Occlusion-Aware Depth Estimation with Adaptive Normal Constraints: Supplementary Materials

Xiaoxiao Long[1], Lingjie Liu[2], Christian Theobalt[2], and Wenping Wang[1]

[1] The University of Hong Kong
[2] Max Planck Institute for Informatics

1 More Results

Depth prediction Figure 1 shows more visual results of our method, mvdepthnet [4] and neuralrgbd [2]. Compared with mvdepthnet [4] and neuralrgbd [2], our estimated depth map has less noise, sharper boundaries and spatially consistent depth values, as can also be seen in the surface normal visualization. Furthermore, the 3D point cloud exported from our estimated depth better preserves global planar features (e.g. the wall corner in the first scene) and local features (e.g. the head model and the keyboard in the second scene).

Surface normal accuracy In Figure 2, we show more visual results of surface normals calculated from the estimated depth. Our method outperforms GeoNet [3] , Yin et al. [5] and Kusupati et al. [1] qualitatively. Compared with our model retrained using VNL, the normal values of our model trained with the *CNM* constraint are more consistent and smoother in planar regions. This indicates that *CNM*, which is a key novelty of our work, indeed contributes significantly to the overall improvement of performance, as compared with VNL.

Video-based 3D reconstruction In Figure 3, more video-based 3D reconstruction results are shown. For the white walls, the feature-less sofa and the table, our reconstructed result is much better than the results by the other two methods in terms of reconstruction quality of local and global structures.

2 Network structures

In this section, we illustrate the network structures used in our pipeline.

DepthNet. The structure of DepthNet is shown in Table 1, and we set $D = 64$ in our paper.

RefineNet. The structure of RefineNet is shown in Table 2, and we set $D = 64$ in our paper.

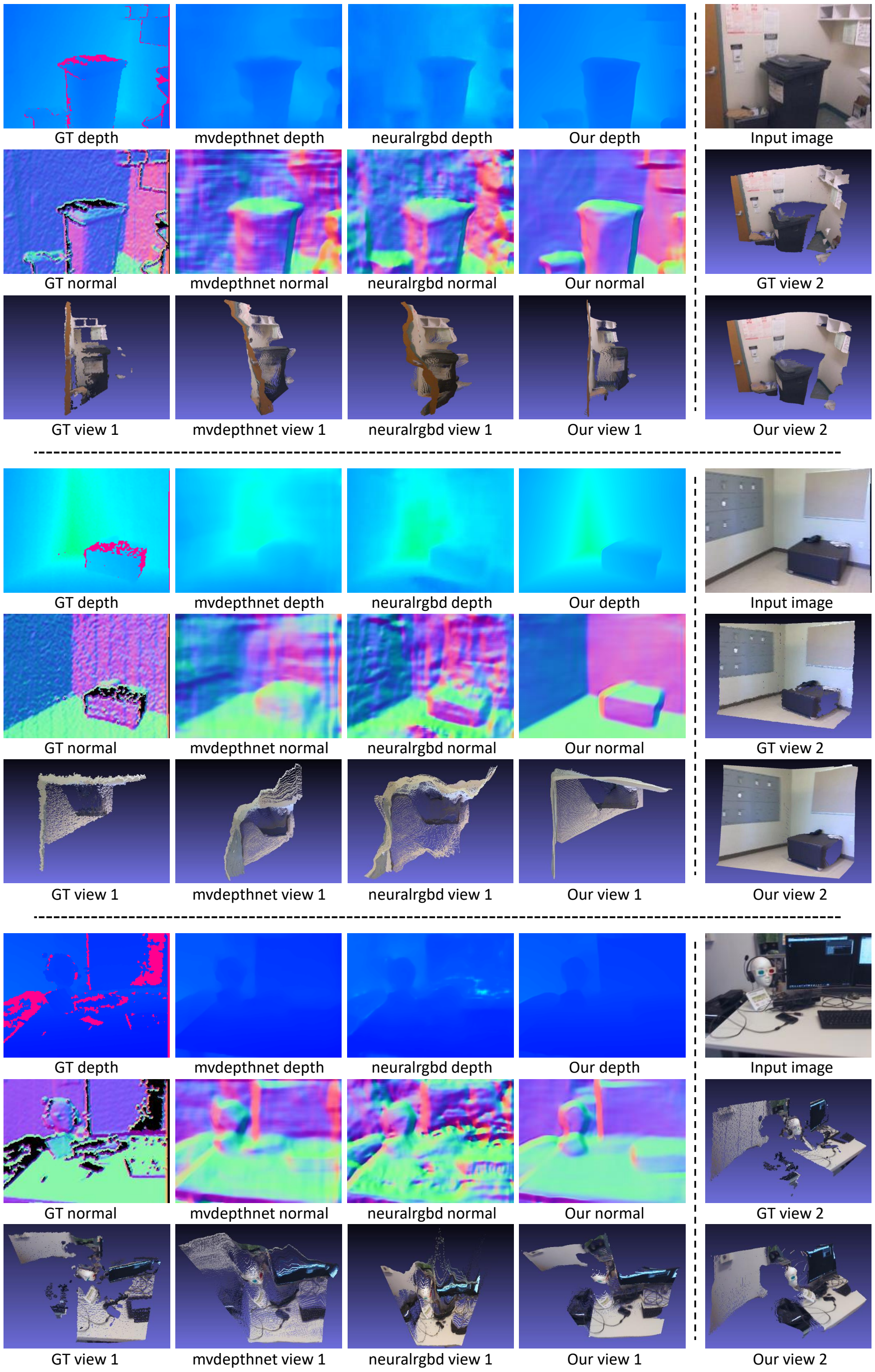

Fig. 1. Depth comparison with mvdepthnet [4] and neuralrgbd [2].

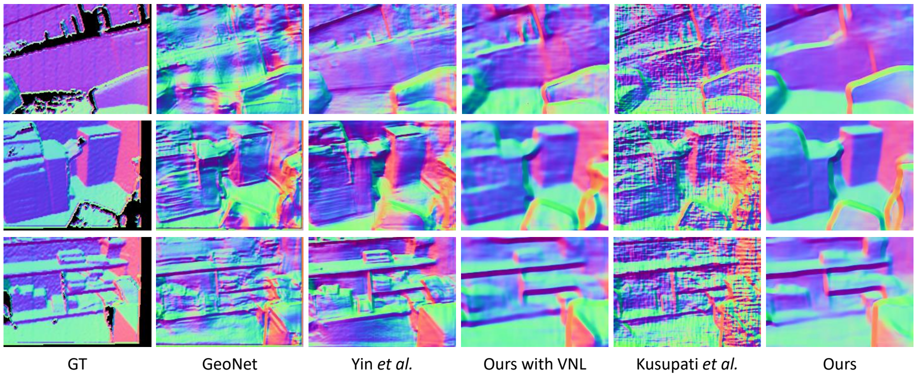

Fig. 2. Visual comparison of surface normal calculated from the estimated depth with GeoNet [3], Yin et al. [5] and Kusupati et al. [1].

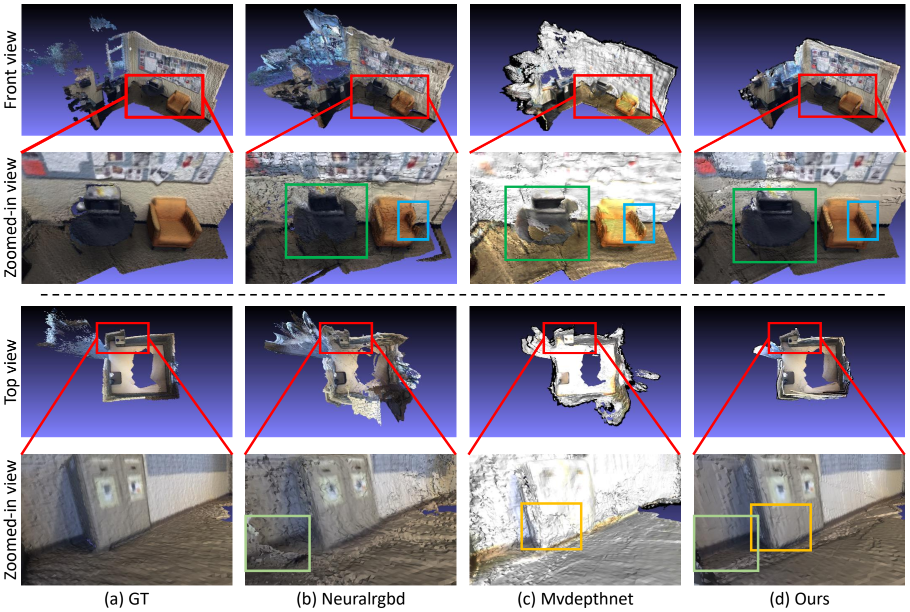

Fig. 3. Comparison with neuralrgbd [2] and mvdepthnet [4] for 3D reconstruction on scenes from ScanNet. (a) With ground truth depth; (b) With estimated depth and confidence map from neuralrgbd; (c) With estimated depth from mvdepthnet; (d) With our estimated depth and occlusion probability map. All reconstructions are done with TSDF fusion [6].

Table 1. The structure of DepthNet, which is built on [4]

Name	Layer components	Layer input	Output dimension
Input	concat (initial cost volume, reference image)		$W \times H \times 67$
conv1	$conv\_2d(7 \times 7, ch\_in = 67, ch\_out = 128, stride = 1), BN, ReLU$ $conv\_2d(7 \times 7, ch\_in = 128, ch\_out = 128, stride = 2), BN, ReLU$	Input	$\frac{1}{2}W \times \frac{1}{2}H \times 128$
conv2	$conv\_2d(5 \times 5, ch\_in = 128, ch\_out = 256, stride = 1), BN, ReLU$ $conv\_2d(5 \times 5, ch\_in = 256, ch\_out = 256, stride = 2), BN, ReLU$	conv1	$\frac{1}{4}W \times \frac{1}{4}H \times 256$
conv3	$conv\_2d(3 \times 3, ch\_in = 256, ch\_out = 512, stride = 1), BN, ReLU$ $conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 512, stride = 2), BN, ReLU$	conv2	$\frac{1}{8}W \times \frac{1}{8}H \times 512$
conv4	$conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 512, stride = 1), BN, ReLU$ $conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 512, stride = 2), BN, ReLU$	conv3	$\frac{1}{16}W \times \frac{1}{16}H \times 512$
conv5	$conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 512, stride = 1), BN, ReLU$ $conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 512, stride = 2), BN, ReLU$	conv4	$\frac{1}{32}W \times \frac{1}{32}H \times 512$
upconv5	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 512), BN, ReLU$	conv5	$\frac{1}{16}W \times \frac{1}{16}H \times 512$
iconv5	$conv\_2d(3 \times 3, ch\_in = 1024, ch\_out = 512), BN, ReLU$	concat (upconv5,conv4)	$\frac{1}{16}W \times \frac{1}{16}H \times 512$
upconv4	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 512), BN, ReLU$	iconv5	$\frac{1}{8}W \times \frac{1}{8}H \times 512$
iconv4	$conv\_2d(3 \times 3, ch\_in = 1024, ch\_out = 512), BN, ReLU$	concat (upconv4,conv3)	$\frac{1}{8}W \times \frac{1}{8}H \times 512$
depth4	$conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 1), Sigmoid$ *bilinear upsample*	iconv4	$\frac{1}{4}W \times \frac{1}{4}H \times 1$
upconv3	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 256), BN, ReLU$	iconv4	$\frac{1}{4}W \times \frac{1}{4}H \times 256$
iconv3	$conv\_2d(3 \times 3, ch\_in = 513, ch\_out = 256), BN, ReLU$	concat (upconv3,conv2,depth4)	$\frac{1}{4}W \times \frac{1}{4}H \times 256$
depth3	$conv\_2d(3 \times 3, ch\_in = 256, ch\_out = 1), Sigmoid$ *bilinear upsample*	iconv3	$\frac{1}{2}W \times \frac{1}{2}H \times 1$
upconv2	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 256, ch\_out = 128), BN, ReLU$	iconv3	$\frac{1}{2}W \times \frac{1}{2}H \times 128$
iconv2	$conv\_2d(3 \times 3, ch\_in = 257, ch\_out = 128), BN, ReLU$	concat (upconv2,conv1,depth3)	$\frac{1}{2}W \times \frac{1}{2}H \times 128$
depth2	$conv\_2d(3 \times 3, ch\_in = 128, ch\_out = 1), Sigmoid$ *bilinear upsample*	iconv2	$W \times H \times 1$
upconv1	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 128, ch\_out = 64), BN, ReLU$	iconv2	$W \times H \times 64$
iconv1	$conv\_2d(3 \times 3, ch\_in = 65, ch\_out = 64), BN, ReLU$	concat (upconv1,depth2)	$W \times H \times 64$
depth1	$conv\_2d(3 \times 3, ch\_in = 64, ch\_out = 1), Sigmoid$	iconv1	$W \times H \times 1$
Output	depth1		$W \times H \times 1$

Table 2. The structure of RefineNet.

Name	Layer components	Layer input	Output dimension
Input	concat (averaged cost volume, initial depths, the diffrence of initial depths)		$W \times H \times 67$
conv1	$conv\_2d(3 \times 3, ch\_in = 67, ch\_out = 128, stride = 1), BN, ReLU$ $conv\_2d(3 \times 3, ch\_in = 128, ch\_out = 128, stride = 2), BN, ReLU$	Input	$\frac{1}{2}W \times \frac{1}{2}H \times 128$
conv2	$conv\_2d(3 \times 3, ch\_in = 128, ch\_out = 256, stride = 1), BN, ReLU$ $conv\_2d(3 \times 3, ch\_in = 256, ch\_out = 256, stride = 2), BN, ReLU$	conv1	$\frac{1}{4}W \times \frac{1}{4}H \times 256$
conv3	$conv\_2d(3 \times 3, ch\_in = 256, ch\_out = 512, stride = 1), BN, ReLU$ $conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 512, stride = 2), BN, ReLU$	conv2	$\frac{1}{8}W \times \frac{1}{8}H \times 512$
	Depth refinement branch		
upconv3_d	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 256), BN, ReLU$	conv3	$\frac{1}{4}W \times \frac{1}{4}H \times 256$
iconv3_d	$conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 256), BN, ReLU$	concat (upconv3_d,conv2)	$\frac{1}{4}W \times \frac{1}{4}H \times 256$
upconv2_d	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 256, ch\_out = 128), BN, ReLU$	iconv3_d	$\frac{1}{2}W \times \frac{1}{2}H \times 128$
iconv2_d	$conv\_2d(3 \times 3, ch\_in = 256, ch\_out = 128), BN, ReLU$	concat (upconv2_d,conv1)	$\frac{1}{2}W \times \frac{1}{2}H \times 128$
upconv1_d	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 128, ch\_out = 64), BN, ReLU$	iconv2_d	$W \times H \times 64$
iconv1_d	$conv\_2d(3 \times 3, ch\_in = 64, ch\_out = 64), BN, ReLU$	upconv1_d	$W \times H \times 64$
depth	$conv\_2d(3 \times 3, ch\_in = 64, ch\_out = 1), Sigmoid$	iconv1_d	$W \times H \times 1$
	Occlusion probability prediction branch		
upconv3_p	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 256), BN, ReLU$	conv3	$\frac{1}{4}W \times \frac{1}{4}H \times 256$
iconv3_p	$conv\_2d(3 \times 3, ch\_in = 512, ch\_out = 256), BN, ReLU$	concat (upconv3_p,conv2)	$\frac{1}{4}W \times \frac{1}{4}H \times 256$
upconv2_p	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 256, ch\_out = 128), BN, ReLU$	iconv3_p	$\frac{1}{2}W \times \frac{1}{2}H \times 128$
iconv2_p	$conv\_2d(3 \times 3, ch\_in = 256, ch\_out = 128), BN, ReLU$	concat (upconv2_p,conv1)	$\frac{1}{2}W \times \frac{1}{2}H \times 128$
upconv1_p	*bilinear upsample* $conv\_2d(3 \times 3, ch\_in = 128, ch\_out = 64), BN, ReLU$	iconv2_p	$W \times H \times 64$
iconv1_p	$conv\_2d(3 \times 3, ch\_in = 64, ch\_out = 64), BN, ReLU$	upconv1_p	$W \times H \times 64$
probability	$conv\_2d(3 \times 3, ch\_in = 64, ch\_out = 1), Sigmoid$	iconv1_p	$W \times H \times 1$
Output	depth and probability		$W \times H \times 1$

References

1. Kusupati, U., Cheng, S., Chen, R., Su, H.: Normal assisted stereo depth estimation. arXiv preprint arXiv:1911.10444 (2019)
2. Liu, C., Gu, J., Kim, K., Narasimhan, S.G., Kautz, J.: Neural rgb (r) d sensing: Depth and uncertainty from a video camera. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 10986–10995 (2019)
3. Qi, X., Liao, R., Liu, Z., Urtasun, R., Jia, J.: Geonet: Geometric neural network for joint depth and surface normal estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 283–291 (2018)
4. Wang, K., Shen, S.: Mvdepthnet: real-time multiview depth estimation neural network. In: 2018 International Conference on 3D Vision (3DV). pp. 248–257. IEEE (2018)
5. Yin, W., Liu, Y., Shen, C., Yan, Y.: Enforcing geometric constraints of virtual normal for depth prediction. In: The IEEE International Conference on Computer Vision (ICCV) (2019)
6. Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J., Funkhouser, T.: 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In: CVPR (2017)