2019 International Conference on Robotics and Automation (ICRA)
Palais des congres de Montreal, Montreal, Canada, May 20-24, 2019

Depth Completion with Deep Geometry and Context Guidance

Byeong-Uk Lee[1], Hae-Gon Jeon[2], Sunghoon Im[1] and In So Kweon[1]

***Abstract*— In this paper, we present an end-to-end convolutional neural network (CNN) for depth completion. Our network consists of a geometry network and a context network. The geometry network, a single encoder-decoder network, learns to optimize a multi-task loss to generate an initial propagated depth map and a surface normal. The complementary outputs allow it to correctly propagate initial sparse depth points in slanted surfaces. The context network extracts a local and a global feature of an image to compute a bilateral weight, which enables it to preserve edges and fine details in the depth maps. At the end, a final output is produced by multiplying the initially propagated depth map with the bilateral weight. In order to validate the effectiveness and the robustness of our network, we performed extensive ablation studies and compared the results against state-of-the-art CNN-based depth completions, where we showed promising results on various scenes.**

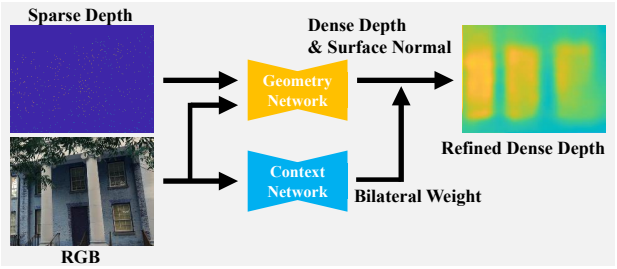
I. INTRODUCTION

3D scene information is being widely utilized in the robotics and computer vision fields for autonomous vehicles, SLAM, augmented reality, and other applications. Unfortunately, all of the current commercial devices used for 3D acquisition have pros and cons in terms of reliability, cost, capturing the environment and scene configuration.

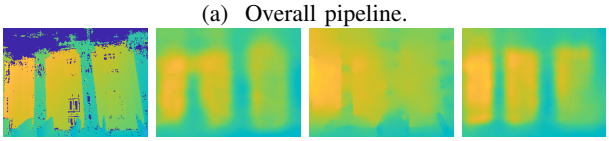
As is widely known, the most reliable device for 3D acquisition is 3D LiDAR which has a wide field of view and depth ranges as well as high accuracy. LiDARs work synergistically with cameras for visual perception tasks such as visual SLAM and segmentation, because the 3D points from the LiDAR provide additional information to the scene. However, 3D LiDARs are cost-prohibitive and provide only sparse measurements. Although structured light-based devices (KINECT, Real Sense, etc.) are able to obtain denser 3D measurements than the 3D LiDARs, they have short scanning ranges, and bright sunlight makes those measurements sparse. A passive approach, stereo matching, is an alternative way to capture the dense 3D depth of a scene, but its reliability with object boundaries of similar colors and textureless regions still remains a problem. Its reliability can be enhanced by using confidence measures [6], [18] which remove unreliable pixels, but they require an additional post-processing depth completion process. Because of the limitations of these commercial devices, depth completion has become an essential issue to resolve before 3D information can be practically used.

[1]Byeong-Uk Lee, Sunghoon Im and In So Kweon are with the School of Electrical Engineering, KAIST, Daejeon 34141, Republic of Korea. E-mail: {view94, dlarl8927, iskweon77}@kaist.ac.kr

[2]Hae-Gon Jeon is with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3890, USA. E-mail: haegonj@cs.cmu.edu



(a) Overall pipeline.



(b) GT depth (c) [13] (d) [28] (e) Ours

Fig. 1: Overall pipeline and results comparisons to state-of-the-art methods.

Traditional methods for depth completion propagate sparse depth points, guided by image-dependent propagation weights which are determined by various approaches. In [11], [21], a joint bilateral filter whose propagation weights depended on the spatial and range information of an image, was used for depth map completion. Park *et al.* [16] proposed a joint weight of color similarity and non-local means. Yang [27] built a minimum spanning tree based on the color intensities of a corresponding image for a non-local depth propagation. Despite various such attempts to make optimal propagation weights, these approaches commonly suffer from severe errors in large homogeneous regions and repeated patterns.

The recent success of convolutional neural networks (CNN) has produced significant progress in depth completion methods in the past few years. CNN-based depth completion can be categorized into two major classes: the use of modified convolution operations [24], [7] and deep feature-guided propagation, which we focus on here. Among the deep feature-guided propagation methods, Ma and Karaman [13] used an encoder-decoder network. The encoder is based on a residual network (ResNet) [5] pre-trained on the ImageNet dataset [19], and the decoder generates dense depth maps. In [10], an encoder-decoder style network was modified for multi-task learning in order to output dense depth maps and semantic segmentation from a single sparse depth map and its corresponding image. Zhang and Funkhouser [28] generated dense depth maps using a weight matrix which describes a surface normal and occlusion boundary. Although the surface normal and the occlusion boundary were estimated by CNNs,

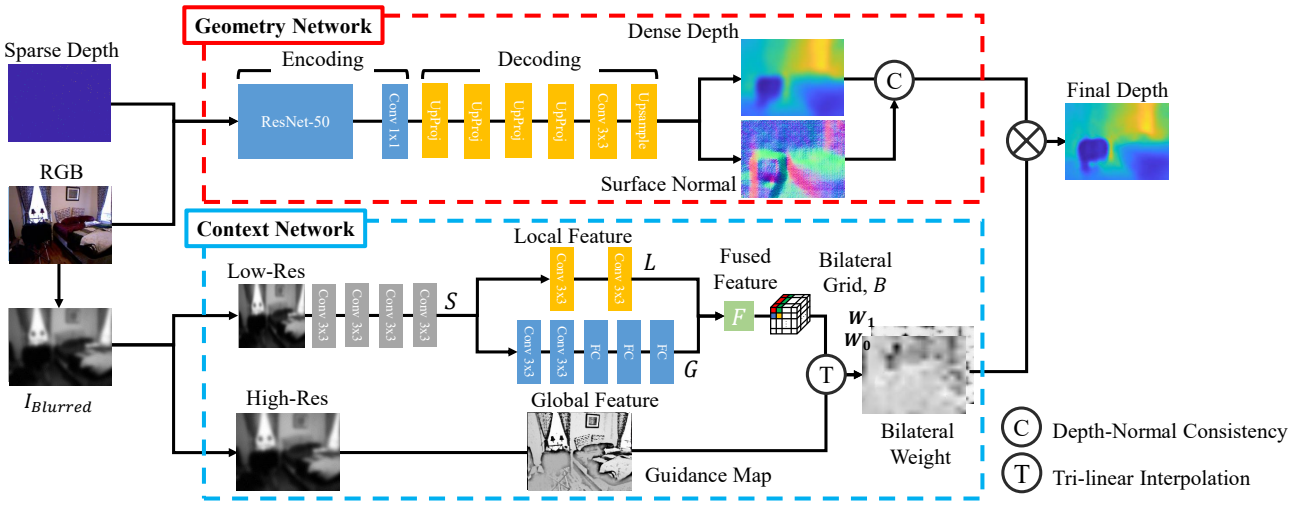


Fig. 2: Overview of the proposed depth completion network.

depth propagation was carried out by solving a traditional least squares problem with the weight matrix.

In this paper, we propose an end-to-end CNN consisting of two novel sub-networks: a geometry network and a context network (See Fig. 1). The geometry network has an encoder-decoder architecture, and predicts a surface normal of a scene and an initial dense depth map. The uniqueness of this approach over previous works [13], [28] is the use of a sparse depth map as well as its corresponding image. By optimizing them jointly in a training step, we were able to alleviate the front-parallel issues of the initial dense depth on slanted surfaces, as demonstrated in a traditional manner [9]. Another contribution of our work is to introduce the concept of a bilateral weight, which was used for CNN-based colorization [8], to predict a sharper dense depth map. The context network learns to produce the bilateral weight to capture local features such as strong edges, sharp corners and thin structures, and global features like the histogram, average intensity or even the scene category of an image. In the end, we produce a final dense depth map by simply multiplying the bilateral weight to the initial dense depth map. We will describe the technical details of our method in Sec. II. We demonstrate the robustness and the effectiveness of the proposed network on various scenes using extensive quantitative and qualitative evaluations, and compare the results to CNN-based state-of-the-art depth completion methods [13], [28] in Sec. III.

II. APPROACH

Our network consists of two sub-networks: a geometry network to jointly estimate dense depth and surface normal, and a context network to integrate contextual information for the depth map. The overall framework is described in Fig. 2.

A. Geometry Network

The geometry network takes four-channel input, where the first three channels represent an RGB image and the last channel is composed of a prior sparse depth. We pass the four-channel input through an encoder-decoder network as shown in the red box of Fig. 2. The encoding part is based on ResNet-50 pre-trained on the ImageNet [19] (which neglects the last average pooling layer and linear transformation layer) with an additional convolution layer. The decoding part contains four up-projection layers as proposed in [12], followed by an upsampling layer. This preserves both the high-level information passed from coarser feature maps and fine local information provided in lower layer feature maps.

Given the four-channel input, our network outputs an initial dense depth map D_{pred} and three channel surface normal vector N_{pred}. We force the depth map and surface normal to be consistent in our training step [1]. To do this, we incorporate a depth-normal consistency term for the normal-guided depth completion into a loss function whose details will be described in Sec. II-C. The joint training improves the accuracy of all tasks, especially on slanted surfaces, while keeping the model capacity fixed as shown in Fig. 4. Implementation detail can be found at our project page in `sites.google.com/view/bulee`.

B. Context Network

It is widely known that edge-preserving filters such as a traditional bilateral filter [3], [15] can produce significant improvements in depth refinement. Inspired by non-learning-based filtering, we adopt a bilateral learning scheme [4], originally designed for high dynamic range image enhancement, and adjust it to obtain a bilateral weight, including the contexture information of a scene, for depth refinement. The bilateral learning scheme uses a bilateral grid rather than simply applying 3D convolutions to feature maps. The bilateral grid allows full connectivity to be expressed in all dimensions of the feature maps by operating 2D convolution on the spatial dimension. This lets us to learn the bilateral weight in a more expressive form than a standard bilateral filter, since it does not discretize the input image.

As shown in the blue box of Fig. 2, the context network is composed of two streams. The network input $I_{blurred}$ is

a gray-scale image, and is blurred by downsampling with a factor of 4, and then upsampled with a factor of 16. Accordingly, the size of reference image I_{ref}, whose size is $H \times W$, is upsampled by a factor of 4. We observe that minimizing the absolute error between the upsampled I_{ref} and $I_{blurred}$ helps recover sharp edges in the final depth maps by extracting more rich and reliable features.

1) Image feature extraction and bilateral grid prediction: The first stream takes a low resolution copy of $I_{blurred}$ and learns both local and global features. The local feature refers to semantic features and spatial location inside the image, and global feature means high-level scene descriptions. We first encode an input image with four convolutional layers with 3×3 filters to extract the low-level image features S. The local feature L is extracted by passing S through two consecutive convolutional layers with 3×3 filters and stride 1. The global feature G is obtained through two convolutional layers with 3×3 filters and stride 2, and three additional fully-connected layers. Finally, the local and global features are fused to form a fused feature F as below:

$$F_c[x,y] = \sigma\left(\sum_c G + \sum_c L[x,y]\right), \tag{1}$$

where x, y are the image pixel locations, c is the channel of the feature, and $\sigma(\cdot)$ is the ReLU activation function. The F is then linearly transformed and reshaped to form a bilateral grid B whose size is $H/s_h \times W/s_w \times d \times ch$. s_h and s_w are the ratio between the spatial size of the bilateral grid and the full-resolution image size. The d is the depth of the bilateral grid and is empirically set to 8. The ch is a channel of the bilateral grid. In this paper, we set ch to 2 because it was sufficient to represent the depth maps using a two dimensional space consisting of a weight and a bias.

2) Guidance map generation: The latter stream handles $I_{blurred}$ to capture high-level features such as edges or boundaries. In our implementation, the full-resolution image is fed to the network in order to obtain a guidance map g. Similar to [4], the guidance map is obtained from a simple pixel-wise nonlinear transformation which sums 16 scaled ReLU functions with 16 pairs of slope and shift for each scale, as defined below:

$$\begin{aligned} g[x,y] &= b_1 + \rho(a \cdot I_{blurred}[x,y] + b_0) \\ \text{s.t.} \quad \rho(x) &= \sum_{l=0}^{15} \psi_l \cdot \max(x - \eta_l, 0), \end{aligned} \tag{2}$$

where a, b_0 and b_1 are a scalar weight and biases, respectively. The function ρ is a summation of the 16 scaled ReLU functions with slopes ψ and thresholds η.

3) Full-resolution-sized bilateral weight acquisition: The final output of the context network is a bilateral weight with the same spatial size as $I_{blurred}$ ($4H \times 4W$), and 2 channels. Since the bilateral grid has a different size than the guidance map, we upsampled the bilateral grid using a tri-linear interpolation as below:

$$\begin{aligned} W_\gamma[x,y] &= \sum_{i,j,k} \tau(s_w x - i)\tau(s_h y - j)\tau(d \cdot g[x,y] - k) B_c[i,j,k] \\ &\text{s.t.} \quad \tau(x) = \max(1 - |x|, 0) \end{aligned} \tag{3}$$

where i, j, and k are the indices for the bilateral grid. Since our bilateral weight has 2 channels, $\gamma \in \{0,1\}$.

In the end, using the final bilateral weight W_γ, we produced a refined image from $I_{blurred}$ as well as a refined depth map $D_{refined}$ as shown below:

$$I_{refined}[x,y] = W_0[x,y] + W_1[x,y] \cdot I_{blurred}[x,y]. \tag{4}$$

$$D_{refined}[x,y] = W_0[x,y] + W_1[x,y] \cdot D_{pred}[x,y]. \tag{5}$$

We observed that W_γ can achieve better depth completion results when it produces a high-quality $I_{refined}$ as well. We have demonstrated this in our ablation study in Sec. III-A.

C. Loss Functions

Our loss function E is a linear combination of depth loss E_D, surface normal loss E_N, depth-normal consistency loss E_C, bilateral weight loss E_B, and final depth loss $E_{D'}$ as follows:

$$E = E_D + \lambda_N E_N + \lambda_C E_C + E_B + E_{D'}, \tag{6}$$

where the balance weights λ_N and λ_C are set to 0.33 and 0.001, respectively[1]. Each term will be described thoroughly in the following subsections, and we denote $\|\cdot\|_1$ and $\|\cdot\|_2$ as L_1-norm and L_2-norm, respectively.

1) Initial depth loss & surface normal loss: The initial depth and surface normal loss are determined as follows:

$$E_D = \sum \| D_{pred} - D_{gt} \|_1, \tag{7}$$

$$E_N = \sum \| N_{pred} - N_{gt} \|_1, \tag{8}$$

where D_{gt} and N_{gt} represent the ground-truth depth and surface normal, respectively. We note that the NYU Depth Dataset V2 [14], which we used as one of training sets, does not provide ground-truth data for surface normal. To train our network, we synthesized surface normal ground-truth data from the ground-truth depth map. In addition, since the NYU Depth Dataset V2 dataset has semi-dense depth maps, we only calculated the errors for pixels with valid depth values.

2) Depth-normal consistency loss: Surface normal of the 3D point p should be orthogonal to the plane where the point p lies. This means that the vectors from point p to its neighbor points q should also be orthogonal to the surface normal vector, i.e., the inner product with the surface normal vector should be zero, and is defined as:

$$E_C = \sum_{p,q \in N_p} \| < v(p,q), N(p) > \|_2^2, \tag{9}$$

where $< \cdot >$ is an inner product. The E_C measures the sum of all inner products between the surface normal vector and tangent vector from the point p to its neighbor q for all 3D points.

[1]The λ_N and λ_C are determined according to the works in [25], [28]

Model	RMSE	Rel	delta_1
No geometry & context	0.281	0.051	96.5
No context	0.241	0.05	96.8
No geometry	0.238	**0.046**	97.1
Ours (1x context)	0.237	0.050	97.1
Ours (2x context)	0.235	0.049	**97.2**
Ours without E_B	0.251	0.050	96.8
Ours	**0.225**	**0.046**	**97.2**

TABLE I: Ablation study: performance changes with and without each component of our network. (Dataset: NYU Depth Dataset V2)

3) Bilateral weight loss & final depth loss: The bilateral weight is used to sharpen the edge of the initial estimated depth map D_{pred}. When training the network, this weight is optimized by two loss terms. One is the bilateral weight loss E_B, and the other one is the final depth loss $E_{D'}$. The bilateral weight loss measures the L_2 loss between the original image with sharp edges and the enhanced image output O:

$$E_B = \sum \| I_{ref} - I_{refined} \|_2^2 . \tag{10}$$

Finally, the L_1-norm between the refined depth result $D_{refined}$ and the ground truth depth is added as follows:

$$E_{D'} = \sum \| D_{refined} - D_{gt} \|_1 . \tag{11}$$

With this loss term, both the geometry network and the context network are optimized to learn a better initial depth and the bilateral weight for depth refinement at the same time.

III. EXPERIMENTS

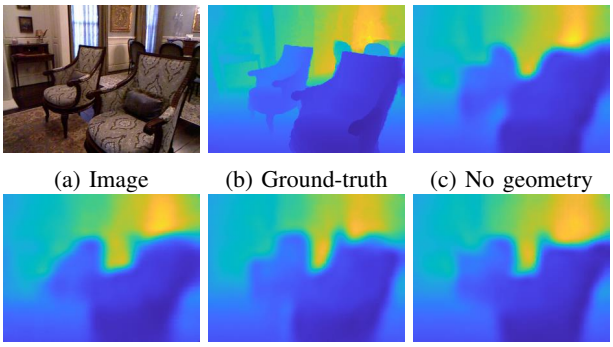
Our network was trained on RGB images and depth maps in NYU Depth Dataset V2. The NYU Depth Dataset V2 was captured from a KINECT, provided semi-dense depth information. With a cross-bilateral filter in authors' provided toolbox[2], we in-painted the depth maps. Our network takes images with 228×304 resolution made by resizing and center-cropping original images. Simple random image transformations were applied for data augmentation, such as random scaling, rotating, flipping, or color adjustment.

In a training procedure, we use image sequences, ground-truth depth maps, and surface normal maps generated from the ground-truth depth maps. The whole network was trained as an end-to-end manner, and the context network was trained from scratch. For the geometry network. The SGD optimizer was used with an initial learning rate of $1e-2$, the momentum of 0.9, and a weight decay of $1e-4$. The context network was also trained with the ADAM optimizer, but with an initial learning rate of $1e-4$ and weight decay of $1e-8$. We used a batch size of 16 and trained for 20 epochs. Our network is implemented by using PyTorch on a computer equipped with two NVidia 1080 Ti GPUs and total training time is about 12 hours.

A. Ablation study

First of all, extensive ablation studies are conducted to examine the effects of each component on our network.

[2] cs.nyu.edu/~silberman/datasets/nyu_depth_v2.html



(a) Image (b) Ground-truth (c) No geometry

(d) No context (e) w/o E_B (f) Ours

Fig. 3: The effect of our network inputs. Results from ablated models (without geometry, context and bilateral weight loss) and our complete model is shown.

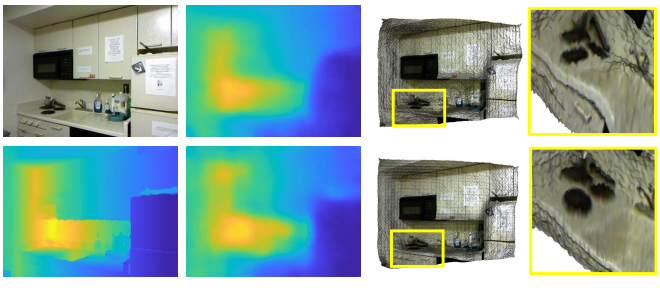
In our evaluations, we use common quantitative measures of depth quality: root mean square error (RMSE), mean absolute relative error (Rel) and inlier ratio within a threshold (delta_1), where delta_n is defined as:

$$\text{delta}_n = \frac{n(\{D_{refined} : \max\{\frac{D_{refined}}{D_{gt}}, \frac{D_{gt}}{D_{refined}}\} < 1.25^n\})}{n(\{D_{gt}\})}, \tag{12}$$

with $n(\cdot)$ as the cardinality of a set. The results are reported in Table I whose examples are shown in Fig. 3.

Depth-normal consistency loss E_C It is shown that geometry and context networks lead to significant performance improvements. The improvement with the geometry network is larger than that with the context network in Table I. The geometry network reliably propagates the initial 3D points guided by the depth-normal consistency. The 3D mesh results in Fig. 4 show that the geometry-consistency term effectively complete dense 3D scenes, even on the slanted and homogeneous regions where most errors made.

Bilateral weight loss E_B Our bilateral weight is learned to refine a blurred image as well as a predicted depth map via the bilateral loss term E_B in Eq. (10). Interestingly, E_B produces high-quality depth maps in Table I and Fig. 3. We observe that the multi-purpose loss for the bilateral weight encourages more effective image and depth map



(a) (b) (c)

Fig. 4: The effect of bilateral weight loss. (a) Reference image and GT depth. (b) Depth map results. (c) 3D meshes (Top: Ours w/o E_C term, Bottom: Ours).

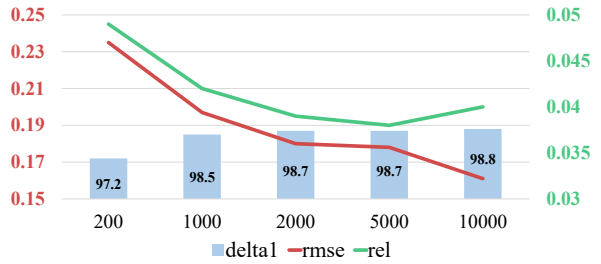


Fig. 5: RMSE & Rel (lower is better) and delta_1 (higher is better) w.r.t the number of depth samples.

usage to improve the semantic consistency and image fidelity simultaneously.

Bilateral weight training scheme We check the effectiveness of upsampling I_{ref} and $I_{blurred}$ to extract better bilateral weight. As shown in Table I, bilateral weights trained with upsampled images by a factor 4 shows better results on all measures than the use of smaller images. In fact, using upsampled images in traditional approaches helps to extract reliable features [23]. In the same manner, our context network also computes better deep features with higher resolution images. Of course, we could train our network with larger images, but this can cause severe blur and memory issue on it.

Number of Sparse Samples We examine the performance of the proposed method with respect to the number of samples. As displayed in Fig. 5, a greater number of samples yields better results in both accuracy and error measurements. Although the more number of sparse priors improves the quality of depth map, the performance is converged when the number of depth samples is more than 5000 which is about 1.5% of the total number of pixels.

B. Comparison with the state-of-the-arts

To test the robustness of our network quantitatively, we compare with the CNN-based state-of-the-art depth completion methods, Ma and Karaman [13] and Zhang and Funkhouser [28], trained on NYU Depth Dataset V2. In this experiment, we also used RMSE, Rel, delta_1 as error measures, and executed public source codes provided by the authors' website[3]. The number of initial depth samples is 200 which is less than 0.3% of the image pixels.

The results on NYU Depth Dataset V2 are shown in Table II and we display depth results in Fig. 7 as examples. We can see that [13] and [28] both give acceptable results by preserving scene context well, but the performance degradations happen in fine structures of scenes. In particular, occlusion boundaries in [28] help to maintain the sharpness of output depth maps. However, the estimated depth maps suffer from severe depth displacement errors as a z-axis because inaccurate normal estimation from single images leads to distort scene geometry, as in Fig. 6. On the other hand, our network can acquire dense depth maps and maintain better overall 3D structure, thanks to accurate surface normal information from both single images and sparse depth information.

[3][13]:github.com/fangchangma/sparse-to-dense, [28]:github.com/yindaz/DeepCompletionRelease

Model	RMSE	Rel	delta_1
Ma and Karaman [13]	0.281	0.051	96.5
Zhang and Funkhouser [28]	0.229	0.049	96.7
Ours	**0.225**	**0.046**	**97.2**

TABLE II: Quantitative evaluation on NYU Depth Dataset V2 (# of sample=200).

Datasets	Model	RMSE	Rel	delta_1
SUN3D	Ma and Karaman [13]	0.152	0.041	97.90
	Zhang and Funkhouser [28]	0.166	0.041	97.02
	Ours	**0.145**	**0.040**	**98.00**
RGBD	Ma and Karaman [13]	0.565	0.089	93.60
	Zhang and Funkhouser [28]	0.335	0.080	92.41
	Ours	**0.293**	**0.066**	**95.60**
MVS	Ma and Karaman [13]	3.344	0.269	93.30
	Zhang and Funkhouser [28]	0.533	**0.072**	92.87
	Ours	**0.514**	0.076	**95.30**

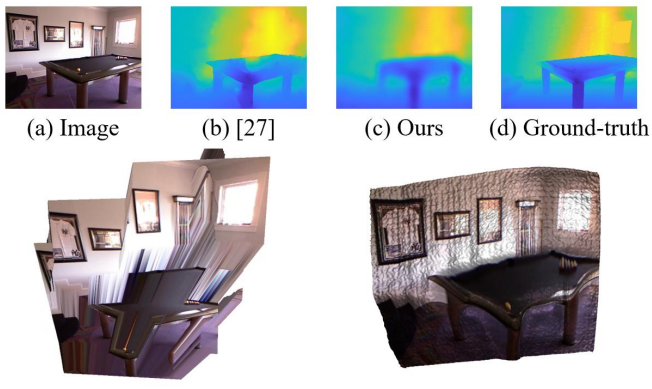
TABLE III: Quantitative evaluation on SUN3D, RGBD, MVS datasets (# of sample=700).

In Table III, we evaluate [13], [28] and our network on SUN3D [26], RGB-D [22] and MVS [20] datasets as shown in Fig. 8. All methods are not trained on these datasets and we verify the generality of our network. As shown in Table III, our network shows promising results on all the datasets, compared to [13] and [28]. In qualitative result, the geometry network guided by sparse depth maps produces useful surface normal information without loss of the generality, and the context network allows to yield sharper depth maps, and therefore our network outputs better result.

IV. CONCLUSION

We have presented an end-to-end CNN for depth completion. Our network mainly consists of two parts: a geometry network for handling slanted surfaces and a context network for preserving depth edges and details. We demonstrated its robustness and effectiveness versus state-of-the-art methods using various quantitative and qualitative evaluations.

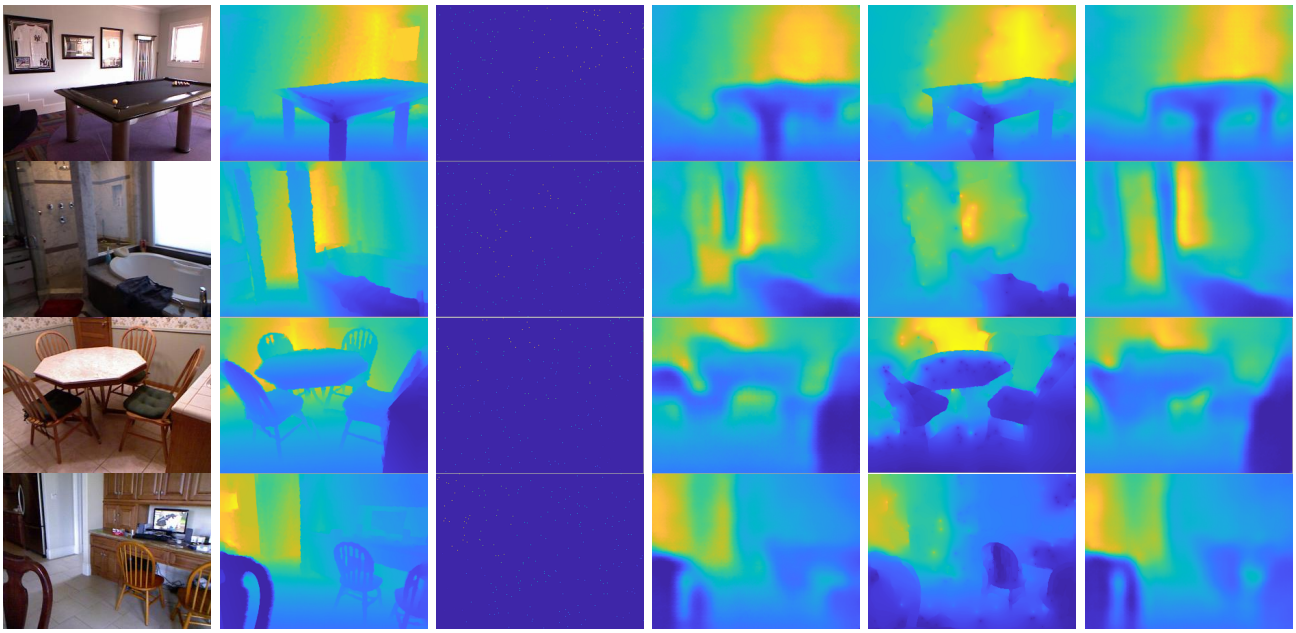
However, there is still room for improving our network.



(a) Image (b) [27] (c) Ours (d) Ground-truth

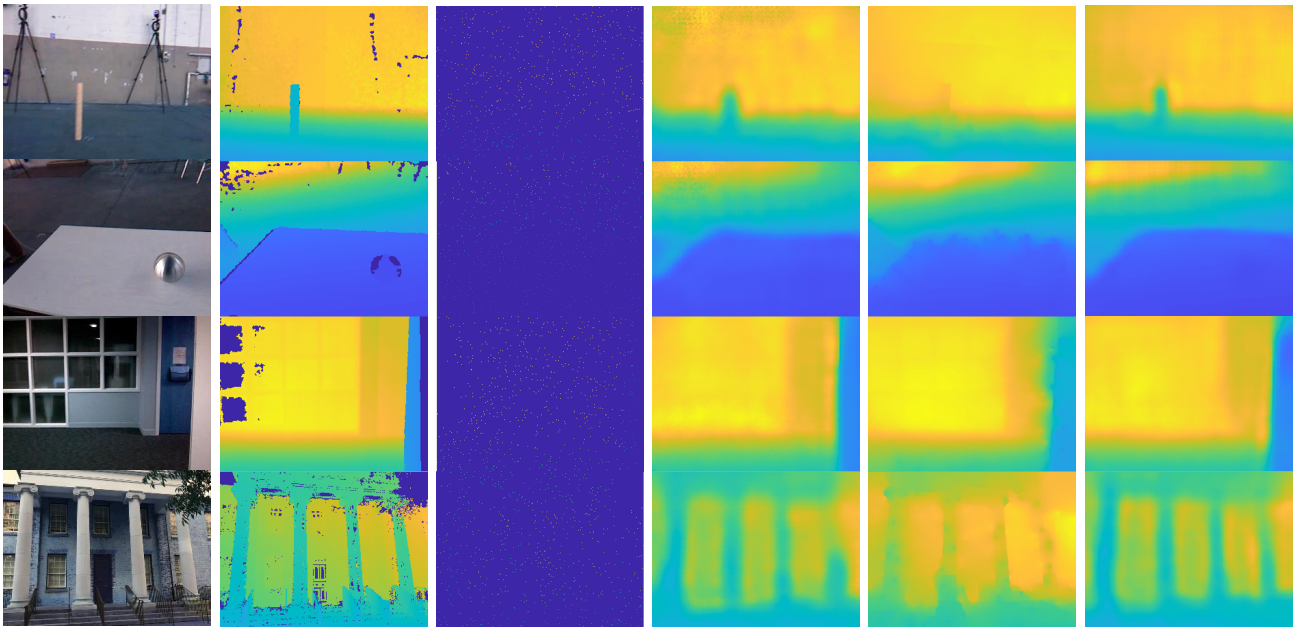
(e) 3D mesh with [27] result (f) 3D mesh with our result

Fig. 6: 3D mesh comparison with the results of [28] and ours



(a) Image (b) Ground-truth (c) Initial depth (d) [13] (e) [28] (f) Ours

Fig. 7: Comparison of depth map results on NYU Depth Dataset V2 on which all methods were trained.



(a) Image (b) Ground-truth (c) Initial depth (d) [13] (e) [28] (f) Ours

Fig. 8: Comparison of depth map results on SUN3D, RGB-D and MVS datasets.

First, our context network often fails to obtain proper features in regions of bright sunlight and deep shadow. We think that applying CNN-based intrinsic decomposition [2] to our network can be a good solution to the problem. In addition, we expect that temporal information of the image sequences will be helpful for refining propagation errors, as demonstrated in [17]. As a future work, we plan to adopt recurrent neural networks for the temporal information.

ACKNOWLEDGEMENT

This was was supported by the Technology Innovation Program funded by the Ministry of Trade, Industry and Energy, South Korea, under Grant 2017-10069072. This work was also in part by Air Force Research Laboratory (AFRL) project FA23861714660. Hae-Gon Jeon was partially supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2018R1A6A3A03012899).

REFERENCES

[1] T. Dharmasiri, A. Spek, and T. Drummond. Joint prediction of depths, normals and surface curvature from rgb images using cnns. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[2] Q. Fan, J. Yang, G. Hua, B. Chen, and D. Wipf. Revisiting deep intrinsic image decompositions. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[3] S. Fleishman, I. Drori, and D. Cohen-Or. Bilateral mesh denoising. In *ACM Transactions on Graphics (TOG)*, volume 22, pages 950–953, 2003.

[4] M. Gharbi, J. Chen, J. T. Barron, S. W. Hasinoff, and F. Durand. Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics (TOG)*, 36(4):118, 2017.

[5] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[6] X. Hu and P. Mordohai. A quantitative evaluation of confidence measures for stereo vision. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2121–2133, 2012.

[7] J. Hua and X. Gong. A normalized convolutional neural network for guided sparse depth upsampling. In *International Joint Conference on Artificial Intelligence*, 2018.

[8] S. Iizuka, E. Simo-Serra, and H. Ishikawa. Let there be color!: joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM Transactions on Graphics (TOG)*, 35(4):110, 2016.

[9] S. Im, H. Ha, G. Choe, H.-G. Jeon, K. Joo, and I. S. Kweon. High quality structure from small motion for rolling shutter cameras. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2015.

[10] M. Jaritz, R. De Charette, E. Wirbel, X. Perrotton, and F. Nashashibi. Sparse and dense data with cnns: Depth completion and semantic segmentation. In *International Conference on 3D Vision (3DV)*, 2018.

[11] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele. Joint bilateral upsampling. *ACM Transactions on Graphics (TOG)*, 26(3):96, 2007.

[12] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *International Conference on 3D Vision (3DV)*, 2016.

[13] F. Ma and S. Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2018.

[14] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2012.

[15] B. M. Oh, M. Chen, J. Dorsey, and F. Durand. Image-based modeling and photo editing. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 433–442, 2001.

[16] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I. S. Kweon. High quality depth map upsampling for 3d-tof cameras. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2011.

[17] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I. S. Kweon. High-quality depth map upsampling and completion for rgb-d cameras. *IEEE Transactions on Image Processing (TIP)*, 23(12):5559–5572, 2014.

[18] M. Poggi and S. Mattoccia. Learning to predict stereo reliability enforcing local consistency of confidence maps. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[19] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[20] J. L. Schonberger and J.-M. Frahm. Structure-from-motion revisited. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[21] I. Shim, S. Shin, Y. Bok, K. Joo, D.-G. Choi, J.-Y. Lee, J. Park, J.-H. Oh, and I. S. Kweon. Vision system and depth processing for drc-hubo+. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2016.

[22] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of rgb-d slam systems. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012.

[23] T. Tuytelaars, K. Mikolajczyk, et al. Local invariant feature detectors: a survey. *Foundations and trends® in computer graphics and vision*, 3(3):177–280, 2008.

[24] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger. Sparsity invariant cnns. In *International Conference on 3D Vision (3DV)*, 2017.

[25] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, and T. Brox. Demon: Depth and motion network for learning monocular stereo. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[26] J. Xiao, A. Owens, and A. Torralba. Sun3d: A database of big spaces reconstructed using sfm and object labels. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2013.

[27] Q. Yang. Stereo matching using tree filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 37(4):834–846, 2015.

[28] Y. Zhang and T. Funkhouser. Deep depth completion of a single rgb-d image. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.