due to the inherent ambiguity of mapping the color measurement into depth. Recently, convolutional neural networks (CNNs) are widely used for monocular depth estimation [7]. To achieve accurate depth estimation, most methods resort to enhance the ability of feature representations for pixel-level regression, e.g., exploiting multi-scale information [8,9] or long-range dependencies [10]. However, despite that overall depth levels are reliably estimated, the recovery of depth details, e.g., object boundaries and fine structures, is unsatisfactory, as shown in Fig. 1.

Intuitively, a color image and the associated depth map are the photometric and geometrical representation of the same scene, and have strong structural correlation. Pixels with similar appearances have more chances of belonging to the same object, and should have close depth values. Previous graphic model-based works [11,12] have achieved superior performance on depth detail recovery by fully exploiting such correlation. For example, the non-local image model [12] is constructed by pair-wise similarity of non-local pixels computed from the associated color similarities. Recently, there also appear many ideas that leverage the relationship between channels or neighboring pixels in a feature map. For example, some methods learn the affinity matrix to capture the nonlocal similarity [13–15], while others focus on self-attention mechanism to capture feature interdependencies in spatial and channel dimensions [16–18]. As mentioned above, considering the nonlocal correlation between pixels could bring significant promotion on detail recovery and potentially help the depth map inference, which has been mostly ignored by existing CNN-based depth estimation methods.

Meanwhile, the nonlocal correlation is computationally intensive, since the pairwise similarity computation is usually applied under the global view. For example, [12] takes spatial distance and color intensity difference in multiple gaussian kernels to compute the pair-wise similarity within an approximate global scale, while [19] uses matrix multiplication to apply the pairwise computation on the whole image domain. This motivates us to design an efficient way to implement it under the CNN architecture.

* Corresponding author.
E-mail address: yexch@tju.edu.cn (X. Ye).

https://doi.org/10.1016/j.patcog.2020.107578
0031-3203/

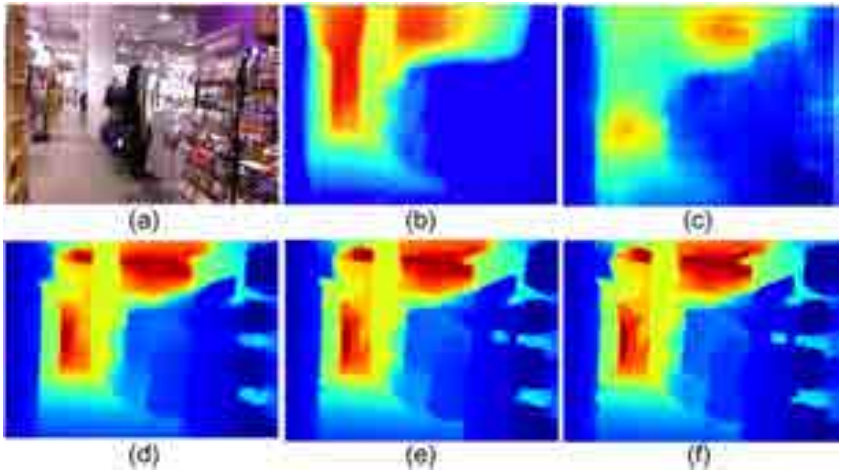

Fig. 1. A depth estimation example. (a) Color image; (b) Lee et al. [20] (c) Ours w/o spatial branch and refinement module; (d) Ours w/o attention module; (e) Ours; (f) Ground truth.

Fig. 2. Two examples to show inferred depths and activation maps (output from nonlocal correlation) from the marked color patches extracted within nonlocal neighborhoods.

Besides, almost all current CNNs need repeated combinations of pooling and downsampling that progressively decrease the resolution of feature map, which leads to low resolution and blurry results due to the loss of spatial information. Some methods [10,21] use conditional random field (CRF) as a separate post-processing. Other advanced methods recast CRF model as a loss function or a part of the network, and train them together [22–24]. The rest works focus on either designing an up-projection block [25] or using auxiliary information (e.g., depth gradient) [24,26] to obtain a high-resolution (HR) depth map. Different from these methods, we exploit a dual-branch architecture to separately consider the spatial and contextual information, and then design a fusion strategy to combine theses features.

In this paper, we simultaneously address the above problems, and propose a novel detail-preserving network (DPNet), i.e., a dual-branch network architecture that fully addresses the above problems and facilitates the depth map inference, as shown in Fig. 4. Specifically, in contextual branch (CB), we propose a nonlocal spatial attention module by introducing non-local filtering strategy to explicitly exploit the non-local correlation in spatial domain to facilitate depth details inference. A fast strategy by replacing the global cross correlation with a nonlocal affinity is proposed to reduce the implementation complexity without sacrificing the estimation performance. Combining with a channel attention module, we apply the dual-attention module on top of the backbone in contextual branch to model the high-level nonlocal dependencies. Meanwhile, we also design a spatial branch (SB) to retain abundant spatial details and generate HR features from input color image. A refinement module (RM) is proposed to fuse the heterogeneous features from both spatial and contextual branches to obtain a high quality depth map. Our main contributions can be summarized as follows:

1) A dual-branch depth estimation network architecture that separately captures low-level and high-level feature representations, which fully addresses the problems of spatial information loss and inaccuracy inference on depth details.
2) A refinement module is proposed to fuse different levels of features from both the branches and obtain the final high quality depth map.
3) A nonlocal spatial attention module is proposed to explicitly exploit the nonlocal correlation in spatial domain. The module structure is designed to reduce the computational complexity without sacrificing the overall prediction accuracy, making it a valuable way to be implemented in faster depth estimation.

We achieve new state-of-the-art results on the popular benchmark NYU v2 dataset. Note that, we train our network on a small labeled training subset that containing 795 image pairs only, but obtain lowest values 0.474 and 0.081 under the *Rmse (lin)* and *Rmse (log)* metric, which improves the performance by 17% and 42% than the second best method (0.571, 0.193), respectively.

2. Related work

2.1. Monocular depth estimation

Previous methods mainly focus on the graphical models with hand-crafted geometry priors [27] or the non-parametric depth transfer techniques [28,29]. Recently, CNNs have been extensively applied into the depth estimation task. One simple way is to use

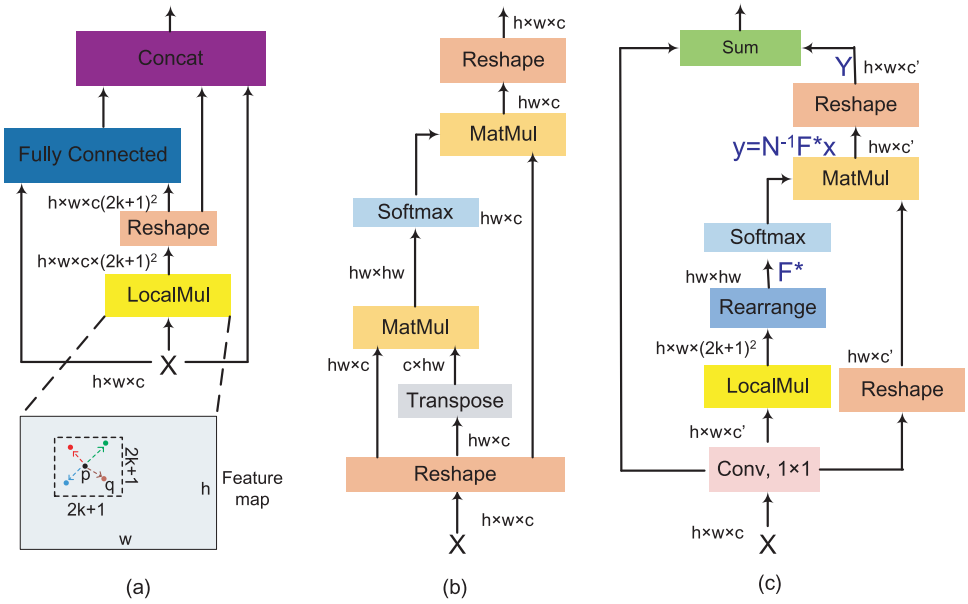

Fig. 3. Comparison of different nonlocal correlation structures to exploit feature similarities in (a) [34], (b) [19], and (c) Our proposed nonlocal spatial attention module. "LocalMul" is defined as the inner product of feature vectors between centering pixel p and its neighbors q in a squared region with the size $2k+1$. The size of the reshaped output feature map from "LocalMul" is $h \times w \times c(2k+1)^2$. "MatMul" is defined as a global cross correlation by multiplying between the reshaped feature map x and its transpose version.

super-pixel based segmentation to divide the whole image into small homogenous regions, and learn the depth for each region using deep networks [21,23].

To achieve precise depth estimation, it is necessary to enhance the ability of feature representations for pixel-level regression. Some methods aim at exploiting multi-scale schemes to capture information at different scales simultaneously, such as using multi-scale inputs [8,30], pyramid dilated convolutions [9], the encoder-decoder structures with long connections (U-net [31]) [22,32], or the recurrent models [33]. To capture long-range context, horizontal or vertical convolutions [10] and pairwise similarity learning [34] have been proposed to explicitly model the relationship between pixels in a given support or direction.

To improve the image quality of the predicted depth map, some methods [10,21] use conditional random field (CRF) as a separate post-processing. Other advanced methods recast CRF model as a loss function or a part of the network, and train them together [22–24]. Lee et al. [20] used the frequency domain analysis to enhance the dataset, and re-designed a depth-balanced loss to achieve better estimation in closer areas. The rest works focus on either designing an up-projection block [25] or using auxiliary information (e.g., depth gradient) [24,26] to obtain a HR depth map.

Recently, some methods use binocular images as supervision to train the monocular depth estimation network. Zhao et al. [35] proposed an unsupervised learning method that simultaneously utilizes labels in the synthetic data and epipolar specific geometric information from the real data for better monocular depth estimation. Chen et al. [36] studied a SceneNet towards scene understanding to perform both semantic segmentation and depth estimation, which is a cross-modal network model integrating both depth and segmentation modalities. Pilzer et al. [37] proposed to predict the synthetic image and disparity opposite to the input image and re-synthesize the input image to build the cycle inconsistency between the original and the reconstructed input image. Then they utilized a refinement network to reduce the inconsistency and refine the final disparity. Wong and Soatto [38] introduced an adaptive regularization to constrain the incorrect penalty of the smoothness term on object boundaries. Puscas et al. [39] proposed a unified deep model consisting of a dual generative adversarial networks (GAN) and a structured conditional random field (CRF). The dual GAN is used to exploit the relationship between stereo images and the CRF provides a structured connection between the discriminator and the generator.

2.2. Nonlocal correlation

Previous filtering [40,41] or global optimization techniques [11,12,42] on the task of depth recovery have focused on depth detail preservation by fully exploiting nonlocal correlation in a nonlocal neighborhood. Recently, there appear many ideas that leverage the relationship between channels or neighboring pixels under the CNN architecture.

The first category is *affinity learning*, in which an affinity is a generic matrix that determines pixel similarity in image or feature space calculated from low-level coherence of appearance or semantic-level similarities in various applications [41,43,44]. Recent techniques resort to learning-based methods [13–15,34] to model the spatial relationship between neighboring pixels instead of hand-crafted design. For example, Liu et al. [14] proposed spatial propagation networks for learning the affinity matrix, which models dense, global pairwise relationships of an image. Ahn and Kwak [13] proposed an AffinityNet that predicts semantic affinity between a pair of adjacent image coordinates. Gan et al. [34] modeled the relationships of different image locations with an affinity layer and combine absolute and relative features in an end-to-end network.

Besides, *self-attention* mechanism originates from the natural language processing field, and has recently been widely used in computer vision to model internal representations by inferring an attention map from a group of feature map. Hu et al. [18] proposed the Squeeze-and-Excitation block, which adaptively recalibrates channel-wise feature responses by explicitly modelling interdependencies. Wang et al. [17] constructed the spatial attention-aware module based on the encoder-decoder structure. Woo et al. [16] proposed a convolutional block attention module (CBAM) to infer attention maps along channel and spatial dimensions sepa-

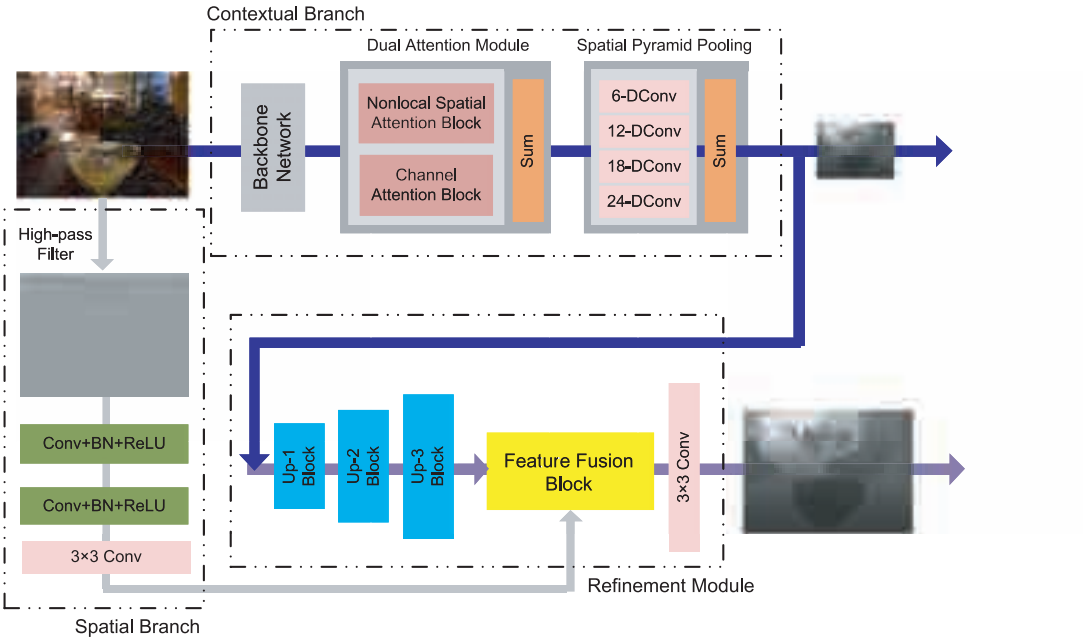

Fig. 4. The proposed deep architecture to estimate depth from monocular RGB image. s-DConv denotes dilated convolution with dilation rate *s*.

rately. Another self-attention mechanism can be categorized as linear algebra based method. Vaswani et al. [45] computed the response at a position in a 1D sequence by attending to all positions and taking their weighted average in an embedding space. Wang et al. [19] extended the self-attention in 1D translation [45] to a more general 2D image processing problems.

3. Motivation

Indeed, existing methods have already exploited global and multi-scale information to some extent by using some enhanced feature representations to estimate depth map reliably. However, depth details, e.g., object boundaries and tiny objects, cannot be predicted by only addressing the overall estimation of depth levels. In contrast, traditional non-local methods, such as [12] and [42], are able to simultaneously achieve global sensing and granular spatial resolution by enlarging effective filtering support and attenuating uncorrelated features, i.e., via similarity search. The motivation is to facilitate the prediction of fine structures by introducing nonlocal filtering strategy to explicitly exploit the pixel relationship in the contextual branch. As shown in Fig. 2, the activation maps obtained from our nonlocal attention module also verifies the idea. Pixels that have similar color appearances with the centering pixel are activated, which are likely to belong to similar depth values. Meanwhile, the loss of rich spatial information also brings a low-resolution and blurry results. This motivates us to further construct a separate spatial branch without any pooling or downsamping to extract HR features, which can remedy the spatial information loss in contextual branch.

Next, we present the motivation of our design methodology. Fig. 3(a) and (b) introduce two relevant nonlocal correlation modules proposed in [34] and [19]. Given the input feature map X of the size $h \times w \times c$, where h, w, c are the height, width, and channels respectively, [34] implicitly modeled the relationship of different image locations with an affinity layer. An operator of Local multiplication (LocalMul) is used to calculate the affinity (or called similarity) matrix in a local region. Wang et al. [19] computes the response at a position by attending to all positions and taking their weighted average in an embedding space, which can be regarded as a global filtering operation on feature maps. The filtering matrix is calculated by the global correlation (MatMul) and Softmax on a feature map. Then the top "MatMul" is used to apply filtering matrix on the reshaped input feature X to obtain the output feature map. Note that both structures use pair-wise multiplication to obtain the pixel similarities. In contrast, Gan et al. [34] restricts the multiplication operation into a fixed nonlocal neighborhood depending on the size K, which can decrease the implementation complexity compared to the "MatMul" operation that applied on the whole image domain. On the contrary, Gan et al. [34] uses learned weights from fully connected layer to fuse the local and nonlocal features, which loses the latent positional correspondence between neighboring pixels, while [19] keeps the merit of computing responses based on relationships between different locations. Inspired by the respective advantages of both modules, we design our nonlocal spatial attention module by introducing non-local filtering strategy in the network to maintain the non-local behavior, but replacing the global cross correlation with a nonlocal affinity to balance the complexity and performance. Experimental section validates the effectiveness of our design from both accuracy and running time, and also demonstrates that a constricted region support for capturing the nonlocal correlation is sufficient to facilitate the depth map inference.

4. Proposed method

Our framework can be divided into three parts, i.e., contextual branch (CB), spatial branch (SB) and refinement module (RM), as shown in Fig. 4. In CB, we use ResNet-101 [46] as our backbone by taking a HR color image as input. The feature maps obtained from the backbone network are fed into the dual attention module, including parallel proposed nonlocal spatial attention (NSA) module and a simple channel attention module. Note that NSA aims to strengthen the nonlocal representations within a feature map by selectively focusing on useful high-level information to guide the depth estimation, while the channel attention module aims to effectively mine the relationship between feature channels and then emphasize useful feature channels.

Then, a spatial pyramid pooling scheme [47] is stacked on our attention module to further capture the multi-scale information. The CB generates an 8 × downsampling LR depth map from the

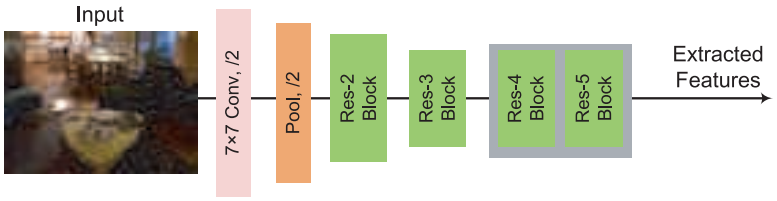

Fig. 5. Backbone network. All the successive skip structures in ResNet-101 model are uniformly marked as a residual block (Res Block).

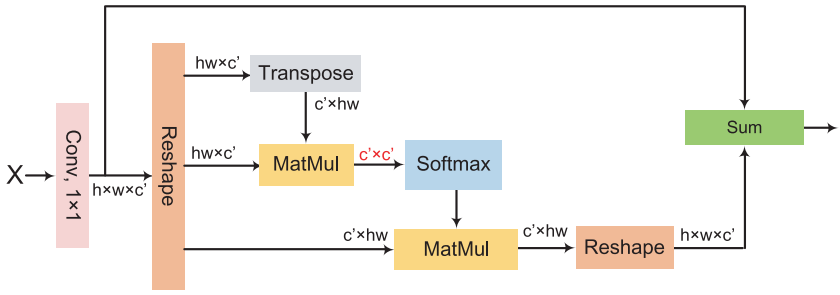

Fig. 6. Channel attention module.

input color image. In SB, we generate a group of feature maps with the same resolution of input color image. In RM, we first design a hierarchical upsampling network based on residue learning structure to progressively upsample the LR depth map from CB to the HR resolution, then a feature fusion block is proposed to fuse both the upsampled features from CB and HR features from SB.

4.1. Contextual branch (CB)

Backbone network. The original ResNet-101 downsamples the feature resolution by five times to aggregate features, resulting a 1/32 output resolution of the input. Therefore, dilated convolutions with the dilation rate set to 2 (2-DConv) [48] are used to replace the downsampling operators in Res-4 Block and Res-5 Block, thus keeping the output resolution at a relative large size, i.e., 1/8 of the input to preserve more spatial details, as shown in Fig. 5.

Dual attention module. As shown in Fig. 3(c), we design a nonlocal spatial attention module to draw nonlocal dependencies over local features generated by the backbone network. We explicitly model between the input and output feature maps X and Y based on the non-local filtering equation:

$$y = N^{-1}F^{*}x, \tag{1}$$

where x, y are the vector versions of the input and output feature maps X and Y, respectively. F^{*} is a filtering matrix that provides an interpretation of the nonlocal structure, where $F^{*}(p,q) = x_p^T x_q$ denotes the similarity between pixels p and q. N is a diagonal matrix in which each element in the diagonal equals to $\Sigma_q F(p,q)$ for a given pixel p. X is first fed into 1×1 convolutions to reduce the number of channels to $c^{'} = 32$. Then we use "LocalMul" operation to compute the pixel similarities in a local squared window with the size of $2k+1$. Note that each pixel can be regarded as a vector with the dimension $c^{'}$, and the "LocalMul" operation for each paired pixels can be regarded as an inner product. Without loss of positional pixel correspondence, we rearrange the feature maps obtained from "LocalMul" according to the relative pixels relationship, to form a sparse filtering matrix F^{*}. Note that the computation complexity can largely reduce from $(h \times w)^2 \times c$ multiplications to the number of $h \times w \times c \times (2k+1)^2$ when using local operation (LocalMul) instead of global correlation (MatMul). The "Softmax" layer equals to the normalization process. Then the top "MatMul" is used to filter on the input reshaped feature x to output the feature map y. Finally, the reshaped feature map Y and the input X are added together to output the result.

Meanwhile, the process to capture the channel relationship is similar to the original self-attention module (Fig. 3(b)) except for the first "MatMul" step, in which channel weighting matrix is calculated in the channel dimension[1]. We do not alter the structure for the channel attention module, since there are only $h \times w \times c'^2$ multiplications due to the compressed channel dimensions (shown in Fig. 6). Note that the channel attention module selectively emphasizes interdependent channel maps by integrating associated features among all channel maps.

Finally we aggregate the output features from the two attention modules to achieve better feature representations. The proposed dual attention module successfully leverages the power of traditional nonlocal filtering strategy and provides rich similarity-measure features for depth estimation, which also alleviates the parameter demand.

Multi-scale module. Four different dilated convolutions (6-, 12-, 18-, and 24-DConv) are applied in parallel as a pyramid structure to extract four feature maps with different receptive fields. Then, we sum the four feature maps to obtain the output depth map. The scheme uses multi-scale fusion to capture objects of different scales, which can enhance the ability of feature representations for pixel-level regression.

4.2. Spatial branch (SB)

Different form contextual branch, the objective of spatial branch is to acquire HR features, which can be regarded as low-level information. A lightweight network with no pooling or downsampling operation is enough to preserve the spatial size of the original input image. Therefore, we first manually extract edge content from color image by a high-pass filter, then send it into a three-layer network. The first two layers include a convolution followed by

[1] In this case, we have $c^{'}$ pixels, and each pixel can be regarded as a vector with the dimension $h \times w$

Table 1
Objective performance comparison between the proposed method and other state-of-the-art methods. The compared methods in top part are trained on a larger training set containing at least 20K image pairs, while the ones in bottom part trained on 795 image pairs from the official splitting.

Method	Error (Lower is better)			Accuracy (Higher is better)		
	RMSE (lin)	Log10	RMSE (log)	δ <1.25	δ <1.25^2	δ <1.25^3
Eigen and Fergus [8]	0.907	-	0.283	61.1%	88.7%	97.1%
Li et al. [26]	0.635	0.063	-	78.8%	95.8%	99.1%
Chakrabarti et al. [49]	0.620	-	0.205	80.6%	95.8%	98.7%
Laina et al. [25]	0.573	**0.055**	0.204	81.1%	95.3%	98.8%
Lee et al. [20]	0.573	-	0.193	81.5%	96.3%	99.1%
Heo et al. [10]	0.571	0.058	-	**81.6%**	**96.4%**	**99.2%**
Ours	**0.474**	0.063	**0.081**	78.4%	94.8%	98.6%
Li et al. [21]	0.821	0.094	0.214	62.1%	88.6%	96.8%
Liu et al. [23]	0.824	0.095	-	61.4%	88.3%	97.1%
Gan et al. [34]	0.631	0.066	-	75.6%	93.4%	98.0%
Ours	**0.474**	**0.063**	**0.081**	**78.4%**	**94.8%**	**98.6%**

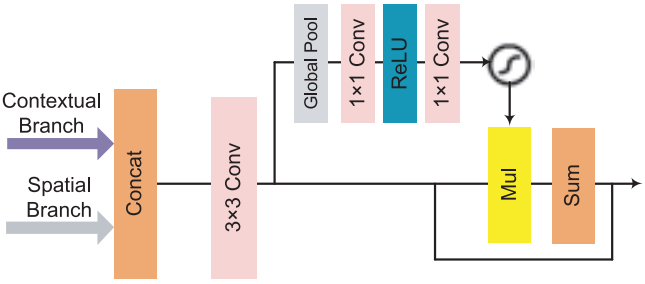

Fig. 7. Feature fusion block.

batch normalization and ReLU, while the last one only contains a 3 × 3 convolution.

4.3. Refinement module (RM)

The LR depth map obtained from CB is fed into three continuous Up-Blocks to progressively enlarge the resolution of the feature maps to the original HR size. All the Up-Blocks have the same structure, sequentially including a convolution layer, a standard skip structure [46], and a transposed convolution layer to upsample the feature map.

Note that the upsampled features from CB and the HR features from SB represent different levels of feature representation. Therefore, we design a feature fusion module to combine these two features. As shown in Fig. 7, the concatenated features from SB and CB are first sent into a 3 × 3 convolution, then pooled to a feature vector to re-weight each channel of the features, like SENet [18]. The whole module can be regarded as a process of feature selection and combination. Finally, a 3 × 3 convolution layer is used to generate the HR depth map.

4.4. Loss function

The overall loss function can be defined as follows:

$$\mathcal{L}(\hat{D}_L, \hat{D}) = \sum_{i=1}^{N} \left(||\hat{D}_L{}^{(i)} - D_L^{(i)}||_2^2 + \lambda ||\hat{D}^{(i)} - D^{(i)}||_2^2 \right), \quad (2)$$

where D, $\hat{D}$, D_L, and $\hat{D}_L$ represent the groundtruth and the predicted HR depth maps and their corresponding LR versions, respectively. λ is a balance parameter. N is the number of training samples. Note that imposing penalties on depth maps of different resolutions is similar to the deeply-supervised network that guides the network training to predict output images at different scales.

5. Experimental results

5.1. Implementation and training details

We train our model based on the TensorFlow framework. All our experiments are conducted on a desktop PC with Intel 4.2GHz i7-7700k CPU, 32GB RAM and Nvidia 1080Ti 11GB GPU. We first train each sub-network separately with respective loss function. For contextual branch, we use the ResNet-101 network parameters pretrained on ImageNet to initialize the network, and randomly initialize the attention and multi-scale modules using the predefined TensorFlow function 'tf.random_normal_initializer'. We then finetune the model weights with its L2-norm loss function for 30 epochs. For spatial branch and refinement module, we randomly initialize their model weights, and train together with its L2-norm loss function for 15 epochs. Finally, all sub-networks are jointly fine-tuned using our proposed loss function with double supervision in Eq. 2 for the additional 30 epochs. We used Adam optimizer with momentum = 0.9, $\beta_1 = 0.9$, $\beta_2 = 0.99$ and $\epsilon = 10^{-8}$. The learning rate is initialized to 1e-4 for all layers and decreased by 0.9 every epoch. The parameter λ is set to 0.6.

For quantitative evaluation, we assess with the five performance metrics: 1) Rmse(lin): $\sqrt{\frac{1}{N}\sum_p (d_p^{gt} - d_p)^2}$; 2) Average Log_{10}: $\frac{1}{N}\sum_p |log_{10} d_p^{gt} - log_{10} d_p|$; 3) Rmse(log): $\sqrt{\frac{1}{N}\sum_p (Logd_p^{gt} - Logd_p)^2}$; 4) Abs Rel: $\frac{1}{N}\sum_p \frac{|d_p^{gt} - d_p|}{d_p^{gt}}$; 5) Accuracy with threshold thr: percentage (%) of d_p s.t. $\max(\frac{d_p^{gt}}{d_p}, \frac{d_p}{d_p^{gt}}) = \delta < thr$. In the following figures, warm color denotes farther distance, while cold color represents closer distance.

5.2. Performance comparison

In this section, an indoor dataset (NYU v2 dataset [50]) and two outdoor dataset (Make3D dataset [51] and KITTI dataset [52]) are used to conduct our performance comparison.

5.2.1. NYU V2 dataset

The benchmark RGB-D NYU v2 dataset [50] is used to conduct our experiments. NYU dataset is an indoor dataset with 120K pairs of RGB and depth maps gathered by a Microsoft Kinect. The image resolution is 640 × 480 pixels. The dataset is split into a training (249 scenes) and a test set (215 scenes). We follow the official splitting, and use 795 image pairs for training, and 654 for testing separately. We augment the training data with rotation and flipping operations. Note that, different from other methods that using at least 20K image pairs as their training subset sampled from the

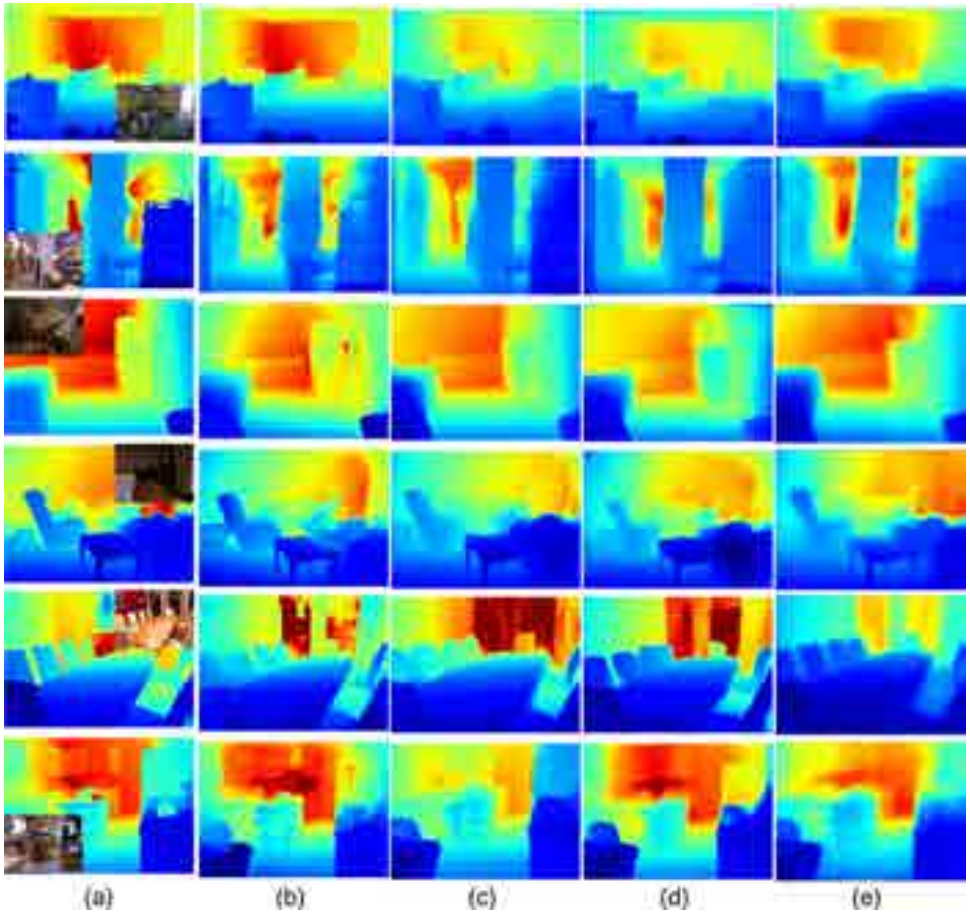

Fig. 8. Visual comparison results between different methods. From left to right, the images are (a) Ground truth depth maps and its corresponding color image; (b) Ours; (c) Chakrabarti et al. [49]; (d) Laina et al. [25]; (e) Lee et al. [20].

full dataset, our model is trained on a small labeled training subset with only 795 image pairs. We evaluate our results compared with other state-of-the-art methods [8,10,20,21,23,25,26,34,49]. The objective performances of the compared methods are provided by their respective papers and shown in Table 1. For fairly comparison, we separate the above methods into two parts: the ones in top part are trained on a larger dataset containing at least 20K image pairs, while others in bottom part trained on 795 image pairs obtained from the official splitting.

Compared with the methods that trained on the same training dataset with ours (bottom part), we achieve the lowest *Error* values on all the three measurements and the highest scores in *Accuracy* with all the three thresholds. Next, we compare with other methods that using training image pairs about 25 times as much as ours (top part). Our method achieves comparable scores under *Log10* and other three metrics in *Accuracy*, but obtains the lowest values in terms of *Rmse (lin)* and *Rmse (log)* metrics. Note that we improve the performance of *Rmse (lin)* and *Rmse (log)* by about 17% and 42% than the second best one (0.571, 0.193), respectively. This explains that *Rmse* measures the square dissimilarity between recovered depth and its corresponding groundtruth, and can amplify the errors when there appears obvious wrong estimation in some regions. Lower *Rmse* values correspond to better protect the estimated scene structure, which demonstrates our effectiveness to infer a precise scene depth map. The visual comparisons in Fig. 8 also verify our viewpoint. The results from Chakrabarti et al. [49] deform severely. Almost all the fine structures and object contours can not be preserved, and large areas are subjective to wrongly predicted depth values. Laina et al. [25] has a relative precise estimated geometry structure than the above method, but also fails to keep the depth boundaries sharp. Lee et al. [20] obtains a relatively higher *Accuracy* values. However, through closely observation, their results are obviously more blurry than ours, and cannot well protect scene structure and details. In general, our results are most similar to the groundtruth, and achieve appealing performance on both objective and visual comparison.

5.2.2. Make3D dataset

Make3D dataset [51] consists of 400 training and 134 testing outdoor images obtained from 3D laser scanner. The depth map resolution is 55 $\times$ 305, while the RGB images is 2272 $\times$ 1704. Following [53], we resize both color images and depth images to 460 $\times$ 345. Then, we augment the training data with rotation and flipping operations. For evaluation, we downsample the estimated depth map to 55 $\times$ 305 and compare against the ground-truth depth map in the original size. We only compute the errors in regions of depths less than 70\,m (C1 criterion). Table 2 shows the comparison results with other methods [10,21,23,25,53]. Early algorithms, such as [21,23,53], can not obtain satisfactory results for all three metrics. We generate comparable performances with Laina et al. [25] and Heo et al. [10] on *rel* and *Log_10* metrics, but obvious better performance than theirs on *Rmse* metric. Fig. 9 shows qualitative results. The results from Laina et al. are subjective to blurring artifacts, and cannot infer the right depth at the areas of far distance, while our method can obtain the most similar results compared to the groundtruths.

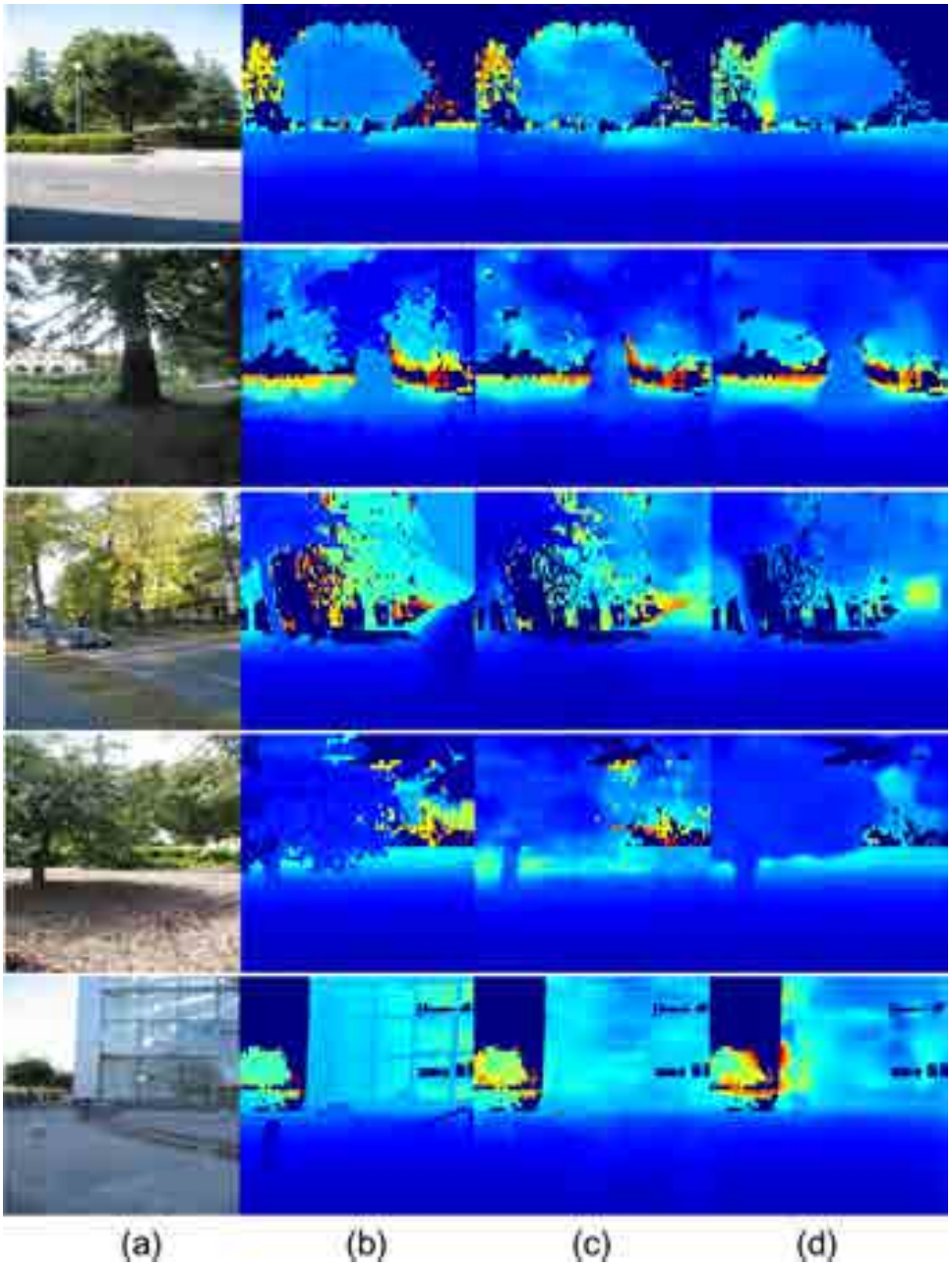

Fig. 9. Visual comparison on Make3D dataset: (a) color image; (b) GT; (c) Ours; (d) Laina et al. [25].

Table 2
Objective performance comparison on Make3D dataset.

Method	Error (Lower is better)		
	Abs Rel	Log10	RMSE (lin)
Liu et al. [23]	0.335	0.137	9.49
Liu et al. [53]	0.314	0.119	8.60
Li et al. [21]	0.278	0.092	7.19
Laina et al. [25]	0.176	0.072	4.46
Heo et al. [10]	**0.171**	0.063	4.46
Ours	**0.171**	**0.062**	**4.17**

5.2.3. KITTI Dataset

The KITTI dataset [52] contains sparse 3D laser measurements taken from a Velodyne laser sensor for outdoor scenes. The Eigen split [30] for KITTI dataset has 22,600 stereo image pairs for training and 697 stereo image pairs for testing. The input resolution for the proposed method is 512*256. We evaluate our results compared with other state-of-the-art methods [23,30,32,35–39,54–56], in which [23,30,32,54] are trained by using groundtruth depth as supervisory signal, while the rest ones using binocular images as supervision. Table. 3 lists the comparison results with the state-of-the-art methods. We achieve the noticeable improvements in most metrics (except for *RMSE log* and $\delta < 1.25^3$), which demonstrates the effectiveness of our method. Fig. 10 shows the qualitative results. The ground-truth depth map is interpolated from sparse measurements for visualization purpose. It can be seen that the proposed method is capable of preserving sharp boundaries at objects and restoring more accurate depth values for slim and distant objects.

5.3. Ablation study

In this section, we use the larger and more challenging NYU v2 dataset to conduct our ablation study.

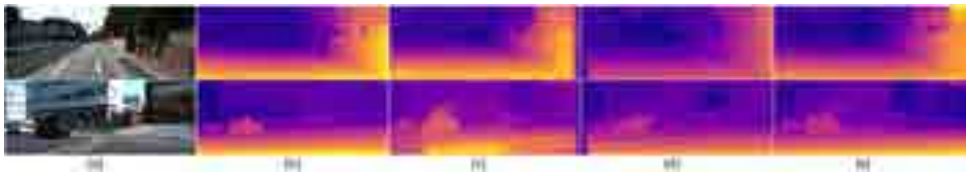

Fig. 10. Qualitative comparison with different methods on KITTI dataset [52]. (a) color image, (b) ground truth, (c) Zhan et al. [56], (d) Pilzer et al. [37], (e) Ours.

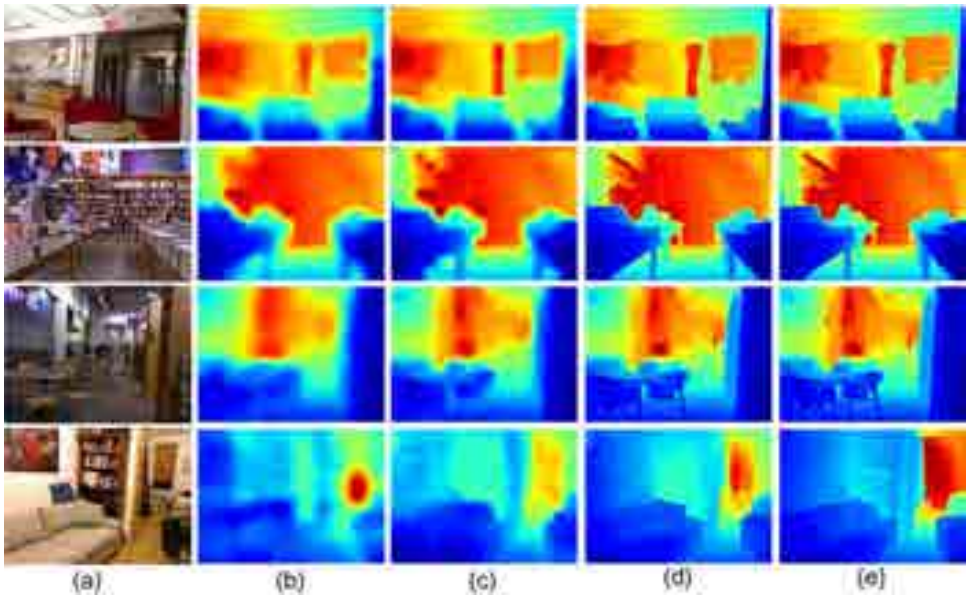

Fig. 11. Visual comparison results of different backbone networks: (a) Color images; Results generated by (b) VGG19; (c) ResNet50; (d) Our proposed baseline. (e) Groundtruth.

Table 3
Objective performance comparison on KITTI dataset using the Eigen split [30].

Method	Error (lower is better)			Accuracy (higher is better)		
	Abs Rel	RMSE(lin)	RMSE(log)	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Eigen et al. [30]	0.203	6.307	0.246	0.702	0.890	0.958
Liu et al. [23]	0.201	6.471	0.273	0.680	0.898	0.967
Nath Kundu et al. [54]	0.167	5.578	0.237	0.771	0.922	0.971
Xu et al. [32]	0.132	-	**0.162**	0.804	0.945	**0.981**
Godard et al. [55]	0.148	5.927	0.247	0.803	0.922	0.964
Zhan et al. [56]	0.144	5.869	0.241	0.803	0.928	0.969
Zhao et al. [35]	0.158	5.285	0.238	0.811	0.934	0.970
Chen et al. [36]	0.118	5.096	0.211	0.839	0.945	0.977
Pilzer et al. [37]	0.142	5.785	0.239	0.795	0.924	0.968
Wong and Soatto [38]	0.133	5.515	0.231	0.826	0.934	0.969
Puscas et al. [39]	0.135	5.582	0.235	0.828	0.933	0.967
Ours	**0.112**	**4.978**	0.210	**0.842**	**0.947**	0.973

Comparison between Different Backbones. We test the performance under different backbone networks. We replace our baseline network by other backbones, i.e., VGG19 and ResNet50, but keep all other modules unchanged for fair comparison. The objective comparison in Table 4 demonstrates the effectiveness of our baseline network, which is designed based on deeper ResNet101 model with dilated convolutions instead of the downsampling operator. Fig. 11 shows the visual comparison results generated from different backbone networks. As the network becomes deeper, the performance is better, i.e., the depth boundaries are more sharp, and the scene structures are protected better. It demonstrates the effectiveness of our baseline network, which is designed based on deeper ResNet101 model with dilated convolutions instead of the downsampling operator at the last two Res-Blocks.

Contributions of each component. To discover the vital elements in our proposed method, we conduct ablation study by gradually integrating each component into our model. The whole framework can be divided into five parts, i.e., the backbone network in contextual branch (baseline), multi-scale module (M), refinement module (RM), spatial branch (SB), nonlocal spatial attention module (NSA), and channel attention module (CA). The comparison results are shown in Table 5. The baseline alone [2] cannot obtain good results. When adding the multi-scale module, the modification of network structure leads to slight performance im-

[2] We use bicubic interpolation to directly upsample the 1/8 output depth map to the original size.

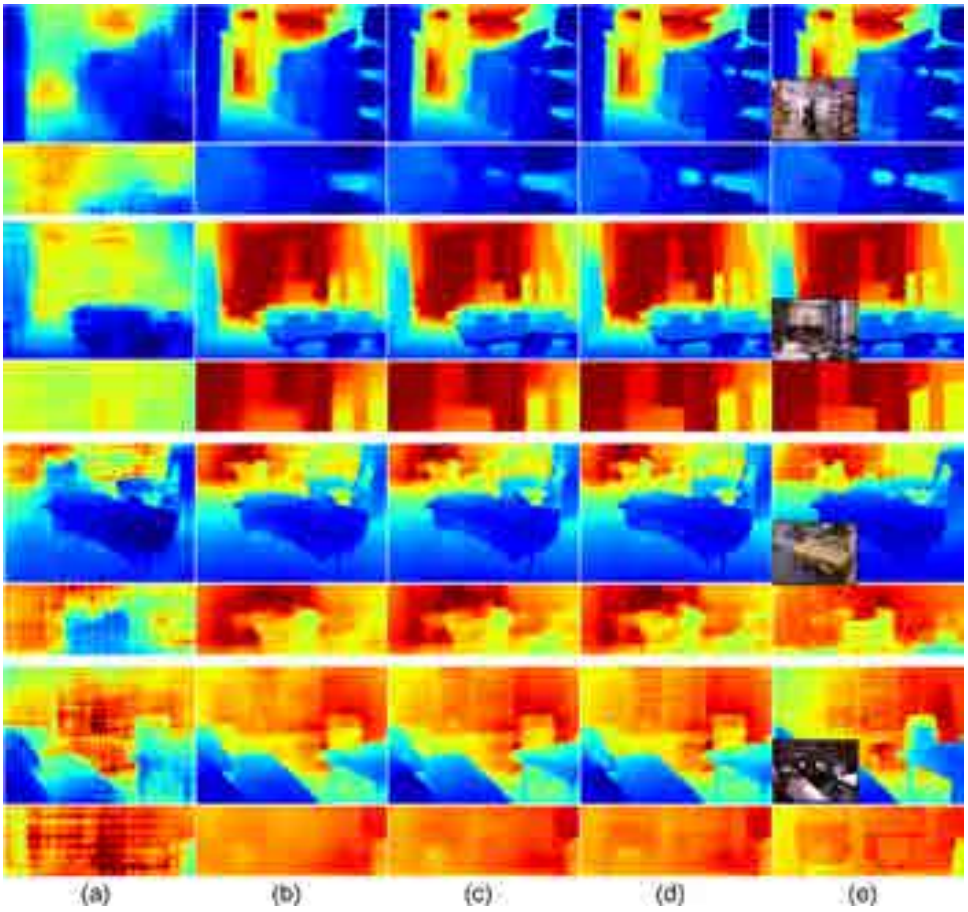

Fig. 12. Visual comparison results of different variants of our method: (a) Baseline; (b) Baseline + M + RM; (c) Baseline + M + RM + NSA; (d) Our full model; (e) Groundtruth depth maps and its corresponding color images.

Table 4
Objective Comparison between different backbone networks.

Backbone	Error (Lower is better)			Accuracy (Higher is better)		
	RMSE (lin)	Log10	RMSE (log)	δ <1.25	δ <1.25^2	δ <1.25^3
VGG19	0.626	0.084	0.106	66.8%	90.7%	97.3%
ResNet50	0.566	0.075	0.097	72.4%	92.5%	97.6%
Ours	**0.474**	**0.063**	**0.081**	**78.4%**	**94.8%**	**98.6%**

Table 5
Quantitative results of different variants of our method.

Method	M	RM	SB	NSA	CA	Error (Lower is better)		
						RMSE (lin)	Log10	RMSE (log)
Baseline						0.784	0.102	0.124
Ours	✓					0.737	0.095	0.118
	✓	✓				0.563	0.075	0.092
	✓	✓	✓			0.544	0.070	0.087
	✓	✓	✓	✓		0.476	0.064	**0.081**
	✓	✓	✓		✓	0.478	0.067	**0.081**
	✓	✓	✓	✓	✓	**0.474**	**0.063**	**0.081**

provement, e.g. in case of *Rmse (lin)* from 0.784 to 0.737. Next, our depth refinement module can be regarded as an advanced depth enhancement method, which improves the performance of *Rmse (lin)* from 0.737 to 0.563. When introducing HR features from spatial branch, the performance sustains growth. Moreover, our spatial and channel attention modules serve as different roles in extracting the informative features, i.e., NSA focuses on strengthening the nonlocal spatial relationship within a feature channel while CA emphasizes the useful feature channels along the channel dimension. Though CA brings less performance improvement than NSA, but still contributes to the final performance. Besides, the computational cost of CA is very small and can be ignored. Therefore, CA can be added to the whole framework to pursue the ultimate performance without increasing the computation burden. As a re-

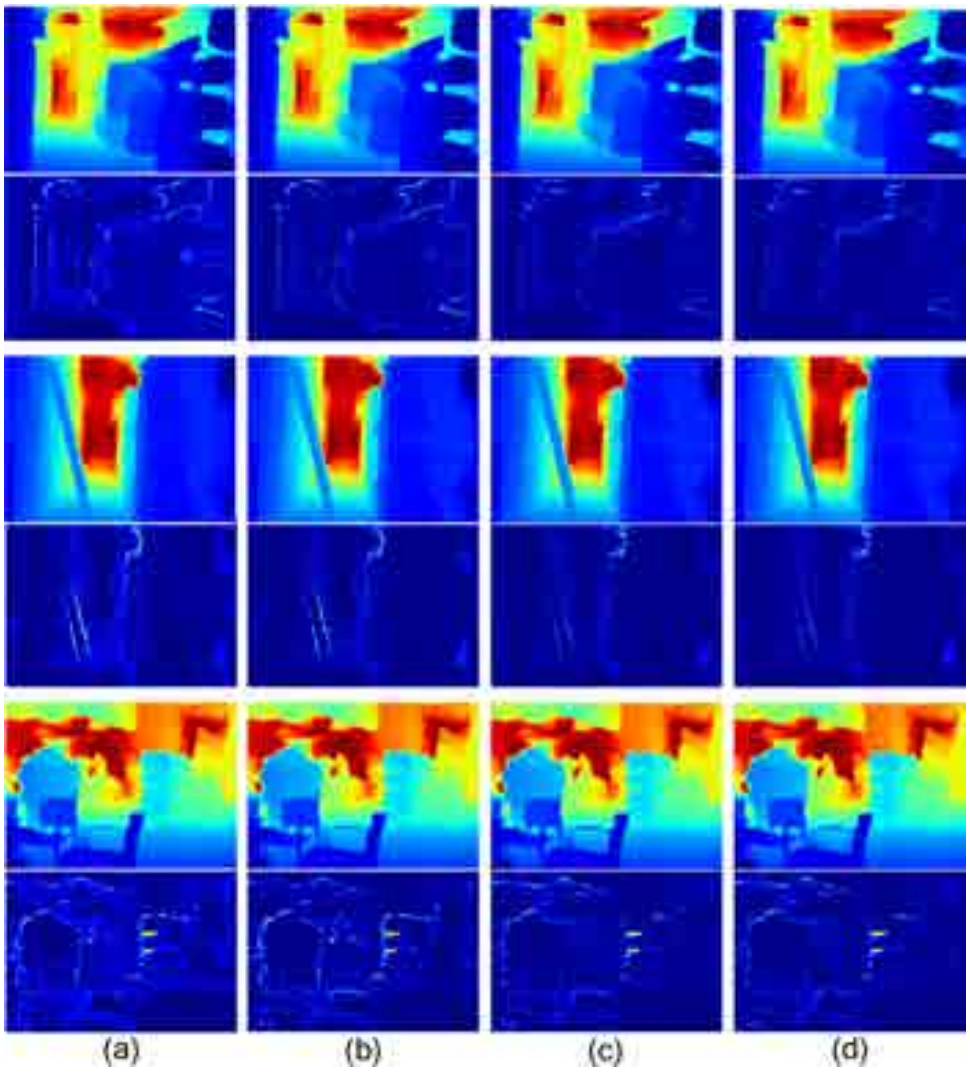

Fig. 13. Visual comparison between different neighborhood size: (a) k = 0; (b) k = 3; (c) k = 6; (d) NLB [19] (k equals to image size). The error maps between groudtruths and generated results are shown for clear visualization (bright color for large errors). As k increases, the inferred depth maps are more accurate.

Table 6
Objective comparison between different neighborhood size. *n, c, k* represent image size, the number of channels, and neighborhood size respectively.

	Error (Lower is better)			Complexity	Time (ms)
K	RMSE (lin)	Log10	RMSE (log)		
0	0.484	0.065	0.083	$O(4nck^2)$	34.0 ~ 48.4
3	0.481	0.065	0.083		
6	0.476	0.064	0.082		
8	**0.474**	**0.063**	**0.081**		
10	**0.474**	**0.063**	**0.081**		
Global	**0.474**	**0.063**	**0.081**	$O(n^2c)$	91.7

sult, the complete proposed framework with all the parts provides state-of-the-art performance. As shown in Fig. 12, when integrating with each module in turn, the scene structures are more clearer and protected better.

Analysis of nonlocal neighborhood *k*. To verify the effectiveness of our revised spatial attention module, we test the impact on the estimation performance under different neighborhood sizes (from 3 × 3 to 21 × 21). Note that the case of $k = 0$ degenerates the attention module to ordinary local convolutions. When the neighborhood size equals to the image size, the "LocalMul" operation approaches to the global cross correlation (Global) proposed in Wang *et.al* (NLB for short) [19]. Table 6 shows the objective comparison results. The performance improves as k gets larger. However, the performance saturates when k is bigger than 8. Note that our spatial attention module with a relative small $k = 6$ still yields competitive results, but has a low complexity of $\mathcal{O}(4nck^2)$ compared to $\mathcal{O}(n^2c)$ of NLB [19] in the construction of the weighting matrix F, since the size k is far smaller than the image size $n = h \times w$. Fig. 13 shows the visual comparison of different neighborhood size. Obviously, the case of $k = 0$ is subject to blurry artifacts and wrong estimations, while the cases of $k = 3$ achieves relatively better performance. The cases of $k = 6$ and NLB generate results with almost no difference compared to each other, but both are more clear than those of $k = 3$. As a result, the proposed spatial attention ($k = 6$) could be a valuable method to be implemented in faster depth estimation where the running time is largely shortened than NLB.

Comparison on different attention modules. For fair comparison, we test the performances of different attention modules based on our proposed framwork and show the results in Table 7. For the spatial attention module only (top part), NLB [19] and ours achieve better performance than Affinity [34], which demonstrates the effectiveness of the explicit design of positional correspondence. For

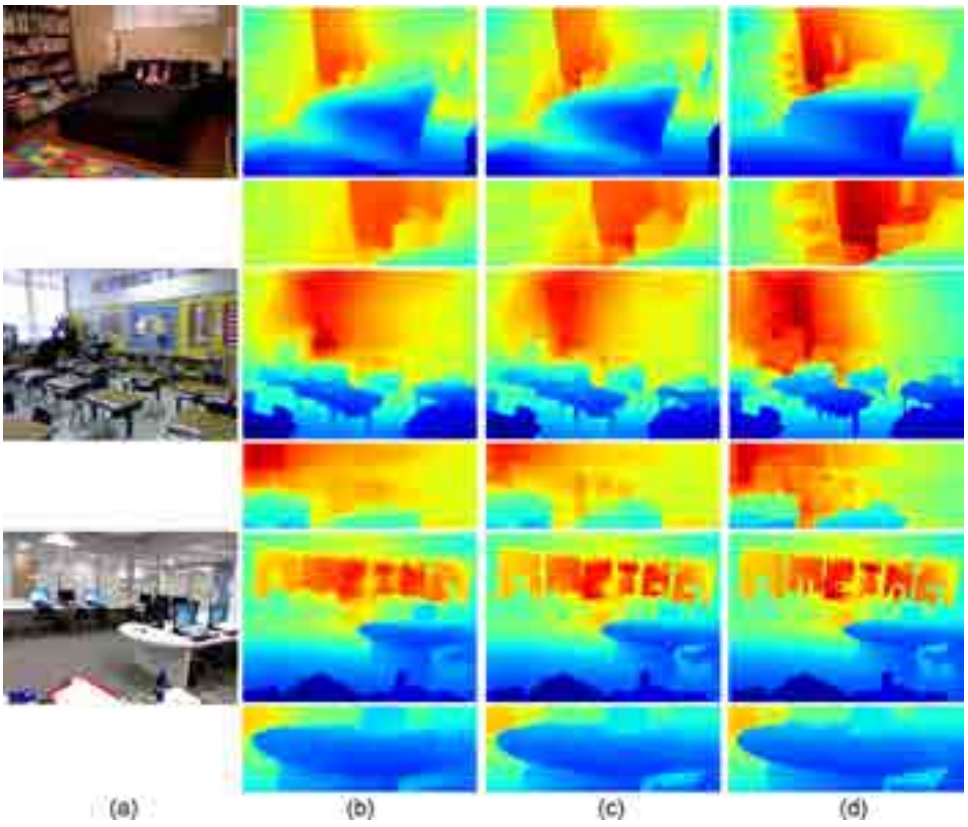

Fig. 14. Visual comparison between CBAM [16] and ours: (a) Color images; Results generated by (b) CBAM [16] and (c) ours; (d) Groundtruth. Note that CBAM and our proposed method both exploit spatial and channel interdependencies, but with different design methodologies. The visual comparisons verify that our proposed attention module obtains more clear object contours and protects fine details and scene structures better than CBAM.

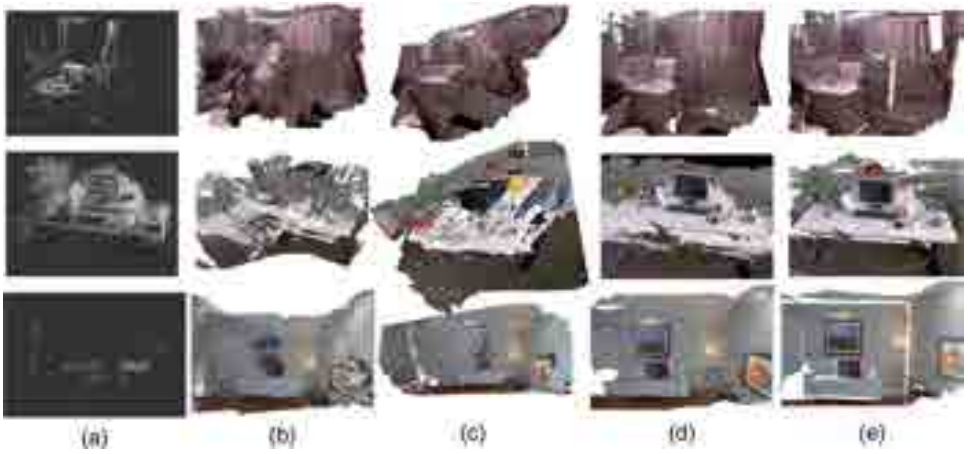

Fig. 15. Qualitative 3D reconstruction results on three scenes. (a) LSD-SLAM, (b) Chakrabarti et al. [49], (c) Laina et al. [25], (d) ours, (e) groundtruth.

Table 7
Objective Comparison between different attention modules. *n, c, k* represent image size, the number of channels, and neighborhood size respectively. The top part, middle part, and bottom part represent the comparison on spatial attention, channel attention and dual attention, respectively.

Module	Error (Lower is better)			Complexity	Time (ms)
	RMSE (lin)	Log10	RMSE (log)		
Affinity [34]	0.631	0.066	0.102	$O(4nck^2)$	49.3
NLB [19]	0.475	0.064	**0.081**	$O(n^2c)$	91.7
Ours_s (k = 6)	0.476	0.064	0.082	$O(4nck^2)$	41.5
SENet_c [18]	0.492	0.070	0.116	-	47.2
Ours_c	0.475	0.064	**0.081**	$O(nc^2)$	40.3
CBAM [16]	0.481	0.064	0.095	-	100.4
Ours_full	**0.474**	**0.063**	**0.081**	$O(nc(c+4k^2))$	80.3

Fig. 16. Different views of our reconstruction results are shown for better visualization.

the channel attention module only (middle part), we achieve better performance and lower computation complexity and runtime than SENet [18]. For dual attention module, we also obtain more appealing results than CBAM [16] that uses simple convolutions and global average pooling to capture contextual dependencies in both spatial and channel dimensions. In terms of running time, Affinity [34] is slightly lower than Ours_s due to the usage of fully connected layer. The inference times for Ours_s and Ours_full also decrease by 50.2ms and 20.1ms compared to NLB and CBAM, respectively.

5.4. Discussions

CBAM [16] also emphasizes the importance to extract informative features by blending cross-channel and spatial information together, which can boost representation power of CNNs. The channel attention of CBAM can be regarded as an improved version of SENet [18], which uses average pooling and max-pooling instead of global pooling from SENet to generate two context descriptors. Then, both CBAM and SENet produce the final channel attention map through two successive fully connected layers, which are subject to more training and test cost than Ours_full and Ours_c, respectively. In contrast, our attention module is free of training parameters, and the runtime is largely shortened by compressing the number of input feature channels into 32 before sending into the channel attention module, which facilitates the computation along the channel dimension.

For spatial attention module, CBAM first applies pooling operations along the channel axis, and then use a convolution layer to generate a spatial attention map. However, there are two disadvantages. First, using convolutions to aggregate features will lose the latent positional correspondence between neighboring pixels. Second, only one convolution layer with the kernel size of 7×7 (the half window size k equals 3) is not enough to capture nonlocal dependencies. Compared to CBAM, we resort to linear algebra based attention module [19] to keep the latent positional correspondence between neighboring pixels and channels, and use the nonlocal cross correlation to balance the algorithm complexity and performance. For fairly comparison, we test the performance of CBAM based on our full model, only replacing our proposed dual attention module. We add extra visual comparisons in Fig. 14. For all the presented cases, we obtain more appealing results than CBAM module [16]. Note that, our proposed method can keep the object contours sharper, and protect fine details and scene structures better than CBAM.

Table 8
Runtime comparison between different methods. The top part and bottom part represent the comparison on NYU and KITTI datasets, respectively. The input test images are resized to 640×480 for all the methods from both parts.

Method	Error (Lower is better)		Time
	RMSE (lin)	RMSE (log)	(ms)
Eigen and Fergus [8]	0.907	0.283	201.3
Liu et al. [23]	0.824	-	175.2
Chakrabarti et al. [49]	0.620	0.205	150.3
Laina et al. [25]	0.573	0.204	72.4
Ours_ResNet50	0.566	0.097	**70.3**
Ours	**0.474**	**0.081**	80.3
Godard et al. [55]	5.927	0.247	87.4
Pilzer et al. [37]	5.785	0.239	93.2
Ours_ResNet50	**5.024**	**0.215**	**70.3**

Besides, runtime (ms) is measured between different approaches as shown in Table 8. For fairly comparison, the test images are resized to 640×480 (about 0.29 MP) for all the methods and all the datasets. For NYU dataset (top part), since Laina et al. [25] constructs their network based on ResNet50, we also give the runtime of our method with ResNet50 as the backbone

(Ours_ResNet50) for fairly comparison. Note that we achieves both the best performance (underlined numbers) and lowest runtime than Laina et al. For KITTI dataset, both compared methods use ResNet50 as their encoder and a symmetric ResNet50 as decoder. In contrast, our method has a light-weight decoder with only three up-projection block and a fusion block, which spends less inference time than other methods and achieves more superior results due to our careful designed encoder.

5.5. Comparison under the vSLAM framwork

The rapid development of visual simultaneous localization and mapping (vSLAM) [57] has created a new visualization and sensing wave for computer vision community. Although the tracking performance of such algorithm is impressive, the generated 3D map is extremely sparse and cannot be used in practical application. We therefore integrate these CNN-based dense depth estimation methods into the SLAM framework to show the effectiveness in addressing the dense scene reconstruction. More importantly, we will compare the performance of different depth estimation methods under the vSLAM framework.

Three video scenes, i.e., 'bathroom 0003' from NYU dataset [58], 'fr1 rpy' from TUM SLAM dataset [59], and 'lr kt0' from ICL-NUIM SLAM dataset [60] are used to conduct our experiments. The tracking poses are extracted from LSD-SLAM [57], and used for dense reconstruction together with the depth estimation results.

As shown in Fig. 15, the original SLAM framwork, i.e., LSD-SLAM, can only obtain very sparse 3D reconstruction, which can not be applied in real conditions. Once combined with dense depth map inferred from CNN, the reconstructed scenes are more distinct. Compared between Chakrabarti et al. [49], Laina et al. [25], and our method, the 3D reconstruction results from ours present precise and undistorted scenes, which are more similar to the groundtruth. Fig. 16 presents our reconstruction results from different views. Visual results demonstrate the effectiveness of our depth estimation network.

6. Conclusion and future work

We present DPNet to fully address the problems of inaccurate depth inference and spatial information loss. Specifically, in contextual branch (CB), we propose an effective and efficient nonlocal spatial attention module by introducing non-local filtering strategy to explicitly exploit the pixel relationship in spatial domain, which can bring significant promotion on depth details inference. Meanwhile, we design a spatial branch (SB) to preserve the spatial information and generate high-resolution features from input color image. A refinement module (RM) is then proposed to fuse the heterogeneous features from both spatial and contextual branches to obtain a high quality depth map. Experimental results show that the proposed method outperforms SOTA methods on benchmark RGB-D datasets.

Our future work lies in two aspects: 1) focusing on dense reconstruction based on vSLAM framework. Note that we only test the performance of depth estimation with the help of camera poses extracted by SLAM. How to use the sparse depth points extracted from ORB features in SLAM to help improve the performance of our depth estimation task is a key point. Besides, the estimated depth can also be used in turn to help to better estimate the camera pose, and therefore make the tracking process more robust. 2) The use of unsupervised manner and domain adaptation techniques. Due to the lack of effective paired training data, and the weak generalization ability when training and testing on different datasets, these techniques are desirable to address the real problem in monocular depth estimation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by National Natural Science Foundation of China (NSFC) under Grant 61702078, 61772106.

References

[1] T. Uricchio, L. Ballan, L. Seidenari, A. Del Bimbo, Automatic image annotation via label transfer in the semantic space, Pattern Recognit. 71 (2017) 144–157.
[2] Y. Wang, J. Liu, Y. Li, J. Fu, M. Xu, H. Lu, Hierarchically supervised deconvolutional network for semantic video segmentation, Pattern Recognit. 64 (2017) 437–445.
[3] A. Martín, A. Adán, 3D real-time positioning for autonomous navigation using a nine-point landmark, Pattern Recognit. 45 (1) (2012) 578–595.
[4] T. Kajihara, T. Funatomi, H. Makishima, T. Aoto, H. Kubo, S. Yamada, Y. Mukaigawa, Non-rigid registration of serial section images by blending transforms for 3d reconstruction, Pattern Recognit. 96 (2019) 106956.
[5] L. Kang, L. Wu, Y. Wei, S. Lao, Y.-H. Yang, Two-view underwater 3d reconstruction for cameras with unknown poses under flat refractive interfaces, Pattern Recognit. 69 (2017) 251–269.
[6] A. Kolb, E. Barth, R. Koch, R. Larsen, Time-of-flight cameras in computer graphics, Comput. Graphics Forum 29 (1) (2010) 141–159, doi:10.1111/j.1467-8659.2009.01583.x.
[7] F. Cheng, X. He, H. Zhang, Learning to refine depth for robust stereo estimation, Pattern Recognit. 74 (2018) 122–133.
[8] D. Eigen, R. Fergus, Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture, in: IEEE ICCV, 2015, pp. 2650–2658.
[9] H. Fu, M. Gong, C. Wang, K. Batmanghelich, D. Tao, Deep ordinal regression network for monocular depth estimation, IEEE CVPR, 2018.
[10] M. Heo, J. Lee, K.-R. Kim, H.-U. Kim, C.-S. Kim, Monocular depth estimation using whole strip masking and reliability-based refinement, ECCV, 2018.
[11] X. Shen, et al., Mutual-structure for joint filtering, IEEE ICCV, 2015.
[12] J. Yang, Color-guided depth recovery from RGB-D data using an adaptive autoregressive model, IEEE TIP 23 (8) (2015) 3443–3458.
[13] J. Ahn, S. Kwak, Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation, IEEE CVPR, 2018.
[14] S. Liu, S.D. Mello, J. Gu, G. Zhong, M.H. Yang, J. Kautz, Learning affinity via spatial propagation networks, NIPS, 2017.
[15] T.-W. Ke, J.-J. Hwang, Z. Liu, S. Yu, Adaptive affinity field for semantic segmentation, ECCV, 2018.
[16] S. Woo, J. Park, J.Y. Lee, I.S. Kweon, Cbam: Convolutional block attention module, ECCV, 2018.
[17] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, X. Tang, Residual attention network for image classification, in: IEEE CVPR, 2017, pp. 6450–6458.
[18] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, IEEE CVPR, 2017.
[19] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, IEEE CVPR, 2018.
[20] J.-H. Lee, M. Heo, K.-R. Kim, C.-S. Kim, Single-image depth estimation based on fourier domain analysis, IEEE CVPR, 2018.
[21] B. Li, C. Shen, Y. Dai, A. van den Hengel, M. He, Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs, in: IEEE CVPR, 2015, pp. 1119–1127.
[22] D. Xu, W. Wang, H. Tang, H. Liu, N. Sebe, E. Ricci, Structured attention guided convolutional neural fields for monocular depth estimation, IEEE CVPR, 2018.
[23] F. Liu, C. Shen, G. Lin, I. Reid, Learning depth from single monocular images using deep convolutional neural fields, IEEE TPAMI 38 (10) (2016) 2024–2039.
[24] Y. Kim, H. Jung, D. Min, K. Sohn, Deep monocular depth estimation via integration of global and local predictions, IEEE TIP PP (99) (2018). 1–1
[25] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, N. Navab, Deeper depth prediction with fully convolutional residual networks, in: Fourth International Conference on 3d Vision, 2016, pp. 239–248.
[26] J. Li, R. Klein, A. Yao, A two-streamed network for estimating fine-scaled depth maps from single rgb images, IEEE ICCV, 2017.
[27] A. Saxena, S.H. Chung, A.Y. Ng, Learning depth from single monocular images, in: NIPS, 2005, pp. 1161–1168. Cambridge, MA, USA
[28] K. Karsch, C. Liu, S.B. Kang, Depth transfer: depth extraction from video using non-parametric sampling, IEEE TPAMI 36 (11) (2014) 2144.
[29] S. Choi, D. Min, B. Ham, Y. Kim, C. Oh, K. Sohn, Depth analogy: data-driven approach for single image depth estimation using gradient samples, IEEE TIP 24 (12) (2015) 5953.
[30] D. Eigen, C. Puhrsch, R. Fergus, Depth map prediction from a single image using a multi-scale deep network, in: NIPS, 2014, pp. 2366–2374.
[31] O. Ronneberger, P. Fischer, T. Brox, U-Net: convolutional networks for biomedical image segmentation 9351 (2017) 234–241.

[32] D. Xu, E. Ricci, W. Ouyang, X. Wang, N. Sebe, Multi-scale continuous crfs as sequential deep networks for monocular depth estimation, in: IEEE CVPR, 2017, pp. 161–169.
[33] X. Cheng, P. Wang, R. Yang, Depth estimation via affinity learned with convolutional spatial propagation network, ECCV, 2018.
[34] Y. Gan, X. Xu, W. Sun, L. Lin, Monocular depth estimation with affinity, vertical pooling, and label enhancement, ECCV, 2018.
[35] S. Zhao, H. Fu, M. Gong, D. Tao, Geometry-aware symmetric domain adaptation for monocular depth estimation, IEEE CVPR, 2019.
[36] P.-Y. Chen, A.H. Liu, Y.-C. Liu, Y.-C.F. Wang, Towards scene understanding: unsupervised monocular depth estimation with semantic-aware representation, in: IEEE CVPR, 2019, pp. 2624–2632.
[37] A. Pilzer, S. Lathuiliere, N. Sebe, E. Ricci, Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation, IEEE CVPR, 2019.
[38] A. Wong, S. Soatto, Bilateral cyclic constraint and adaptive regularization for unsupervised monocular depth prediction, IEEE CVPR, 2019.
[39] M.M. Puscas, D. Xu, A. Pilzer, N. Sebe, Structured coupled generative adversarial networks for unsupervised monocular depth estimation, in: International Conference on 3D Vision, 2019.
[40] M. Dongbo, L. Jiangbo, M.N. Do, Depth video enhancement based on weighted mode filtering, IEEE Trans. Image Process. 21 (3) (2012) 1176–1190.
[41] J. Yang, X. Ye, P. Frossard, Global auto-regressive depth recovery via iterative non-local filtering, IEEE Trans. Broadcast. PP (99) (2018) 1–15.
[42] P. Jaesik, K. Hyeongwoo, T. Yu-Wing, M.S. Brown, K. In So, High-quality depth map upsampling and completion for rgb-d cameras, IEEE Trans. Image Process. 23 (12) (2014) 5559–5572.
[43] A. Levin, D. Lischinski, Y. Weiss, A closed form solution to natural image matting, IEEE TPAMI 30 (2) (2007) 228–242.
[44] Q. Yan, L. Xu, J. Shi, J. Jia, Hierarchical saliency detection, in: IEEE CVPR, 2013, pp. 1155–1162.
[45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, NIPS, 2017.
[46] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE CVPR, 2016, pp. 770–778.
[47] L.C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE TPAMI PP (99) (2017). 1–1
[48] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, CoRR (2015). abs/1511.07122
[49] A. Chakrabarti, J. Shao, G. Shakhnarovich, Depth from a single image by harmonizing overcomplete local network predictions, in: NIPS, 2016, pp. 2658–2666.
[50] N. Silberman, D. Hoiem, P. Kohli, R. Fergus, Indoor segmentation and support inference from rgbd images, in: ECCV, 2012, pp. 746–760.
[51] A. Saxena, S. Min, A.Y. Ng, Make3d: learning 3d scene structure from a single still image 31(5) (2009) 824–840.
[52] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, Vision meets robotics: the KITTI dataset, Int. J. Rob. Res. 32 (11) (2013) 1231–1237.
[53] M. Liu, M. Salzmann, X. He, Discrete-continuous depth estimation from a single image, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014.
[54] J. Nath Kundu, P. Krishna Uppala, A. Pahuja, R. Venkatesh Babu, Adadepth: Unsupervised content congruent adaptation for depth estimation, in: IEEE CVPR, 2018, pp. 2656–2665.
[55] C. Godard, O. Mac Aodha, G.J. Brostow, Unsupervised monocular depth estimation with left-right consistency, IEEE CVPR, 2017.
[56] H. Zhan, R. Garg, C. Saroj Weerasekera, K. Li, H. Agarwal, I. Reid, Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction, IEEE CVPR, 2018.
[57] J. Engel, T. Schöps, D. Cremers, LSD-SLAM: Large-scale direct monocular SLAM, in: European Conference on Computer Vision, Springer, 2014, pp. 834–849.
[58] N. Silberman, D. Hoiem, P. Kohli, R. Fergus, Indoor segmentation and support inference from rgbd images, in: European Conference on Computer Vision, 2012, pp. 746–760.
[59] J. Sturm, N. Engelhard, F. Endres, W. Burgard, D. Cremers, A benchmark for the evaluation of rgb-d SLAM systems, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 573–580.
[60] A. Handa, T. Whelan, J. Mcdonald, A.J. Davison, A benchmark for rgb-d visual odometry, 3d reconstruction and SLAM, in: IEEE International Conference on Robotics and Automation, 2014, pp. 1524–1531.

Xinchen Ye, received the B.E. degree and Ph.D. degree from the Tianjin University, Tianjin, China, in 2012 and 2016, respectively. He was with the Signal Processing Laboratory, EPFL, Lausanne, Switzerland in 2015 under the Grant of the Swiss federal government. He has been a Faculty Member of Dalian University of Technology, Dalian, Liaoning, China, since 2016, where he is currently an Associate Professor with the DUT-RU International School of Information Science and Engineering. His current research interests include image/video processing and 3D imaging. As a co-author, he received the Platinum Best Paper Award in the IEEE ICME 2017. He won the Rising Star Award in 2018 ACM Turing Celebration Conference-China (ACM TURC 2018).

Shude Chen is currently a undergraduate at the School of Software in Dalian University of Technology. His research interests include computer vision and deep learning.

Rui Xu received the B.S. and M.S. degrees in 2001 and 2004, respectively from the school of electronic and information, South China University of Technology. He received the Ph.D. degree in 2007 from the graduate school of science and engineering, Ritsumeikan University, Japan. He worked in Sanyo Electric Co., Ltd., Japan, from 2008 to 2010. He worked in Yamaguchi University and Ritsumeikan University from 2010 to 2015. Since December 2015, he has been served as an associate professor at Dalian University of Technology. His research fields include intelligent computing in medical images and computer vision.