Deep Neural Network for Handcrafted Cost-based Multi-view Stereo

Yoonbae Jeon[1], In Kyu Park[1]
[1]Inha University, Department of Electrical and Computer Engineering
Incheon 22212, Korea

ABSTRACT

In the last decades, depth estimation from multi-view has been treated as an ill-posed problem. This problem becomes severe with limited data, such as sparse-view cases. However, with the availability of convolutional neural network (CNN), recent learning-based depth estimation methods have become effective on occluded and texture-less areas, whereas prior works still suffer when handling such issues. They utilize features from the CNN layer to construct cost volume and regress the input volume with a regression network. To overcome those concerns, we introduce a unique approach by combining hand-crafted and learning-based strategies. Specifically, we utilize the normalized cross-correlation (NCC) cost volume, which is more robust to noise than simple L1 and L2 costs, to improve the photo-consistency between local patches. The entire construction pipeline is implemented by PyOpenCL to speed up the processing time. Finally, we employ the network that estimates depth by regressing the handcrafted cost-based plane sweeping volume.

Keywords: multi-view stereo, plane-sweeping stereo, depth estimation, neural network, GPGPU

1. INTRODUCTION

The main purpose of multi-view stereo (MVS) algorithm is to estimate precise depth information by utilizing camera pose information and multiple input images acquired at an arbitrary angle of view. The depth image obtained through MVS can be used to reconstruct a point cloud, voxel, or mesh into three dimensions. Recently, these works have been applied in the autonomous vehicle industry and AR/VR, which are major fields of the Fourth Industrial Revolution, high-accuracy depth images, and 3D information, which require high-quality service. Unfortunately, previous MVS techniques have limitations when the current circumstances are given. They can only operate well when the occlusion between adjacent images is small, and they entail massive computational complexity with a long processing time.

Recently, many researchers have emerged in the field of computer vision to solve various problems by using deep neural networks with a large number of data sets. Following this trend, recent approaches have attempted to solve the MVS problem by using deep neural networks based on convolutional neural network (CNN)[1-5]. MVSNet[1] is the representative CNN-based MVS method, which reconstructs dense depth images and 3D point clouds given multiple input images and camera position information. DeepMVS[3] estimates depth information from the cost volume on the basis of the patch-matching score, which is mainly used in the existing MVS algorithm. However, MVSNet still has a limitation as it is unable to reconstruct 3D and predict elaborate depth images when the number of input images is small. Meanwhile, the drawback of DeepMVS is its high dependence on external library utilization and a long inference time.

In this paper, with the input cost volume constructed by plane-sweeping stereo technique[7,8] from a small number of images, we propose a method to obtain a dense depth image by using a neural network encoder–decoder structure based on CNN proposed by MVDepthNet[2]. The proposed method is divided into an input cost volume building module and a depth information estimation process through a deep neural network. By utilizing a Python-based GPGPU framework with PyTorch, we train the whole network in an end-to-end fashion. Figure 1 shows the overall flowchart of the proposed method. In the experiment result section, we show that our network can estimate dense depth even with a small number of source images. We summarize our contributions as follows:

- We utilize an OpenCL–based GPGPU algorithm to implement the hand-crafted cost volume construction process, which is not supported by the existing deep learning framework.
- Our proposed method is the first of its kind, combining hand-crafted cost volume with a learned regression network to produce dense depth estimations.

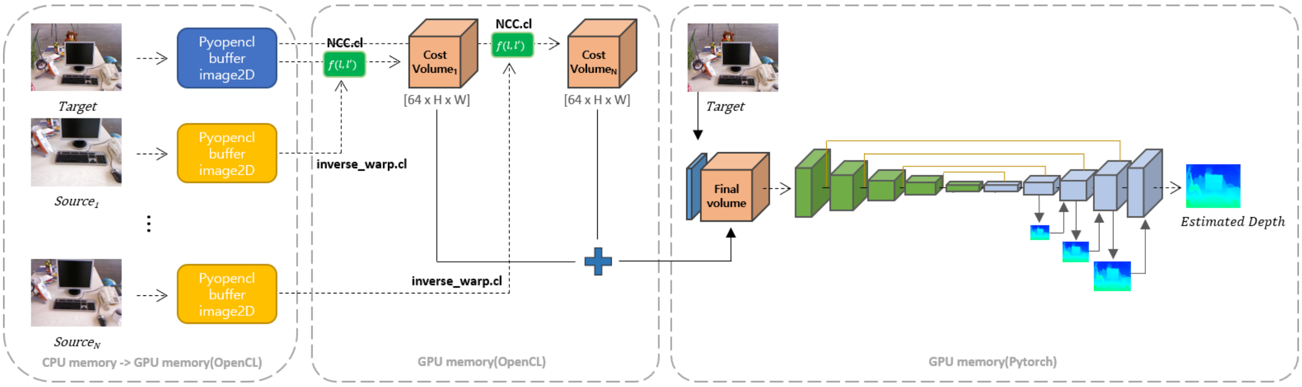

Figure 1. Overall pipeline of the proposed method.

2. PROPOSED METHOD

In this paper, we propose an encoder–decoder structure network that regresses hand-crafted cost volume constructed with the NCC cost function. No deep learning framework that supports window–based NCC cost calculation exists; thus, we propose to utilize OpenCL to accelerate cost volume construction. We utilize OpenCL command queue features that enable the entire system to train the network with multiple CPU cores.

2.1 Cost volume construction

The input of the network is cost volume which is constructed by target view and source views. First, we construct plane-sweeping volume by concatenating inverse-warped source images. With given camera extrinsic and intrinsic parameters, homography matrix H at each depth level d_i for warping the source image to the target coordinates is defined as

$$H = K\left(R_{r,t} + t_{r,t}\left(0\ 0\ \frac{1}{d_i}\right)\right)K^{-1} \tag{1}$$

where d_i means the i^{th} unit depth divided by 64 steps from 0.5m to 50m, K is the intrinsic matrix, and $R_{r,t}$, $t_{r,t}$ denote the relative pose matrix between the target image and the source image. After the homography matrix H for each depth d_i is obtained, the source images are converted to OpenCL Image2D memory and warped at target coordinates by the inverse warp module to construct plane-sweep volume. Each input cost volume layer consists of the NCC cost between the target view and each plane-sweep volume layer. To reduce the effects of occlusion and noise between the target view and the inverse-warped source views, the final cost volume is generated by averaging the cost volume. To speed up the process of constructing cost volume, which requires high computational complexity, we utilize the GPGPU parallel algorithm. We implement both the inverse warp module and cost calculation module with the OpenCL framework from scratch. We adopt PyOpenCL to ensure compatibility with the Python-based deep learning framework. Our implementation of the inverse-warp module is 6.3 times faster than the PyTorch grid_sample function, which operates the same features.

2.2 Cost volume regression network

In this paper, we utilize a CNN-based encoder-decoder network to regress the cost volume. Each encoder and decoder network consists of five-level convolutional layers. Specifically, they are constructed with filter sizes of 7 × 7 and 5 × 5 for the first and second layers, respectively, and 3 × 3 for the other layers. As shown in Figure 1, the encoder layer takes the input volume, including the target image concatenated by the cost volume of size $D_L \times H \times W$. High-level features extracted from each encoder layer are propagated to the decoder layer through skip connection. Except for the first decoder layer, each decoder layer estimates an inverse depth map with the propagated high-level feature and depth information predicted from the previous stage. In the prediction layer, a 3 × 3 convolutional layer is used to regress the channel of the output features followed by sigmoid activation for inverse depth estimation. The final output size of the inverse depth map is $H \times W$.

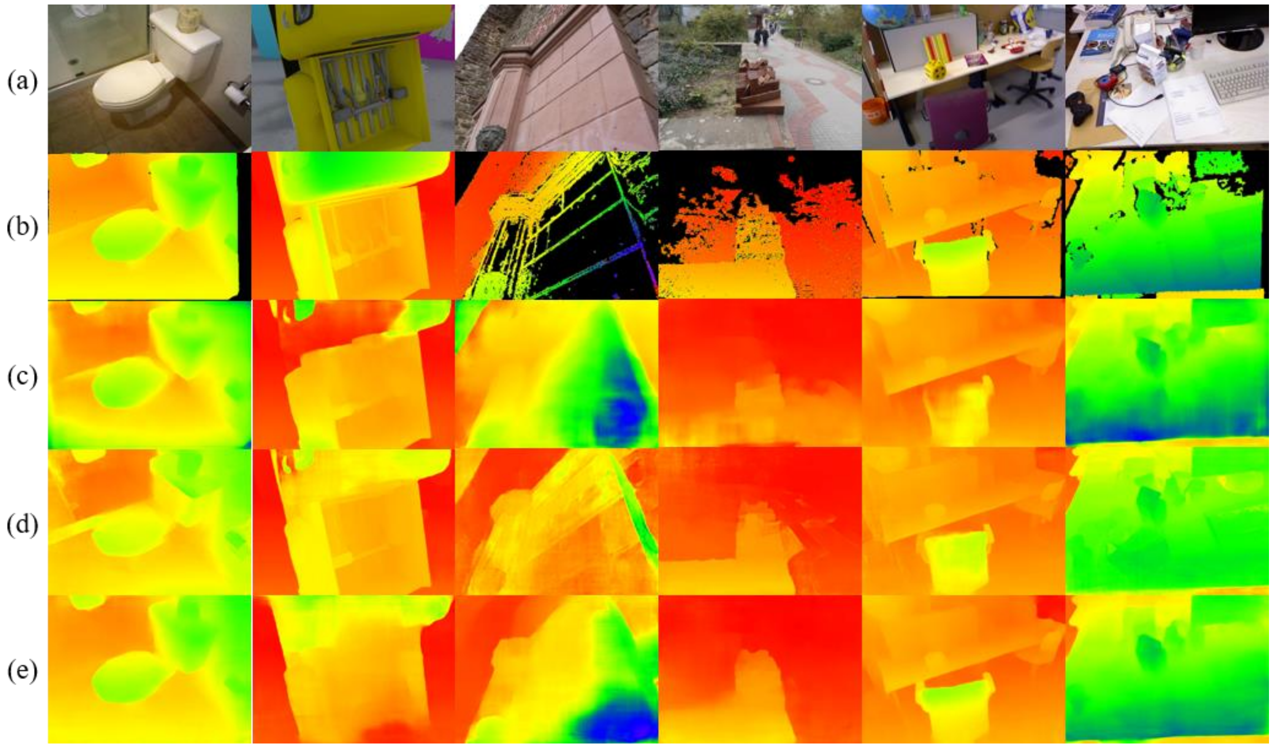

Figure 2. Qualitative results of DeMoN dataset between the state-of-the-art MVS networks. (a), (b) are input images and ground-truths. (c), (d), (e) are the results of MVDepthNet, DPSNet and our approach, respectively. Note that two to five source images are used to construct the cost volume in each network.

3. EXPERIMENTAL RESULTS

We used the DeMoN[4] dataset to train and evaluate our network. The DeMoN dataset is composed of SUN3D[9], TUM-RGBD[10], which are acquired indoors with an RGBD camera, MVE[11] acquired from outdoors, and SCENES11[3,4] synthesized with the ShapeNet[12] dataset. Each dataset includes multiple images acquired at arbitrary viewpoints with ground-truth depth, and camera parameters. For qualified cost volume construction, we select the target image only if the paired ground-truth depth map has more than 70% of the depth range to be measured and source images that have enough view angle difference and baseline length compared with the target view. Our network is implemented on PyTorch and trained with the Adam optimizer set with default parameters. We set the batch size as 12 with a learning rate of 1e-4 and train the network with 40,000 iterations. All images are scaled to 480 × 352 to reduce the training time. For the robust depth estimation, our network is trained with the sum of L1 loss between the downscaled ground-truth depth and the predicted depth at each decoder layer, defined as:

$$L = \sum_{k=0}^{3} \frac{\lambda_k}{n} \sum | \frac{1}{d_k} - \delta_k | \qquad (2)$$

where d_k and δ_k are the mean scaled ground-truth depth and inverse depth predicted from the k^{th} decoder layer, respectively, and λ_k represents the weight of the k^{th} step, which is fixed at 0.25.

In this work, we compare our method with two state-of-the-art MVS networks, namely, MVDepthNet and DPSNet. Figure 2 shows the qualitative comparison results of the depth map predicted with the DeMoN dataset, and the quantitative comparison with the 7SCENES[13] *office* test datasets (*office-02, office-06, office-07, office-09*) is shown in Table 1. We utilize the metrics of absolute difference error (Abs diff), square relative error (Sq Rel), root mean squared error (RMSE), and inlier ratios (Acc) for quantitative comparison. For qualitative results, we select the target image among the validation set given by DPSNet and source images following the upper view-selection rule. Compared with other methods, our network can predict more precise depth maps on various datasets, as indicated by qualitative and quantitative evaluations.

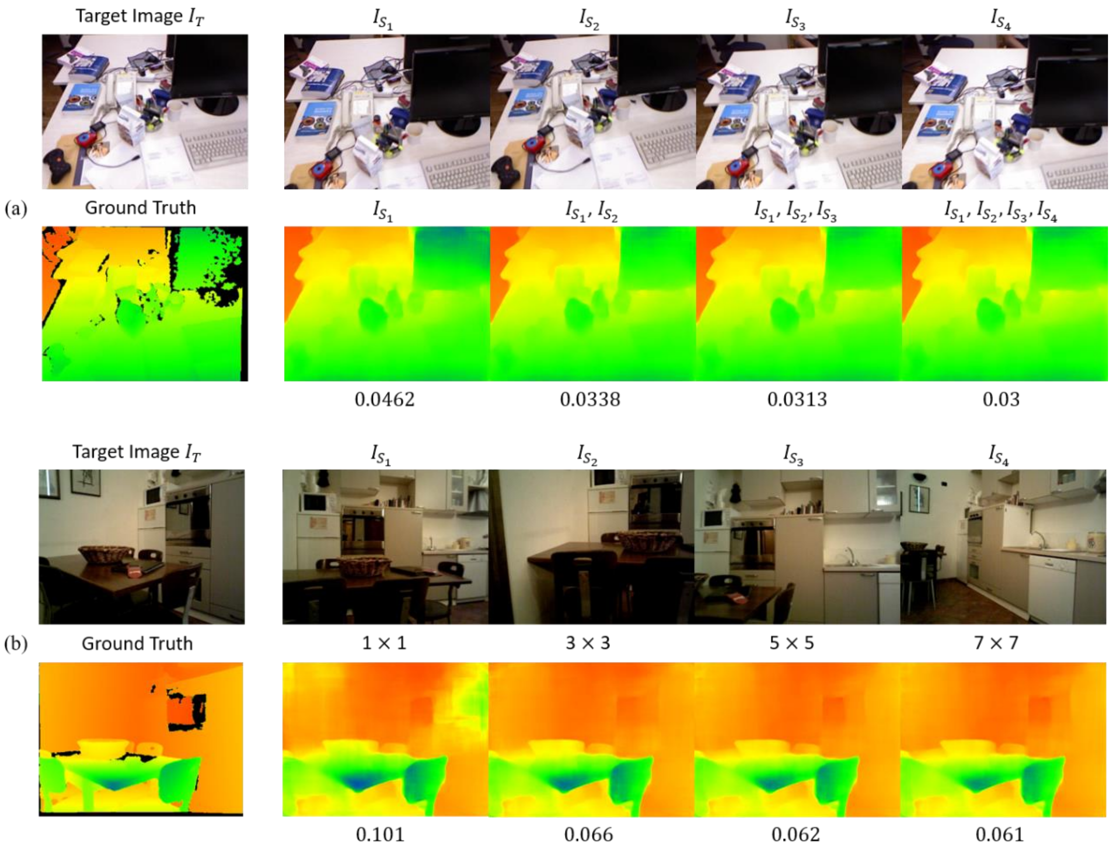

Figure 3. Ablation study: performance comparison w.r.t different number of source images and different window size of NCC cost. The number below the depth map represents the absolute difference error.

We also conduct ablation studies by varying the number of source frames and the window size of NCC cost. As shown in Figure 3, a larger number of source images and window size yields more precise depth maps because any lack of information caused by occlusion decreases.

Table 1. Comparison results between MVDepthNet, DPSNet, and our method.

Methods	**Abs diff**	**Sq Rel**	**RMSE**	**Acc** ($\delta < 1.25$)
MVDepthNet	0.1131	0.0523	0.1442	0.6734
DPSNet	0.1104	0.0451	0.1366	0.6609
Ours	**0.0940**	**0.0311**	**0.1243**	**0.6758**

4. CONCLUSION

In this paper, we propose a U-Net shaped cost volume regression network that takes input volume constructed by using the hand-crafted method, which utilizes the GPGPU algorithm. Our method is able to exploit the importance of the cost volume context, which was generally wasted by previous MVS works. Moreover, our implementation delivers an advanced approach that trains the whole network in adequate batch sizes in an end-to-end fashion. Our experimental results prove that our approach, which achieved better qualitative and quantitative performances compared with other methods, is superior.

ACKNOWLEDGEMENT

This work was supported by Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government (MSIT) (2017-0-00142, Development of Acceleration SW Platform Technology for On-device Intelligent Information Processing in Smart Devices and 2020-0-01389, Artificial Intelligence Convergence Research Center (Inha University)).

REFERENCES

[1] Y. Yao, et al., “MVSNet: Depth inference for unstructured multi-view stereo,” in *Proc. of European Conference on Computer Vision*, 767-783 (2018).

[2] K. Wang and S. Shen, “MVDepthNet: Real-time multiview depth estimation neural network,” in *Proc. of International Conference on 3D Vision*, 248-257 (2018).

[3] P.-H. Huang, et al., “DeepMVS: Learning multi-view stereopsis,” in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2821-2830 (2018).

[4] B. Ummenhofer, et al., “DeMoN: Depth and motion network for learning monocular stereo,” in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 5038-5047 (2017).

[5] S. Im, H.-G. Jeon, S. Lin, and I. S. Kweon, “DPSNet: End-to-end deep plane sweep stereo,” in *Proc. of International Conference on Learning Representations*, 2019.

[6] H. Hirschmuller and D. Scharstein, “Evaluation of stereo matching costs on images with radiometric differences,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 31(9), 1582-1599 (2008).

[7] R. T. Collins, “A space-sweep approach to true multi-image matching,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 358-363 (1996).

[8] D. Gallup, et al., “Real-time plane-sweeping stereo with multiple sweeping directions,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 1-8 (2007).

[9] J. Xiao, A. Owens, and A. Torralba, “SUN3D: A database of big spaces reconstructed using sfm and object labels,” in *Proc. of IEEE International Conference on Computer Vision*, 1625-1632 (2013).

[10] J. Sturm, et al., “A benchmark for the evaluation of RGB-D SLAM systems,” in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 573-580 (2012).

[11] S. Fuhrmann, F. Langguth, and M. Goesele, “MVE-a multi-view reconstruction environment,” in *Proc. of EurographicsWorkshop on Graphics and Cultural Heritage*, 11-18 (2014).

[12] A. X. Chang, et al., “ShapeNet: An information-rich 3d model repository,” *arXiv preprint arXiv:1512.03012*, 2015.

[13] J. Shotton, et al., “Scene coordinate regression forests for camera relocalization in RGB-D images,” in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2930-2937 (2013).