MSDC-Net: Multi-Scale Dense and Contextual Networks for Stereo Matching

Zhibo Rao*, Mingyi He*‡, Yuchao Dai*, Zhidong Zhu*, Bo Li*, Renjie He*†
* Northwestern Polytechnical University, Xian 710129, China
† Nanyang Technological University, 639798, Singapore
‡ Email address: myhe@nwpu.edu.cn (Mingyi He)

***Abstract*—Disparity prediction from stereo images is essential to computer vision applications such as autonomous driving, 3D model reconstruction, and object detection. To more accurately predict disparity map, a novel deep learning architecture (called MSDC-Net) for detecting the disparity map from a rectified pair of stereo images is proposed. Our MSDC-Net contains two modules: the multi-scale fusion 2D convolution module and the multi-scale residual 3D convolution module. The multi-scale fusion 2D convolution module exploits the potential multi-scale features, which extracts and fuses the different scale features by Dense-Net. The multi-scale residual 3D convolution module learns the different scale geometry context from the cost volume which aggregated by the multi-scale fusion 2D convolution module. Experimental results on Scene Flow and KITTI datasets demonstrate that our MSDC-Net significantly outperforms other approaches in the non-occluded region.**

I. Introduction

Disparity estimation aims at predicting the disparity d from a pair of stereo images, which is an essential intermediate component toward 3D scene reconstruction and understanding. For example, the disparity map information can benefit tasks such as autonomous driving for vehicles [1], [2], object detection and recognition [3], [4], and 3D model reconstruction [5]–[7].

In general, traditional stereo matching methods care more about how to accurately compute the matching cost and how to apply local or global information to refine the disparity map [8]–[10]. C. Rhemann *et al.* replaced the cost volume by considering cost aggregation methods as joint filtering. Their method proved that simple linear image filters such as a box or gaussian filter could even be used for cost aggregation [11]. K. Zhang *et al.* took advantage of cross-scale cost aggregation to optimize the cost volume. It showed cross-scale framework is useful and leads to significant improvements [12]. K. Zhang *et al.* employed an area-based local stereo matching algorithm for all image regions to evaluate disparity map, efficient approach that finding the matching points of given points within a predefined support window [13]. H. Simon *et al.* proposed a semi-global matching approach based on the coarse-to-fine (CTF) strategy to accelerate convergence and to avoid unexpected local minima [14]. The traditional methods tend to be highly explanatory and adaptable.

Recently, deep learning has made considerable achievements in understanding semantics from the raw data in matching corresponding points. Compared with the conventional methods, deep learning based methods are capable of making significant improvements in both precision and speed [15]–[18]. GC-Net employed the hierarchical 3D convolutions architecture to learn context from the cost volume which concatenated by each unary features [17]. SsSM-Net proposed a novel training loss to exploit the loop constraint in image warping and to handle the texture-less areas, leaving it can self-improve by adapting itself to new imageries [19]. SGM-Net utilized the penalties estimation method to control the smoothness and discontinuity of the disparity map [20]. PSM-Net [18] exploited global context information by spatial pyramid pooling (SPP) [21], [22] and dilated convolution architectures [23]. Mayer *et al.* introduced two end-to-end net-works for disparity estimation (Disp-Net) and optical flow (Flow-Net). They also created a large synthetic dataset called scene flow to improve the state-of-the-art [24]. S. Zagoruyko *et al.* trained a non-learned cost aggregation and regularization combined deep network to match 55 image patches. It shows that multiple neural network architectures specifically adapt to the stereo matching task [25]. The main idea of these methods is how to learn the context information from left-right images.

In this work, we propose a novel multi-scale dense and contextual networks (MSDC-Net) to exploit global context information in stereo matching effectively. We design a multi-scale fusion 2D convolution module to improve the global context understanding ability by extracting the cross-scale feature. Moreover, we redesign the 3D convolution module from the GC-Net and introduce the multi-scale residual 3D convolution, which improves the utilization of global context information. The experimental results prove that the proposed architecture outperforms GC-Net in learning global context.

Our main contributions can be summarized as:

- We propose an end-to-end learning framework for stereo matching without any post-processing.
- We design a multi-scale fusion 2D convolution module for incorporating global context information from images.
- We redesign a multi-scale residual 3D convolution to learn the regional support of context information.

II. Our Method

In this section, we present multi-scale dense and contextual network (MSDC-Net) in detail. The network architecture is illustrated in Fig. 1. Our model consists of four steps: multi-scale features extraction and fusion, cost volume construction, feature matching, and disparity map regression. First, the

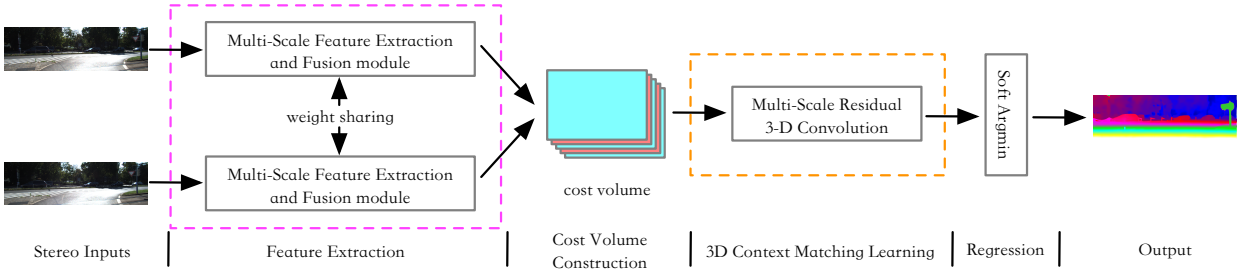

Fig. 1: **Our end-to-end deep stereo regression architecture, MSDC-Net** (Multi-Scale Dense and Context architecture).

multi-scale fusion 2D convolution module is applied to extract and fuse the multi-scale features as shown in Sec. II-A. Then, feature pairs are alternated to aggregate cost volume as shown in Sec. II-B. After that, the matching features are learned and the size of features volume is recovered by the multi-scale residual 3D convolution module as shown in Sec. II-C, and the disparity map is obtained by features volume regression as shown in Sec. II-D. Finally, we introduce the loss function of MSDC-Net as shown in Sec. II-E. The implementation detail is described in the following subsections, respectively.

A. Feature Extraction and Fusion

To get a robust descriptor which could represent the ambiguities in the photometric region and can incorporate local context, it is common to use a feature representation to capture local context [26]. We could use the deep feature representation to incorporate hierarchical context information by the Dense-Net [26]. In our model, we design a multi-scale fusion 2D convolution module through a series of 2D convolutional operations as shown in Fig. 3. The basic feature number is 32, and each convolutional layer is followed by a BN layer and a ReLU layer. The multi-scale fusion 2D convolution module contains two parts: different scale feature extraction and multi-scale features fusion.

Different scale feature extraction part is applied to extract the features with different size from image pairs. This part owns the 51 convolution layers with the different convolution filters. To reduce the calculation, we adopt the 5×5 convolutional filter with the stride of two to subsample the image pairs. Following this layer, we apply the dense block consisting of 16 convolutional layers with 3×3 convolutional filters and direct connections between four convolutional layers. Thus, we could obtain the feature size about $1/2H \times 1/2W$. In the same way, we could get the feature size about $1/4H \times 1/4W$ and $1/8H \times 1/8W$, and padding the feature size to $1/2H \times 1/2W$. Then we concatenate the different size features to obtain the aggregating features volume.

Multi-scale feature fusion part fuses the aggregated features volume to form a cost volume. It has 18 convolution layers with the different filters. To avoid losing the critical information and fusing the aggregating features volume, we adopt 128 convolutional filters with the size 5×5 and the stride of two to subsample the aggregating features volume. Then the dense block fuses these features. We obtain the unary features by passing stereo images with the same parameter.

B. Cost Volume Construction

Similar to the conventional stereo matching algorithms, we construct a four-dimensional cost volume ($H \times W \times D \times F$) by concatenating the fusion features at each disparity level as shown in Fig. 2. Specifically, the stereo matching cost is computed using the deep unary features of stereo image pairs to preserve prior knowledge of stereo vision.

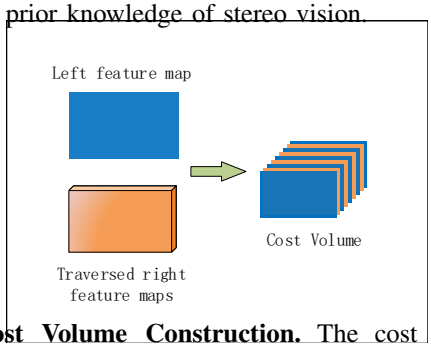

Fig. 2: **Cost Volume Construction.** The cost volume is constructed via concatenating the fusion features. The blue rectangle represents a fusion feature map of the left image. the orange cube represents the set of traversed fusion feature maps of the right image from 0 to the disparity range $D/4$.

C. Feature Matching

Given the fusion feature assembled cost volume, we learn the matching cost at each candidate disparity from the different size unary feature and the regularization from the local context. The encoder-decoder networks often cost a vast amount of calculation and very difficult to train. Thus, we redesign the 3D convolution module which could better learn the context of tiny objects and improve the efficiency of the learning process. To exploit the relationship between disparity, height, and width of image pairs, we present the multi-scale residual 3D convolution module, which contains two parts: multi-scale residual feature matching and scale recovery as shown in Fig. 4. In the module, the basic feature number is 32.

The multi-scale feature matching part is applied to match the geometry features from cost volume. This part has four levels, and there are residual 3D convolution layers between each subsampling. When subsampling, the features number will become double. When up-sampling, the features number will decrease one time. Moreover, we pass the features information

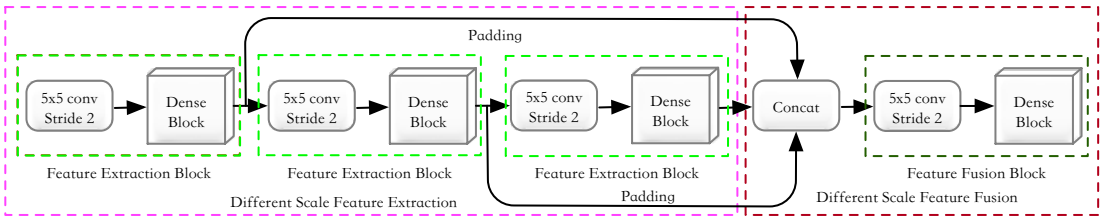

Fig. 3: **The multi-scale fusion 2D convolution module.** Different scale feature extraction part uses the dense block to extract the features with different size. Multi-scale features fusion part fuses the multi-scale features to form a cost volume.

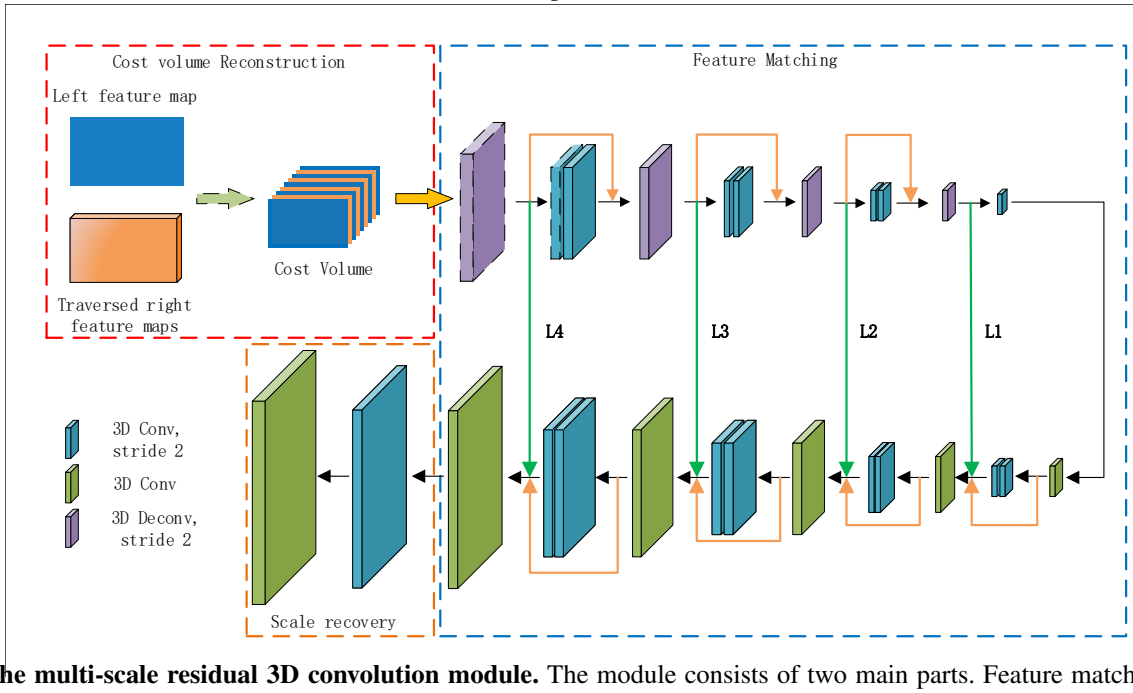

Fig. 4: **The multi-scale residual 3D convolution module.** The module consists of two main parts. Feature matching part uses the hierarchical 3D convolution architecture as a basic frame, then adopts residual mode as shown by green arrows to pass the features toward the same level 3D convolution layer. Scale recovery uses two deconvolution operations to recover scale.

between the same level 3D convolution layer to avoid losing the critical information. After this part, we obtain the matching feature but in a low resolution $1/4H \times 1/4W \times 1/4D$.

The scale recovery part is applied to recover the size of the input image. In this part, we adopt two deconvolution operations to recovery scale. The output of the scale recovery part is a final feature volume with size $H \times W \times D$.

D. Disparity Map Regression

Compared with the classification-based matching method, the regression is more robust and effective. First, we get the probability of each disparity value d which can be calculated from a final feature volume c via the softmax operation $\sigma(\cdot)$. Second, we obtain the predicted disparity $\hat{d}$ which can be calculated as the sum of each disparity d weighted, as:

$$\hat{d} = \sum_{d=0}^{D_{\max}} d \times \sigma(-c_d) \tag{1}$$

where c represents the final cost volume with size $H \times W \times D$, and $\sigma(\cdot)$ represents the softmax operation.

E. Loss Function

The overall consideration based on the stereo matching research, we think the absolute error value of predicted result and ground-truth should be used in the different Loss function. Compared to L_2 loss function, the L_1 loss is widely used in object detection because of its robustness and low sensitivity to outliers. To fit in with the stereo matching task, the L_1 loss function is redefined as:

$$L\left(d, \hat{d}\right) = \frac{1}{N} \sum_{i=1}^{N} \mathrm{Smooth}_{\mathrm{L}_1}\left(d_i - \hat{d}_i\right) \tag{2}$$

in which

$$\mathrm{Smooth}_{\mathrm{L}_1}(x) = \begin{cases} \frac{1}{3}x^2, & \text{if } |x| < 3. \\ |x|, & \text{otherwise.} \end{cases} \tag{3}$$

where N is the total number of labeled pixels where the value is not 0, d is the ground-true disparity, and $\hat{d}_i$ is the predicted disparity. In the loss function, we set three as the critical point. Because we hope when the predicted pixels value less than three, the influence of the points more less impact on the network, and three as the critical point is the error which could be accepted.

TABLE I: **The compared experiment of different model variants on the synthetic Scene Flow dataset [24].**

Model Type	*Error Rate* (%)			*Error*		Param.	Time (ms)
	> 1 px	> 3 px	> 5 px	MAE (px)	RMS (px)		
Single scale 2D and 3D conv (replace conv layers 1-97)	28.7	18.2	16.4	7.34	24.8	4.6M	0.76
Single scale 2D conv (replace 2D conv layers 1-68) w Multi-Scale Residual 3D conv	14.9	9.5	8.1	3.6	17.9	4.6M	0.62
Single scale 3D conv (replace 3D conv layers 69-97) w Multi-Scale Fusion 2D conv	15.8	9.2	7.4	3.8	16.2	4.6M	0.74
MSDC-Net	11.6	8.7	6.4	1.6	11.3	4.6M	0.75

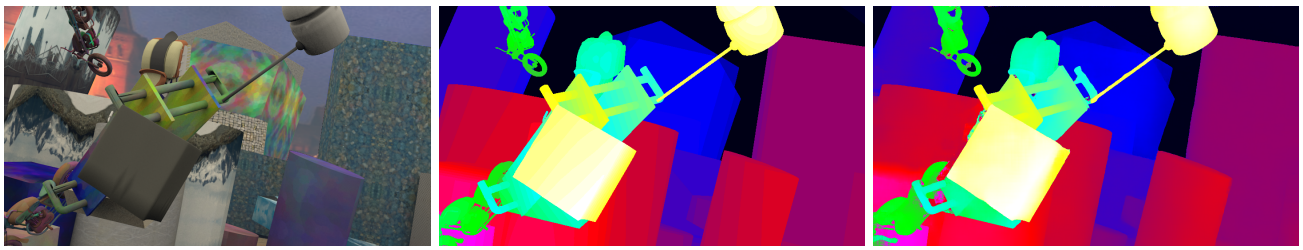

Fig. 5: **Sceneflow test data qualitative results.** From left: left stereo input image, ground-truth, disparity prediction.

III. Experiments

In this section, the performance of the proposed method is evaluated on two widely used stereo datasets: Scene Flow [24] and KITTI [27]. We firstly show our experiment details on the training process. After that, we discuss the different parameters of our model and justify a series of our design models in the various parameters. Finally, we compare the performance of our method with the state-of-art on the KITTI stereo dataset.

A. Experimental parameters

The proposed MSDC-Net is implemented using Tensorflow. All models were end-to-end trained by Adam Optimizer with a constant learning rate of 1×10^{-3} and $\beta_1 = 0.9, \beta_2 = 0.99$. Color normalization is performed on each image to ensure the pixel intensities ranged from 0 to 1. To increase the sample, the input images are randomly cropped to size 256×512 from a pair of normalized stereo images. The maximum disparity is set to $D = 192$. We trained our model on four Nvidia 1080Ti GPUs with the batch size of 8. The training process took 50 epochs for Scene Flow and 1000 epochs for KITTI.

B. Model Design Analysis

To verify the effectiveness of our design, we present an ablation study to compare a series of different model variants. We apply the Scene Flow dataset [24] for the experiments, which contains $35,454$ training and $4,370$ testing images with 540×960. As shown in [17] and [18], the large dataset used to train the model without over-fitting, it could help to evaluate the model correctly. Moreover, the Scene Flow dataset has dense ground truth and removes any discrepancies which caused by wrong labels. To evaluate different model variants, we first train each model for 50 epochs to obtain the models, then verify the models from the test images, the result as shown in Tab. I and Fig. 5.

As shown in Tab. I and Fig. 5, the multi-scale fusion 2D convolution and multi-scale residual 3D convolution architectures perform excellent performance and significantly outperform single scale 2D and 3D convolution architectures. Compared with the stereo matching method based on 2D convolution architectures, the 3D convolution architectures need more computational cost both in the training process and prediction process. However, 3D convolution architectures largely promote the performance at the same time. Moreover, we proposed the multi-scale residual 3D convolution architectures are easy to train and convergence.

C. Experiment with KITTI

To evaluate the performance of our model, we fine-tune the model which pre-trained on Scene Flow for a further 1000 epochs on KITTI 2012 and 2015 respectively. The KITTI 2012 and 2015 are real-world datasets with challenging and varied road scene, which contain 194 training stereo image pairs in the KITTI 2012 and 200 training stereo image pairs in the KITTI 2015 with sparse ground-truth disparities. Moreover, the datasets prepare another 194 testing image pairs in the KITTI 2012 and 200 testing image pairs in the KITTI 2015 without ground-truth disparities. To prevent our model form over-fitting on the KITTI, we divided the whole training data into a training set 80% and a validation set 20%. We show the representative results of our method in Fig. 6. In addition, we evaluate the performance of our model with the state-of-art methods on the KITTI server in Tab. II and Tab. III.

As shown in Fig. 6, our method could predict dense and clean disparity maps. MSDC-Net benefited from the Dense block has strong features extraction ability, compared to other methods. Thus, MSDC-Net could obtain more robust results, even in ill-posed regions. Our approach outperforms previous deep learning methods, which produce noisy and inaccuracy disparity maps. For this reason, these algorithms do not use multi-scale feature extraction and fusion architecture. Moreover, the layers of these algorithms are often more shallow but have more parameters; it maybe limits the performance in the matching task.

As shown in Tab. II and Tab. III, Our model is better than GC-Net, SGM-Net, etc., which were reported by the KITTI evaluation server. Our method not only achieves state of the

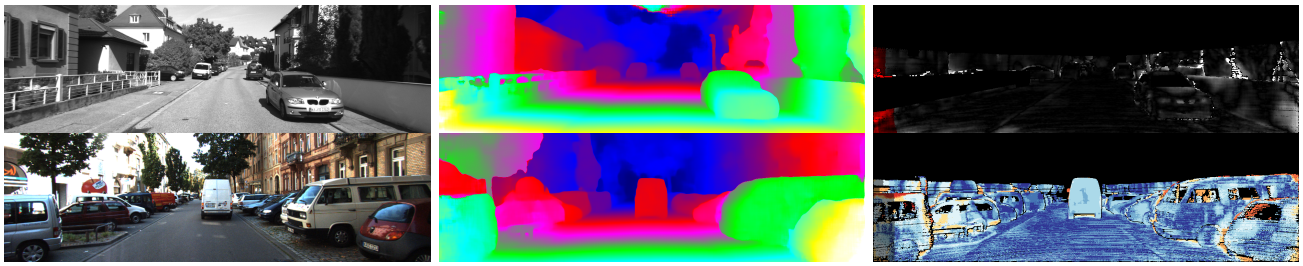

Fig. 6: **KITTI 2012 and 2015 test data qualitative results.** From left: left stereo input image, disparity prediction, error map.

TABLE II: **Results on KITTI 2012 stereo benchmark.** (different pixels threshold, as of 29 June 2019)

Method	*2 pixels* (%)		*3 pixels* (%)		*4 pixels* (%)		*5 pixels* (%)			
	Out-Noc	**Out-All**	**Out-Noc**	**Out-All**	**Out-Noc**	**Out-All**	**Out-Noc**	**Out-All**	**Avg-Noc**	**Avg-All**
L-ResMatch [15]	3.64 %	5.06 %	2.27 %	3.40 %	1.76 %	2.67 %	1.50 %	2.26 %	0.7 px	1.0 px
MC-CNN-acrt [16]	3.90 %	5.45 %	2.37 %	3.63 %	1.90 %	2.85 %	1.64 %	2.39 %	0.7 px	0.9 px
GC-NET [17]	**2.71 %**	**3.46 %**	1.77 %	2.30 %	1.36 %	1.77 %	1.12 %	1.46 %	0.6 px	0.7 px
SsSMnet [19]	3.34 %	4.24 %	2.30 %	3.00 %	1.82 %	2.39 %	1.53 %	2.01 %	0.7 px	0.8 px
SGM-Net [20]	3.60 %	5.15 %	2.29 %	3.50 %	1.82 %	2.39 %	1.60 %	2.36 %	0.7 px	0.9 px
MSDC-Net	2.78 %	3.47 %	**1.64 %**	**2.12 %**	**1.22 %**	**1.58 %**	**0.98 %**	**1.26 %**	**0.5 px**	**0.6 px**

TABLE III: **Results on KITTI 2015 stereo benchmark.** (as of 29 June 2019)

Method	*All pixels* (%)			*Non-Occluded pixels* (%)			
	D1-bg	**D1-fg**	**D1-all**	**D1-bg**	**D1-fg**	**D1-all**	**Runtime**
L-ResMatch [15]	2.72 %	6.95 %	3.42 %	2.35 %	5.76 %	2.91 %	48s
MC-CNN-acrt [16]	2.89 %	8.88 %	3.89 %	2.48 %	7.64 %	3.33 %	67s
GC-NET [17]	2.21 %	6.16 %	2.87 %	2.02 %	5.58 %	2.61 %	0.9s
PSMNet [18]	**1.86 %**	4.62 %	2.32 %	**1.71 %**	4.31 %	2.14 %	**0.41s**
SsSMnet [19]	2.70 %	6.92 %	3.40 %	2.46 %	6.13 %	3.06 %	0.8s
SGM-Net [20]	2.66 %	8.64 %	3.66 %	2.23 %	7.44 %	3.09 %	67s
MSDC-Net	1.96 %	**3.77 %**	**2.26 %**	1.83 %	**3.57 %**	**2.12 %**	0.7s

art results for both KITTI 2012 and 2015 benchmarks but also a little better than most competing approaches in the non-occluded region. Compared to other methods, our architecture more explicitly leverages different scale geometry by 2D and 3D modules, resulting in an improvement in performance.

IV. Conclusions

In this work, we propose a highly efficient network architecture for stereo matching. The proposed framework consists of two main modules: the multi-scale fusion 2D convolution module and the multi-scale residual 3D convolution module. The multi-scale fusion 2D convolution module incorporates different levels of feature maps to form a cost volume. The multi-scale residual 3D convolution module further learns to regularize the cost volume via repeated top-down/bottom-up processes. The proposed method has been verified through the different experiments. Experimental results show that the proposed method could predict a dense, clean and precise disparity map from image pairs. For future work, we are interested in exploring the potential of generative adversarial networks and explicit semantics to improve our disparity map prediction in the visual occlusion region.

V. Acknowledgment

This work was supported in part by Natural Science Foundation of China (61420106007, 61671387 and 61871325).

References

[1] A. Seki and M. Okutomi, "Robust obstacle detection in general road environment based on road extraction and pose estimation," in *IEEE Intelligent Vehicles Symposium*, 2006, pp. 437–444. 1

[2] H. Oleynikova, D. Honegger, and M. Pollefeys, "Reactive avoidance using embedded stereo vision for MAV flight," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 50–56. 1

[3] X. Chen, K. Kundu, Y. Zhu, A. Berneshawi, H. Ma, S. Fidler, and R. Urtasun, "3d object proposals for accurate object class detection," in *International Conference on Neural Information Processing Systems (NIPS)*, 2015, pp. 424–432. 1

[4] C. Zhang, Z. Li, Y. Cheng, R. Cai, H. Chao, and Y. Rui, "Meshstereo: A global stereo model with mesh alignment regularization for view interpolation," in *IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2057–2065. 1

[5] A. Seki, O. J. Woodford, S. Ito, B. Stenger, M. Hatakeyama, and J. Shimamura, "Reconstructing fukushima: A case study," in *International Conference on 3D Vision (3DV)*, 2014, pp. 681–688. 1

[6] B. Li, C. Shen, Y. Dai, A. V. D. Hengel, and M. He, "Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1119–1127. 1

[7] B. Li, Y. Dai, and M. He, "Monocular depth estimation with hierarchical fusion of dilated cnns and soft-weighted-sum inference," in *Pattern Recognition*, 2018. 1

[8] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2008. 1

[9] Q. Yang, "A non-local cost aggregation method for stereo matching," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 1402–1409. 1

[10] O. Woodford, P. Torr, I. Reid, and A. Fitzgibbon, "Global stereo reconstruction under second-order smoothness priors," *IEEE Transactions on*

Pattern Analysis and Machine Intelligence, vol. 31, no. 12, pp. 2115–28, 2009. 1

[11] A. Hosni, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 2, pp. 504–511, 2013. 1

[12] K. Zhang, Y. Fang, D. Min, L. Sun, S. Yan, S. Yang, and Q. Tian, "Cross-Scale Cost Aggregation for Stereo Matching," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 5, pp. 965–976, 2014. 1

[13] K. Zhang, J. Lu, and G. Lafruit, "Cross-based local stereo matching using orthogonal integral images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 1073–1079, 2009. 1

[14] S. Hermann and R. Klette, "Evaluation of a new coarse-to-fine strategy for fast semi-global stereo matching," *Lecture Notes in Computer Science*, vol. 7087, pp. 395–406, 2011. 1

[15] A. Shaked and L. Wolf, "Improved Stereo Matching with Constant Highway Networks and Reflective Confidence Learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6901–6910. 1, 5

[16] J. Žbontar and Y. Le Cun, "Computing the stereo matching cost with a convolutional neural network," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1592–1599. 1, 5

[17] A. Kendall, H. Martirosyan, S. Dasgupta, and P. Henry, "End-to-End Learning of Geometry and Context for Deep Stereo Regression," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 66–75. 1, 4, 5

[18] J.-R. Chang and Y.-S. Chen, "Pyramid Stereo Matching Network," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 5410–5418. 1, 4, 5

[19] Yiran Zhong, Yuchao Dai, and Hongdong Li, "Self-Supervised Learning for Stereo Matching with Self-Improving Ability," in *arXiv preprint*, 2017. 1, 5

[20] A. Seki and M. Pollefeys, "SGM-Nets: Semi-Global Matching with Neural Networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6640–6649. 1, 5

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015. 1

[22] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6230–6239. 1

[23] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2018. 1

[24] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4040–4048. 1, 4

[25] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4353–4361. 1

[26] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2261–2269. 2

[27] M. Moritz, H. Christian, and G. Andreas, "Object scene flow," *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)*, vol. 140, pp. 60–76, 2018. 4