SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

November 2020, Vol. 63 212101:1–212101:11
https://doi.org/10.1007/s11432-019-2803-x

InStereo2K: a large real dataset for stereo matching in indoor scenes

Wei BAO[1], Wei WANG[1,2], Yuhua XU[1,2*], Yulan GUO[3,4*], Siyu HONG[3] & Xiaohu ZHANG[5]

[1]*School of Electrical Engineering and Automation, Hefei University of Technology, Hefei* 230009*, China;*
[2]*Orbbec Research, Shenzhen* 518052*, China;*
[3]*School of Electronics and Communication Engineering, Sun Yat-sen University, Guangzhou* 510006*, China;*
[4]*College of Electronic Science and Technology, National University of Defense Technology, Changsha* 410073*, China;*
[5]*School of Aeronautics and Astronautics, Sun Yat-sen University, Guangzhou* 510006*, China*

Received 31 August 2019/Revised 30 November 2019/Accepted 17 January 2020/Published online 31 July 2020

Abstract Deep neural networks have shown great success in stereo matching in recent years. On the KITTI datasets, most top performing methods are based on neural networks. However, on the Middlebury datasets, these methods usually do not perform well. The KITTI datasets are collected in outdoor scenes while the Middlebury datasets are collected in indoor scenes. It is commonly believed that the community still lacks a large labelled dataset for stereo matching in indoor scenes. In this paper, we introduce a new stereo dataset called InStereo2K. It contains 2050 pairs of stereo images with highly accurate groundtruth disparity maps, including 2000 pairs for training and 50 pairs for test. Experimental results show that our dataset can significantly improve the performance of several latest networks (including StereoNet and PSMNet) on the Middlebury 2014 dataset. The large scale, high accuracy and rich diversity of the proposed InStereo2K dataset provide new opportunities to researchers in the area of stereo matching and beyond. It also takes end-to-end stereo matching methods a step towards practical applications.

Keywords stereo matching, depth estimation, convolutional neural network, dataset

Citation Bao W, Wang W, Xu Y H, et al. InStereo2K: a large real dataset for stereo matching in indoor scenes. Sci China Inf Sci, 2020, 63(11): 212101, https://doi.org/10.1007/s11432-019-2803-x

1 Introduction

Stereo matching is a key step in 3D reconstruction. It has numerous applications in the fields of autonomous driving [1,2], robotics [3], UAVs, augmented reality (AR), and 3D modeling [4,5]. It takes two rectified images as its input and establishes dense correspondences between pixels of these two images to compute disparities. In practice, foreground-background occlusion is inevitable and makes the task really challenging. In addition, feature matching is also ambiguous in scenes with low or repetitive textures [6].

In recent years, deep convolutional neural network (CNN)-based stereo matching methods have shown great success. These end-to-end disparity computation networks have achieved good performance in both speed and accuracy [7]. On the KITTI datasets [1,2], most of the top ranking methods are based on neural networks. However, on the Middlebury datasets, these end-to-end networks do not perform well. Note that, the KITTI datasets are acquired in outdoor scenes, while the Middleburry datasets are collected in indoor scenes. A major reason for the performance drop on the Middleburry datasets lies in the insufficiency of training datasets for indoor scenes. In this paper, we introduce a new large

* Corresponding author (email: xyh_nudt@163.com, guoyulan@sysu.edu.cn)

dataset, namely, InStereo2K. The InStereo2K dataset contains 2050 pairs of stereo images with highly accurate groundtruth disparity maps, which are obtained using a structured light system. Experimental results show that our dataset can improve the performance of several state-of-the-art networks (including StereoNet [8] and PSMNet [9]) significantly on the Middlebury dataset [10].

The main contributions of this study are as follows:

(1) We introduce a new stereo dataset called InStereo2K. It contains 2050 pairs of stereo images with highly accurate disparity maps (2000 for training and 50 for test). To the best of our knowledge, it is the largest publicly available stereo dataset for indoor real scenes. It is an order of magnitude larger than existing stereo datasets, including KITTI [1, 2] and Middlebury [10, 11]. It can be used to train deep neural networks and to comprehensively test stereo matching methods. The dataset is available at the web[1)].

(2) We introduce an approach to improve the generalization performance of stereo matching networks using our real dataset and the existing synthetic dataset. We also present several practical training strategies to improve a network's performance. Using our dataset and training strategies, the ranking of PSMNet on the test set of the Middlebury 2014 benchmark is improved by 26.

2 Related work

In this section, we briefly review related work on datasets and deep convolutional neural networks for stereo matching.

2.1 Stereo datasets

The Middlebury stereo dataset [10] has been widely used in the evaluation of stereo matching methods. The disparity maps in this dataset were calculated using structured-light techniques and they were very accurate. However, this dataset only contains dozens of image pairs and is insufficient to train a deep neural network. The KITTI datasets [1, 2] were collected for automotive driving scenarios. The KITTI stereo 2012 dataset consists of 194 training image pairs and 195 test image pairs with a resolution of 1242 $\times$ 375 pixels, where the disparity labels were transformed from Velodyne LiDAR points. The KITTI stereo 2015 dataset consists of 200 training scenes and 200 test scenes. Compared to KITTI 2012, it comprises dynamic scenes for which the groundtruth has been established in a semi-automatic process. The ETH3D dataset [12] consists of 27 training image pairs and 20 test image pairs with a resolution of about 0.3 MP. The Scene Flow dataset [13] is a synthetic dataset, which contains 35855 pairs of stereo images. It has significantly boosted the research of CNN-based stereo matching methods. However, there is a huge gap between the synthetic domain and the real domain. To achieve improved performance, existing deep stereo matching networks are usually trained on this synthetic dataset and then fine-tuned on real but small specific datasets (e.g., the KITTI dataset [1, 2]).

2.2 Stereo matching networks

Traditional stereo matching methods commonly consist of four steps, including matching cost calculation, cost aggregation, disparity calculation, and disparity refinement [14]. MC-CNN [15] is the first method to calculate the matching cost between two image patches using a deep convolutional neural network. Meanwhile, its remaining steps still follow a traditional approach, including cross-based cost aggregation, semi-global matching, left-right consistency check, sub-pixel interpolation, median filtering and bilateral filtering [16]. This architecture needs multiple forward passes to calculate matching costs at all possible disparities. Therefore, the computational complexity of this method is high. Following MC-CNN [15], several methods were proposed to improve the computational efficiency [17] and matching accuracy [18].

Different from the architecture of MC-CNN, DispNet [13] used an end-to-end encoder-decoder architecture for disparity regression. The feature extraction and cost calculation steps were seamlessly integrated

1) https://github.com/yuhuaxu/stereodataset.

to the encoder part. The disparities were directly regressed in a forward pass. The end-to-end DispNet can run efficiently, with 0.06 s being consumed on a single Nvidia GTX Titan X GPU. GC-Net [19] introduced the concept of cost volume in traditional stereo matching into disparity estimation networks. Specifically, GC-Net used 3D convolutions upon a 4D cost volume to incorporate contextual information and used a differentiable soft argmin module to regress disparities. In StereoNet [8], disparity was first estimated from a very low resolution (e.g., 1/8 resolution) cost volume. The disparity was then hierarchically up-sampled and refined using a pixel-to-pixel refinement network, which leveraged image colors as a guide. The network can run at 60 fps on a Titan X GPU. iResNet [20] incorporated all four steps of stereo matching by explicitly introducing a residual network for disparity refinement. iResNet ranked the first in the stereo matching task of robust vision challenge, which was held in conjuction with CVPR 2018. PSM-Net [9] used pyramid feature extraction and a stacked hourglass block [21] with twenty-five 3D convolutional layers to further improve the accuracy. In GA-Net [22], a semi-global aggregation layer (which was a differentiable approximation of semi-global matching) and a local guided aggregation layer (which followed a traditional cost filtering strategy to refine thin structures) were proposed. It outperformed GC-Net with fewer parameters.

Although notable progresses have been achieved in stereo matching for deep neural network-based methods, the performance on indoor scenes is still limited. Therefore, a large indoor stereo dataset is highly required for the development of new methods.

3 Our dataset

In this section, we first present the system for the collection of the InStereo2K dataset. Then, we describe our new dataset in details.

3.1 The structured light system

Because structured light systems have the advantage of high accuracy in 3D reconstruction [23], we designed an active-stereo 3D imaging system to obtain accurate groundtruth disparity maps. The system consists of two color cameras with a resolution of 1280 × 960, and a projector with a resolution of 1024 × 768, as shown in Figure 1. The pixel size of the camera's CCD sensor is 3.75 μm. The lens of each camera (i.e., Computar M0814-MP2) has a focal length of 8 mm. Given focal length F, baseline B, and depth Z, the disparity d is obtained by $d = \frac{BF}{Z}$ for a stereo vision system. Therefore, the disparity d is proportional to baseline B. To obtain disparities with relatively uniform distribution, we used two different baselines, i.e., 5 cm and 10 cm.

In literature, several phase-shifting 3D imaging systems were proposed using the structure with one projector and one camera. However, we used a structure with two cameras and one projector. With this structure, we only need to calibrate the parameters of the two cameras, but do not need to calibrate the parameters of the projector (including geometric parameters and radiometric parameters). Specifically, in our system, gray-code [11] and phase-shift are combined to reconstruct a scene. The column-coding information of the gray-code method and the phase information of the phase-shift method are used to establish pixel correspondence between the left and right views. It does not require any geometric constraint imposed by the projector, and thus no projector calibration is required. Because camera calibration is relatively easier than projector calibration, our structure can be calibrated more easily. These calibrated camera parameters are used to rectify stereo images to obtain a horizontal epipolar geometry. The principle of the 3D imaging system is based on phase-shifting and stereo matching, which is similar to [23].

During measurement, the projector first projects k_1 cosine phase-shifted fringe patterns onto the surface of a target and the corresponding images are captured by these two synchronous cameras. The intensity of the i-th image with a phase shift δ_i is

$$I_i(x, y) = I'(x, y) + I''(x, y)\cos(\phi(x, y) + \delta_i), \tag{1}$$

Figure 1 (Color online) An illustration of the structured light system.

where I' denotes the average intensity, I'' is the intensity modulation, and ϕ represents the phase. For each camera, the relative phase map is calculated using these k_1 images [24]:

$$\phi(x,y) = \tan^{-1}\left(-\frac{a_2(x,y)}{a_1(x,y)}\right), \tag{2}$$

$$\begin{pmatrix} a_0(x,y) \\ a_1(x,y) \\ a_2(x,y) \end{pmatrix} = \boldsymbol{A}^{-1}(\delta_i)\boldsymbol{B}(x,y,\delta_i), \tag{3}$$

where

$$\boldsymbol{A}(\delta_i) = \begin{pmatrix} k_1 & \sum\cos(\delta_i) & \sum\sin(\delta_i) \\ \sum\cos(\delta_i) & \sum\cos^2(\delta_i) & \sum\cos(\delta_i)\sin(\delta_i) \\ \sum\sin(\delta_i) & \sum\cos(\delta_i)\sin(\delta_i) & \sum\sin^2(\delta_i) \end{pmatrix}, \tag{4}$$

$$\boldsymbol{B}(x,y,\delta_i) = \begin{pmatrix} \sum I_i \\ \sum I_i\cos(\delta_i) \\ \sum I_i\sin(\delta_i) \end{pmatrix}. \tag{5}$$

These relative phase maps cannot be directly used to recover the depth of the target. Instead, they will be used for stereo matching. Then, k_2 binary gray-code patterns [11] are projected. Corresponding images captured by these two cameras are used for phase unwrapping, that is, to determine the period number of the phases [23]. So far, we can obtain two absolute phase maps for the left and right views, respectively. Using the absolute phase of each pixel, it is easy to establish sub-pixel correspondences between the left and right views.

For a pixel p_L in the left view, its corresponding point in the right view should have a similar relative phase value. We determine its correspondence p_R by finding the pixel with minimum phase difference from p_L along the same line in the right view. To obtain the subpixel location of the corresponding point, we fit a three-order polynomial using five pixels centered at p_R using the linear least square method. The polynomial is defined as

$$x_R(\phi) = a_{0,R} + a_1\phi + a_2\phi^2 + a_3\phi^3. \tag{6}$$

Then, the subpixel location of p_R is determined as $x_R(\phi(x_L))$, as illustrated in Figure 2.

To improve the adaptiveness of our system to targets with multiple albedos, we use three different camera exposure times to scan each scene, which is similar to [10]. In this way, three disparity maps can

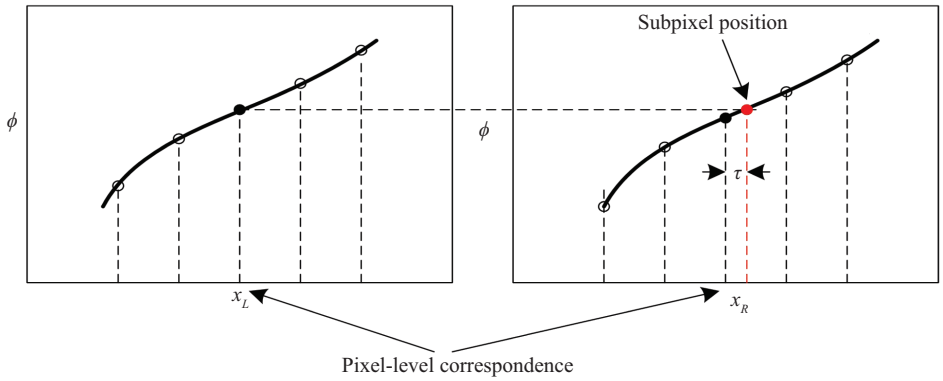

Figure 2 (Color online) Subpixel refinement.

Figure 3 (Color online) Samples in the InStereo2K dataset.

be obtained. To fuse these three disparity maps, we first calculate the average image of these phase-shift images in each scan. For each pixel, the disparity value that is closest to a give value g_f among these three average images at the selected pixel is considered as the fused result.

For the gray-code patterns, we set k_1=8. To achieve a trade-off between measurement accuracy and efficiency, we set the number of phase-shift patterns k_2 to 8. For each scene, we capture 49 images (8 phase-shifts and 8 gray-codes for 3 times, and 1 ambient light) for each camera. Among them, 48 images are used to calculate the disparity map. Left-right consistence check (LRC) and speckle filtering are used to remove outliers from disparity maps. Finally, the left and right disparity maps are saved together with their two corresponding RGB images under the ambient light.

Note that, Scharstein et al. [10] illuminated the scene from 4–18 projector positions to minimize the shadowed areas for view reconstruction. For each projector position, 120 images were captured under three different exposures. The system has to be re-calibrated when the projector position is changed. Although the disparity map obtained by [10] is dense, the scanning process is very time-consuming. Therefore, the method in [10] is unsuitable for the collection of large-scale stereo matching datasets. In contrast, our system can automatically scan a view in just one minute.

3.2 The InStereo2K dataset

Our dataset consists of 2050 pairs of RGB images with their highly accurate disparity maps. Within this dataset, 2000 pairs are used as the training set and 50 pairs are used as the test set. This dataset covers different indoor scenes including offices, classrooms, bedrooms, living rooms and dormitories. Figure 3 shows some samples of our dataset. Table 1 shows the comparison between our dataset and several existing datasets [1, 2, 10–13, 25–27]. Compared to KITTI 2012 and KITTI 2015, our dataset is one order of magnitude larger in the number of labelled images. In terms of disparity map quality, our disparity maps are much denser. The resolution of images in the Middlebury 2014 dataset is higher than that in our dataset. However, it only contains 23 pairs of training images. More importantly,

Table 1 A comparison between our InStereo2K dataset and several existing stereo datasets

Dataset	Synthetic/Natural	#Frames	Stereo	Depth	Resolution
Middlebury 2003 [11]	Natural	2	√	√	1800 × 1500
Middlebury 2005 [25]	Natural	9	√	√	∼1.5 MP
Middlebury 2006 [25]	Natural	21	√	√	∼1.5 MP
Middlebury 2014 [10]	Natural	23	√	√	∼6 MP
KITTI 2012 [1]	Natural	194	√	√	1242 × 375
KITTI 2015 [2]	Natural	200	√	√	1242 × 375
ETH3D [12]	Natural	27	√	√	∼0.3 MP
InStereo2K	Natural	2000	√	√	1080 × 860
Scene Flow [13]	Synthetic	35855	√	√	960 × 540
Sintel [26]	Synthetic	1064	√	√	1024 × 436
SYNTHIA [27]	Synthetic	∼200000	√	√	960 × 720

the images in Middlebury 2014 are acquired with high-end SLR cameras, while ours are collected with ordinary industrial cameras (which have higher noise levels). Our cameras are closer to those used in most practical applications. Although the resolution of our original RGB images is 1280 × 960, the RGB images and disparity maps are cropped to 1080 × 860 because stereo rectification introduces several invalid pixels. To improve the distribution of disparity values, we reduce the size of these RGB images by half and add them to the training set of our experiments. Note that, these half-resolution images are only used in our experiments but not included in the InStereo2K dataset.

4 Experiments and discussion

In this section, we show the role of our dataset in the training of deep stereo matching networks. To further improve stereo matching performance, we also test several network fine-tuning strategies.

4.1 Neural networks for test

We use two latest stereo matching networks in our experiments, including PSMNet [9] and StereoNet [8].

PSMNet. PSMNet [9] is one of the state-of-the-art methods in KITTI Stereo Evaluation 2015, which exploits global context information at the whole-image level. It consists of a spatial pyramid pooling (SPP) module [28] for the incorporation of global contexts and a stacked hourglass 3D CNN module for cost volume regularization. Specifically, three 2D convolutions are firstly cascaded and four residual blocks are followed to extract features. Then, the SPP module is applied to gather context information. The left and right feature maps are concatenated to build a cost volume, which is fed into a stacked hourglass 3D CNN for regularization. The stacked hourglass architecture has three mean hourglass networks, where each network produces a disparity map. Consequently, the stacked hourglass 3D CNN produces three disparity maps and losses. The final loss is the weighted average of these three losses.

StereoNet. StereoNet [8] extracts features from stereo images using a Siamese network. It builds a cost volume at a low resolution (i.e., 1/8 of the original resolution). To aggregate contexts across both the spatial domain and the disparity domain, the cost volume is filtered with four 3D convolutions. The final cost volume is used to compute a coarse disparity map with the soft argmin function. Finally, the coarse disparity maps are hierarchically refined to recover small details and thin structures. The hierarchical network uses the left color image as a guide, and applies six residual blocks (which use atrous convolutions) to refine the upsampled disparity map. Owing to its simplified architecture, StereoNet can run in real-time.

4.2 Training approach

We first pre-train these two networks using the synthetic Scene Flow dataset [13] to obtain initial models. Next, we fine-tune these initial models with our InStereo2K dataset to achieve better generalization

performance. Then, we use the training set of Middlebury 2014 as a test set to evaluate the generalization performance of different models. Note that, the Middlebury 2014 dataset is not used in any of our training phases. Therefore, the scenes in Middlebury 2014 are unseen for these networks.

4.3 Fine-tuning strategies

Here, we discuss several strategies for the fine-tuning of these networks. For each network, we have tested 6 different cases for network fine-tuning.

(a) Without fine-tuning (test with the pre-trained model).
(b) Fine-tuning using KITTI 2015 only.
(c) Fine-tuning using both Scene Flow and KITTI 2015.
(d) Fine-tuning using our InStereo2K dataset only.
(e) Fine-tuning using both Scene Flow and our InStereo2K dataset.
(f) Fine-tuning using both Scene Flow and our InStereo2K dataset (with augmentation in color).

Mayer et al. [29] showed that data augmentation (including augmentations in color and geometry) was very important for a synthetic dataset. In this paper, we also test the role of data augmentations in the real scene dataset (i.e., case (f)).

4.4 Implementation details

All experiments are implemented using TensorFlow on a Tesla K80 GPU.

For PSMNet, the initial model is trained using Adam [30] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The input images are randomly cropped to the size of 512×256. The batch size is set to 1. The learning rate is set to 0.001, and the training step is stopped at the 200k-th iteration. During the fine-tuning of the initial model, for cases (b) and (d), the learning rate is initially set to 0.001, and then reduced to 0.0001 at the 40k-th iteration. The fine-tuning step is stopped at the 60k-th iteration. For cases (c), (e) and (f), the learning rate is initially set to 0.001, and then reduced to 0.0001 at the 60k-th iteration. The fine-tuning step is stopped at the 90k-th iteration.

For StereoNet, during the training of the initial model on Scene Flow, we use RMSProp[2)] to optimize the model with a decay learning rate. The input images are randomly cropped to the size of 512×400. The batch size is set to 1. The learning rate is initially set to 0.001, and then reduced by a half at the 38k-th, 75k-th and 110k-th iterations. The training step is stopped at the 150k-th iteration. During the fine-tuning of the initial model, we use different parameters for different cases. For cases (b) and (d), the initial learning rate is set to 0.0005 and then reduced by a half at the 15k-th and 30k-th iterations. The fine-tuning step is stopped at the 50k-th iteration. For cases (c), (e) and (f), the initial learning rate is set to 0.0005 and then reduced by a half at the 15k-th, 30-th and 60k-th iterations. The fine-tuning step is stopped at the 90k-th iteration. In the following part, we use 'network name'-'training case' to represent the models of different network structures trained under different conditions. For example, PSMNet-A is used to represent the model of PSMNet trained under case (a).

4.5 Experimental results and analyses

In this subsection, bad 2.0 errors (percentage of pixels whose errors are larger than 2.0), average absolute errors in pixels and bad 4.0 errors are used as evaluation metrics to test the performance of these two networks under different cases. The quantitative results are listed in Tables 2–4.

It can be observed that, when these models are fine-tuned with the KITTI 2015 dataset (i.e., case (b)), the bad 2.0 error, bad 4.0 error, and the average absolute error of PSMNet can be reduced significantly as compared to the pre-trained model without fine-tuning (i.e., case (a)). However, the bad 2.0 error of StereoNet is not improved. When the models are fine-tuned with both the KITTI 2015 and Scene Flow datasets (i.e., case (c)), the performance of both models can further be improved. When these models are fine-tuned using our dataset only (i.e., case (d)), their performance is improved and outperforms case (c).

2) Hinton G, Srivastava N, Swersky K. Neural networks for machine learning. Lecture 6a. Overview of mini-batch gradient descent. https://www.cs.toronto.edu/tijmen/csc321/slides/lecture_slides_lec6.pdf.

Table 2 Evaluation on the Middlebury 2014 dataset (bad 2.0 error)

Model	Case (a)	Case (b)	Case (c)	Case (d)	Case (e)	Case (f)
StereoNet [8]	60.2	65.1	51.2	48.8	**40.8**	45.4
PSMNet [9]	52.2	30.3	28.8	24.8	**23.0**	**23.0**

Table 3 Evaluation on the Middlebury 2014 dataset (average absolute error in pixels)

Model	Case (a)	Case (b)	Case (c)	Case (d)	Case (e)	Case (f)
StereoNet [8]	22.5	20.4	16.5	**11.6**	12.7	14.4
PSMNet [9]	17.5	6.94	6.6	10.1	**3.94**	4.64

Table 4 Evaluation on the Middlebury 2014 dataset (bad 4.0 error)

Model	Case (a)	Case (b)	Case (c)	Case (d)	Case (e)	Case (f)
StereoNet [8]	42.0	49.6	34.6	32.3	**25.1**	30.3
PSMNet [9]	37.0	18.3	17.7	15.1	13.1	**12.6**

Figure 4 (Color online) Disparity maps achieved by StereoNet. The 1st row shows the color images in the training set of Middlebury 2014, the 2nd row shows the results of StereoNet-A, the 3rd row shows the results of StereoNet-C, and the last row shows the results of StereoNet-E.

Furthermore, when these models are fine-tuned using both our dataset and Scene Flow (i.e., case (e)), the error is further reduced. In addition, it can be observed that color enhancement cannot further improve the accuracy for both networks significantly (please see the results of case (f)). From Tables 2–4, we can find that the fine-tuning strategy in case (e) achieves the best performance.

The disparity maps obtained by these networks are shown in Figures 4 and 5. The results of these models trained only using the synthetic dataset have many artifacts (case (a); see the 2nd row in Figures 4 and 5). After fine-tuning with both the real scene dataset and the synthetic dataset, the results become more accurate (case (c); see the 3rd row in Figures 4 and 5). In addition, the models trained with both Scene Flow and our InStereo2K dataset (i.e., case (e)) achieve better accuracy than case (c). There are two reasons for this observation. First, our dataset has more training images than KITTI. Second, the images in our dataset are collected in indoor scenes, and the test dataset (i.e., Middlebury 2014 dataset) is also collected in indoor environment.

Finally, we upload the results of PSMNet-E on the test set of Middlebury 2014 for online evaluation, where the groundtruth is unknown. By August 18, 2019, the PSMNet's ranking (PSMNet_2000) is raised

Figure 5 (Color online) Disparity maps achieved by PSMNet. The 1st row shows the color images in the training set of Middlebury 2014, the 2nd row shows the results of PSMNet-A, the 3rd row shows the results of PSMNet-C, and the last row shows the results of PSMNet-E.

Figure 6 (Color online) Disparity maps achieved by PSMNet-E on the test set of Middlebury 2014. The 1st row shows the color images in the test set of Middlebury 2014, the 2nd row shows the results of PSMNet-E, and the 3rd row shows the results of PSMNet model in [9].

by 26 (from 108th to 82nd) in terms of bad 2.0 error in non-occlusion regions. When using the average absolute error in all regions as the evaluation metric, the ranking is raised by 12 (from 38th to 26th). Figure 6 shows some visual examples. PSMNet-E achieves higher prediction accuracy in texture-less regions and richer reconstruction details of objects than PSMNet.

5 Conclusion

In this paper, we introduce a new dataset for the training of stereo matching networks. It contains more than 2000 pairs of stereo images with highly accurate disparity maps. Experimental results show that our dataset can significantly improve the generalization performance of deep stereo matching networks on the Middlebury 2014 dataset. In addition, we observe that better generalization performance can be achieved using both real and synthetic datasets. Our dataset can be used to promote end-to-end stereo matching methods towards practical applications.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61402489, 61972435, 61972435, 61602499), Natural Science Foundation of Guangdong Province (Grant No. 2019A1515011271), Fundamental Research Funds for the Central Universities (Grant No. 18lgzd06), and Shenzhen Technology and Innovation Committee (Grant No. 201908073000399).

References

1 Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2012
2 Menze M, Geiger A. Object scene flow for autonomous vehicles. In: Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015
3 Li D, Liu N, Guo Y, et al. 3D object recognition and pose estimation for random bin-picking using partition viewpoint feature histograms. Pattern Recogn Lett, 2019, 128: 148–154
4 Khan S H, Guo Y, Hayat M, et al. Unsupervised primitive discovery for improved 3D generative modeling. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019. 9739–9748
5 Wang W, Gao W, Hu Z Y. Effectively modeling piecewise planar urban scenes based on structure priors and CNN. Sci China Inf Sci, 2019, 62: 029102
6 Yan T, Gan Y, Xia Z, et al. Segment-based disparity refinement with occlusion handling for stereo matching. IEEE Trans Image Process, 2019, 28: 3885–3897
7 Liang Z, Guo Y, Feng Y, et al. Stereo matching using multi-level cost volume and multi-scale feature constancy. IEEE Trans Pattern Anal Mach Intell, 2019. doi: 10.1109/TPAMI.2019.2928550
8 Khamis S, Fanello S, Rhemann C, et al. StereoNET: guided hierarchical refinement for real-time edge-aware depth prediction. In: Proceedings of the European Conference on Computer Vision (ECCV), 2018. 573–590
9 Chang J R, Chen Y S. Pyramid stereo matching network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018. 5410–5418
10 Scharstein D, Hirschmüller H, Kitajima Y, et al. High-resolution stereo datasets with subpixel-accurate ground truth. In: Prcoeedings of German Conference on Pattern Recognition. Berlin: Springer, 2014. 31–42
11 Scharstein D, Szeliski R. High-accuracy stereo depth maps using structured light. In: Prcoeedings of 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003
12 Schöps T, Schönberger J L, Galliani S, et al. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: Prcoeedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2017
13 Mayer N, Ilg E, Hausser P, et al. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016. 4040–4048
14 Scharstein D, Szeliski R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int J Comput Vision, 2002, 47: 7–42
15 Zbontar J, LeCun Y. Stereo matching by training a convolutional neural network to compare image patches. J Mach Learn Res, 2016, 17: 2
16 Mei X, Sun X, Zhou M, et al. On building an accurate stereo matching system on graphics hardware. In: Prcoeedings of IEEE International Conference on Computer Vision Workshops (ICCV Workshops), 2011. 467–474
17 Luo W, Schwing A G, Urtasun R. Efficient deep learning for stereo matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016. 5695–5703
18 Shaked A, Wolf L. Improved stereo matching with constant highway networks and reflective confidence learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017. 4641–4650
19 Kendall A, Martirosyan H, Dasgupta S, et al. End-to-end learning of geometry and context for deep stereo regression. In: Proceedings of the IEEE International Conference on Computer Vision, 2017. 66–75
20 Liang Z, Feng Y, Guo Y, et al. Learning for disparity estimation through feature constancy. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018. 2811–2820
21 Newell A, Yang K, Deng J. Stacked hourglass networks for human pose estimation. In: Prcoeedings of European Conference on Computer Vision. Berlin: Springer, 2016. 483–499
22 Zhang F, Prisacariu V, Yang R, et al. GA-Net: guided aggregation net for end-to-end stereo matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019
23 Lohry W, Chen V, Zhang S. Absolute three-dimensional shape measurement using coded fringe patterns without phase

unwrapping or projector calibration. Opt Express, 2014, 22: 1287–1301
24 Zhang S, Yau S T. Generic nonsinusoidal phase error correction for three-dimensional shape measurement using a digital video projector. Appl Opt, 2007, 46: 36–43
25 Scharstein D, Pal C. Learning conditional random fields for stereo. In: Prcoeedings of IEEE Conference on Computer Vision and Pattern Recognition, 2007. 1–8
26 Butler D J, Wulff J, Stanley G B, et al. A naturalistic open source movie for optical flow evaluation. In: Prcoeedings of European Conference on Computer Vision. Berlin: Springer, 2012. 611–625
27 Ros G, Sellart L, Materzynska J, et al. The synthia dataset: a large collection of synthetic images for semantic segmentation of urban scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016. 3234–3243
28 He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell, 2015, 37: 1904–1916
29 Mayer N, Ilg E, Fischer P, et al. What makes good synthetic training data for learning disparity and optical flow estimation? Int J Comput Vis, 2018, 126: 942–960
30 Kingma D P, Ba J. Adam: a method for stochastic optimization. 2014. ArXiv: 14126980