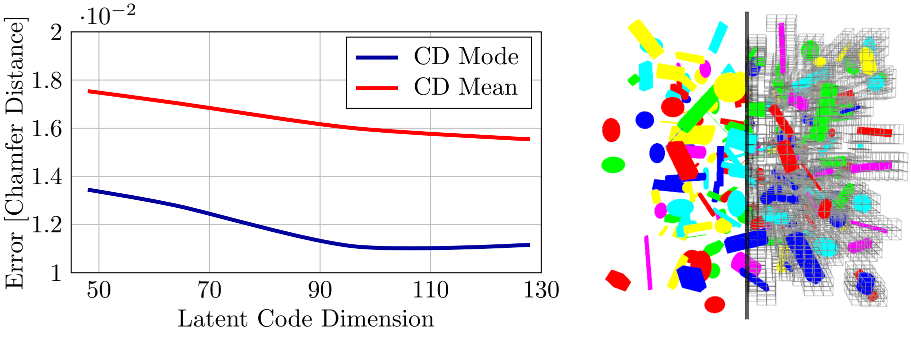
Supplemental Material: Deep Local Shapes: Learning Local SDF Priors for Detailed 3D Reconstruction

Rohan Chabra[1,3], Jan E. Lenssen[2,3], Eddy Ilg[3], Tanner Schmidt[3], Julian Straub[3], Steven Lovegrove[3], and Richard Newcombe[3]

[1] University of North Carolina at Chapel Hill
[2] TU Dortmund University
[3] Facebook Reality Labs

The supplementary material consists of detailed information about the experimental setup in Sec. 1, a detailed view on local shapes in Sec. 2, additional metrics for the 3D Warehouse dataset comparison in Sec. 4, and more results, comparisons and details about experiments in Sec. 5. Additionally, we provide a video alongside this document, showing reconstructions in motion.



(a) Latent code size

(b) Primitives for training

Fig. 1: Fig. (a) shows the effect of changing the latent code dimensions on the Chamfer distance test error on airplanes class of 3D Warehouse [1]. Fig. (b) shows an example for a scene containing 200 primitives shapes as used for training the local shape priors. On the right side, the instantiated local shape blocks are shown.

1 Experimental Setup

Autodecoder Network The DeepLS autodecoder network is a lighter version of the network proposed for DeepSDF [8]. It consists of four fully-connected layers, separated by leaky ReLUs and a tanh at the end, producing values in $[-1, 1]$ that are then scaled by the chosen SDF truncation value. Each layer has

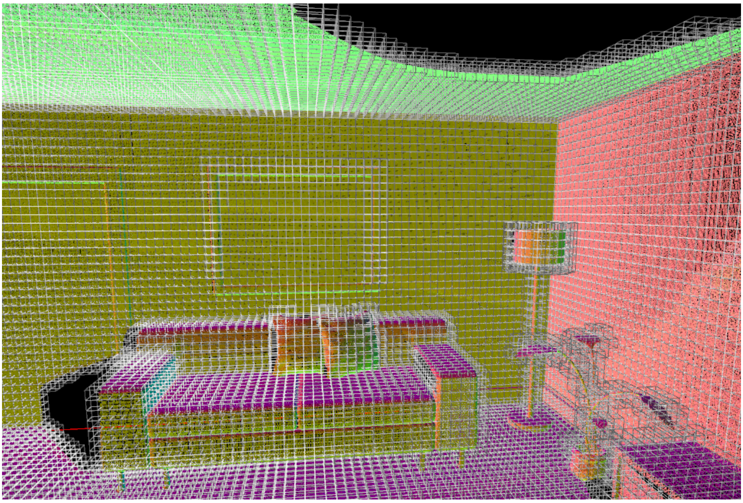


Fig. 2: Instantiated local shape blocks in a scene. The blocks are allocated sparsely, based on available depth data, which makes the approach scale well to real world inputs.

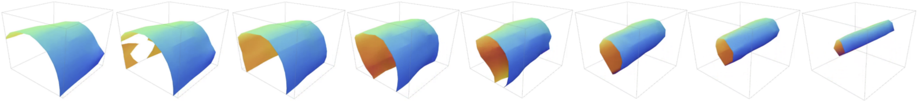


Fig. 3: Interpolation in latent space of a local shape code between a flat surface and a pole.

128 output neurons. Fig. 1a shows the result of a small study to find the best latent code size in a trade-off between accuracy and compression. We chose a latent size of 125, leaving us with 128 input neurons for the first network layer.

Training The output of the network is trained to produce truncated SDF values. To this end, tanh is also applied on the appropriately scaled ground truth SDF before computing the loss against the network output. We chose the scale so that the interval $[-0.9, 0.9]$ after tanh covers approximately two blocks. We optimize codes and network parameters using the Adam optimizer with initial learning rate of 0.01, which we decrease twice over the course of training.

Training Data The training data to learn local shape priors consists of three different categories of shapes. The first category contains simple primitive shapes, as shown in Fig. 1b, with random 6-DOF pose in space. The second category consists of 3D Warehouse [1] training meshes: We sampled a subset of 200 models from each training set of the classes *airplane*, *chair*, *lamp*, *sofa*, and *table*. Each model was split into $32 \times 32 \times 32$ local shape blocks. The last category consists of models from the Stanford 3D scanning repository [2], namely *bunny* and *dragon*.

2 Local Shape Space

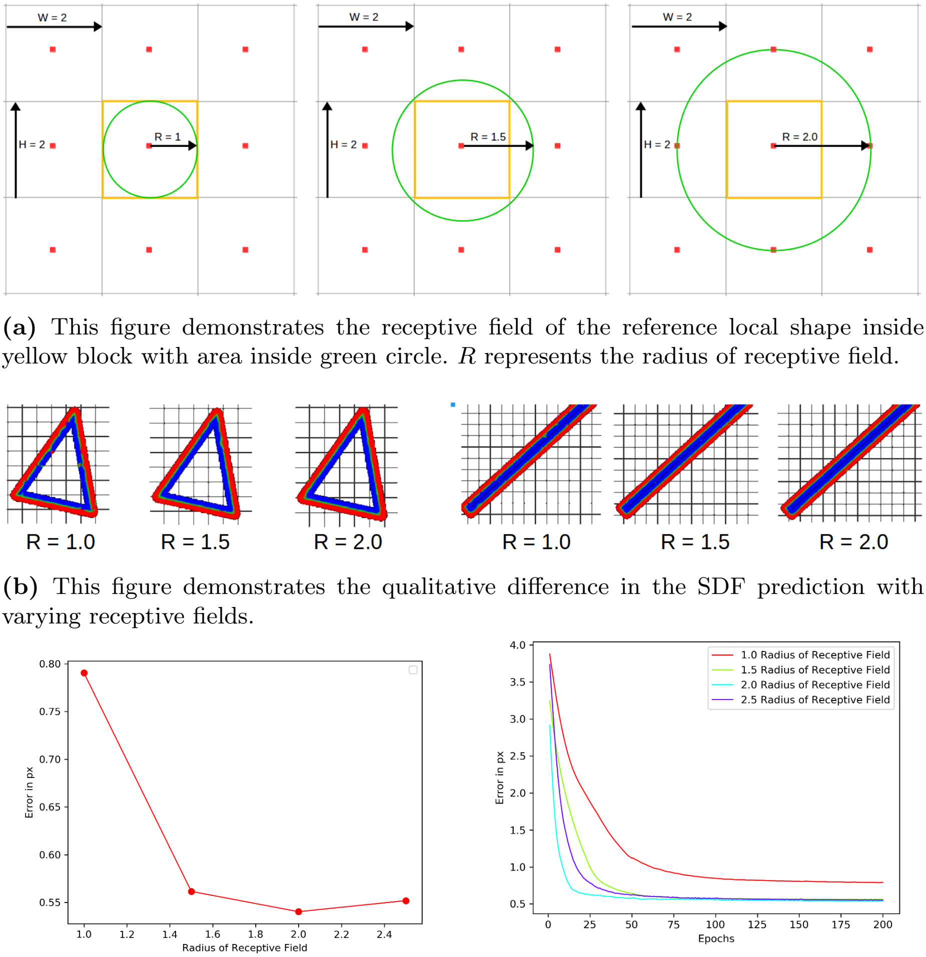
In order to give a better intuition about the space of learned local priors, interpolation sequences between local surfaces are provided in Fig. 3. It should be noted that, in general, the space of possible functions in a voxel is much larger. Therefore, training local priors heavily restricts the space of solutions to those producing local SDF functions that describe reasonable surfaces. The behavior of local shapes over the course of optimization is shown in the accompanying video. Additionally, Fig. 2 show all allocated blocks in a scene, which together reconstruct the whole surface.

3 Shape border consistency

In order to better understand the border consistency among the borders of local shapes we used simple 2D scenes often composed of simple primitive shapes such as triangles, rectangle and circles. In training and testing session we sample points around these shapes and extract SDF measurements as described in DeepSDF [8]. Note, we color code these sample points with red for positive, blue for negative and green for zero SDF measurements. In all the 2D experiments we use roughly 1000 samples inside a grid cell (local shape spatial size) in training session and 100 samples in test session. We report testing error as the SDF prediction error in 2d (pixels).
In the following experiment, in order to study shape border consistency we increased the receptive field of local shapes as shown in Figure 4a. By receptive field we mean the physical space of input samples for a particular local shapes. In general, we observe improvement in SDF prediction on the boundaries of local shapes with increasing receptive field as shown in Figure 4b. Although, we observe a critical point in the receptive field after which the performance drops as shown in Figure 4c. As increasing receptive field makes the local shapes bigger and more complicated so more parameters in the network F_θ are required to express the space of shapes. Hence, each F_θ has a critical point in the receptive field. We also observe the early convergence in optimization for optimal receptive field as shown in Figure 4d.

4 3D Warehouse [1] Comparison - Additional Metric

In Tab 1 we extend the comparison on 3D Warehouse [1] objects on other metrics. In addition to the Chamfer distance we show mesh accuracy, which is defined as the maximum distance d such that 90% of generated points are within d if the ground truth mesh. All metrics show the similar trend that DeepLS achieves way higher accuracy than the related object-level representations.



(a) This figure demonstrates the receptive field of the reference local shape inside yellow block with area inside green circle. R represents the radius of receptive field.

(b) This figure demonstrates the qualitative difference in the SDF prediction with varying receptive fields.

(c) Error in SDF prediction with increasing receptive field. Critical point in receptive field is observed.

(d) Plot shows the test error with increasing iterations in optimization. Optimal receptive field shows early convergence.

Fig. 4: This figure demonstrates the effect of receptive field on the quality of reconstruction of SDF.

CD, mean	chair	plane	table	lamp	sofa
AtlasNet-Sph.	0.752	0.188	0.725	2.381	0.445
AtlasNet-25	0.368	0.216	0.328	1.182	0.411
DeepSDF	0.204	0.143	0.553	0.832	0.132
DeepLS	**0.030**	**0.018**	**0.032**	**0.078**	**0.044**
CD, median					
AtlasNet-Sph.	0.511	0.079	0.389	2.180	0.330
AtlasNet-25	0.276	0.065	0.195	0.993	0.311
DeepSDF	0.072	0.036	0.068	0.219	0.088
DeepLS	**0.023**	**0.011**	**0.026**	**0.019**	**0.039**
Mesh acc., mean					
AtlasNet-Sph.	0.0330	0.0130	0.0320	0.0540	0.0170
AtlasNet-25	0.0180	0.0130	0.0140	0.0420	0.0170
DeepSDF	0.0090	0.0040	0.0120	0.0130	0.0040
DeepLS	**0.0009**	**0.0008**	**0.001**	**0.0012**	**0.0011**

Table 1: Representing unknown shapes from the 3D Warehouse [1] test set. In addition to the Chamfer distance, we provide mesh accuracy [9]. Lower is better for all metrics. It can be seen that all metrics show a similar trend.

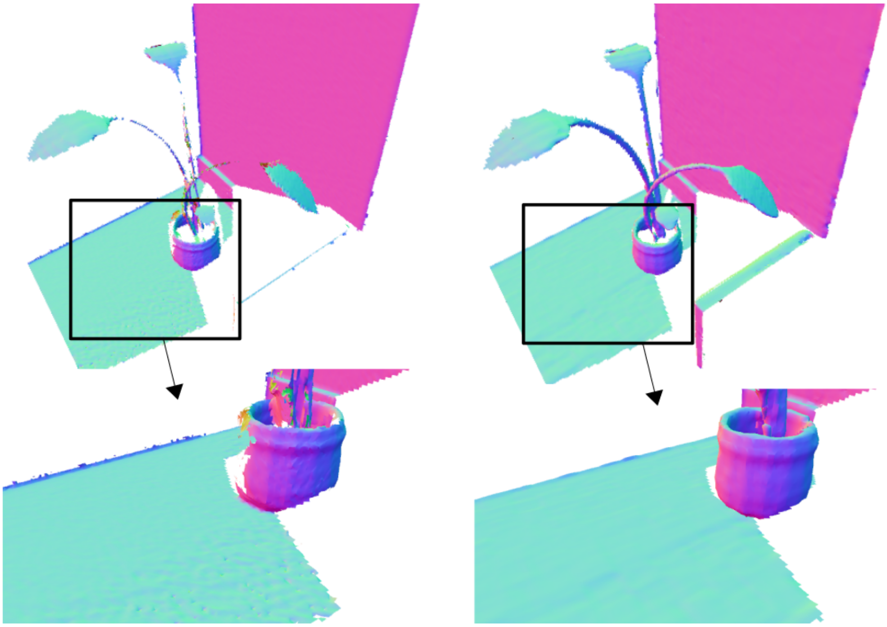
5 Scene Experiments

Here, we explain the process from depth maps to SDF samples for real scenes in more detail and provide qualitative results. See also the provided video for further results.

5.1 Sample Generation

Sample generation from depth scans consists of the following steps: (1) For a given scene, we generate a collection of 3D points from depth maps. (2) For each depth point, we create one sample with zero SDF, and several positive and negative SDF samples by moving the sample along the pre-computed surface normal by 1.5 cm and −1.5 cm, respectively. The accompanying SDF value is chosen as the moved distance.

(3) We generate additional free space samples along the observation rays. Further, we weight each set of points inversely based on the depth of the initial scan point, to ensure that accurate points closer to the scanning device are weighted higher. This procedure is described in detail in TSDF Fuison [7]. Similar to traditional SDF fusion approaches [7], DeepLS exposes a parameter which controls the size of the region around actual depth samples in which marching cubes is performed. Varying this parameter leads to the mesh accuracy vs. completion trade-off, discussed in the main paper.



(a) TSDF Fusion

(b) DeepLS

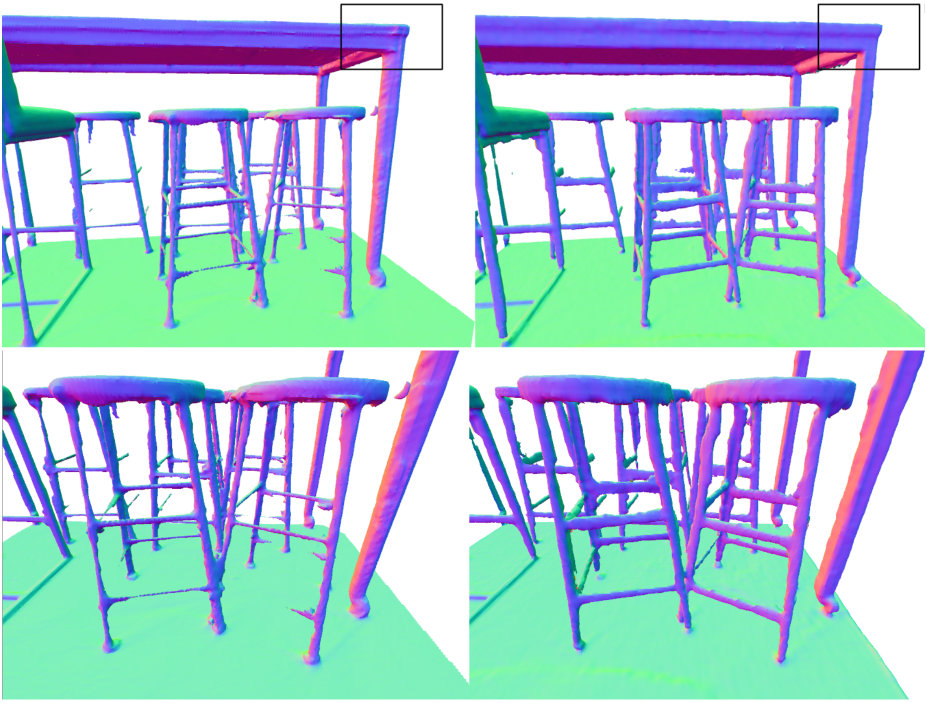
Fig. 5: The figure shows a part of the ICL-NUIM kt0 scene [5], reconstructed from samples with artitificial noise of $\sigma = 0.015$. DeepLS shows better denoising properties than TSDF Fusion. For the whole ICL-NUIM benchmark scene, DeepLS achieves a surface error of **6.41** mm with **71.04** % completion while TSDF Fusion has an error of 7.29 mm and 68.53 % completion.

5.2 Comparisons for Synthetic Noise

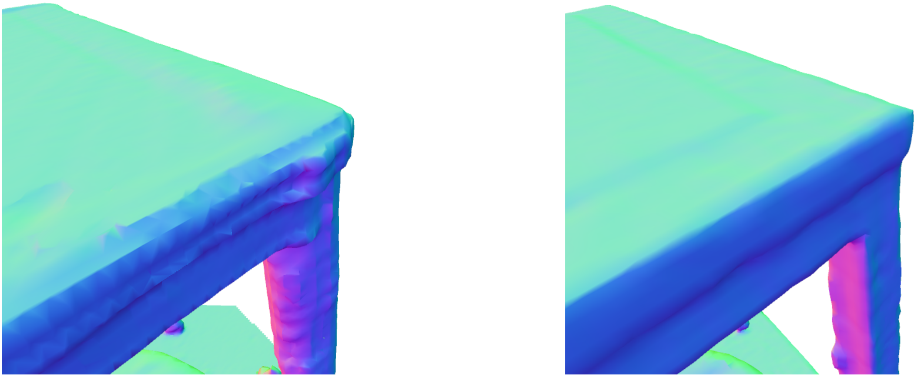
Fig. 5 shows results of DeepLS and TSDF Fusion on an ICL-NUIM benchmark scene with artificial noise of $\sigma = 0.015$. The learned local shape priors of DeepLS effectively are able to find plausible surfaces given the noisy observations, which results in smoother surfaces in comparison to TSDF Fusion.

5.3 Qualitative Results

We show additional qualitative results on real scanned data in Fig. 6, Fig. 7 and in the supplemented video. Both scenes showed in the figures were captured using a handheld structured light sensor system as was used for capturing the Replica dataset [10] and in related work [11, 3]. An in-house SLAM system, similar to state-of-the-art systems [4, 6], was used to provide 6 degree of freedom (DoF) poses for individual depth frames from the sensor. It can be seen that DeepLS succeeds in representing small details like the bars of chairs while TSDF Fusion tends to loose these details. Also, we observe sharper corners (c.f. 6b) and more complete surfaces (c.f. 7b) with DeepLS.

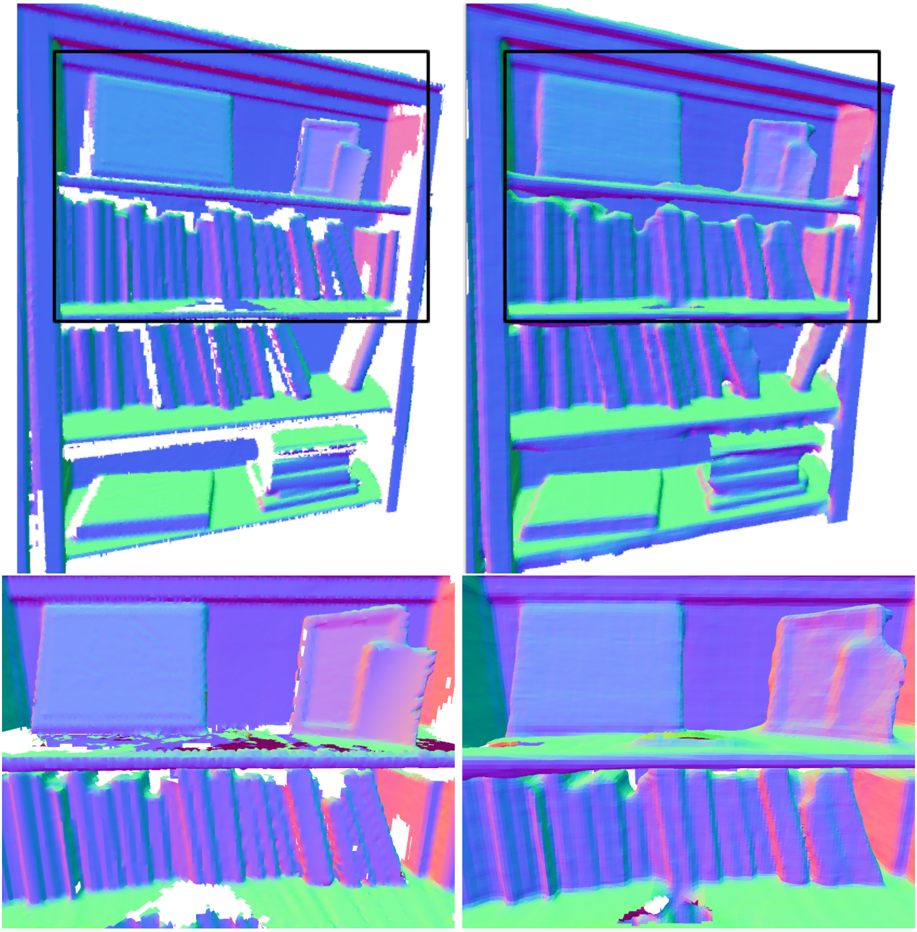


(a) DeepLS (right) captures thin chair legs better than TSDF Fusion (left) which tends to loose those details.



(b) Zoomed view of region marked with black box in (a). DeepLS (right) represents sharper corners and smoother planes than TSDF Fusion (left).

Fig. 6: Qualitative comparison of TSDF Fusion (left) with DeepLS (right) on real scanned data prepared using a structured light sensor system [10]. The figure (b) is the magnified region marked with black box in figure (a).



(a) TSDF Fusion [7]

(b) DeepLS

Fig. 7: We show the scene reconstruction quality of DeepLS vs TSDF Fusion [7] on a partially scanned real scene dataset using a structured light sensor system [10]. This figure shows that DeepLS provides better local shape completion than TSDF Fusion. The bottom row represents the zoomed in view marked with black box in the top row.

References

1. 3D Warehouse. https://3dwarehouse.sketchup.com/
2. The Stanford 3D Scanning Repository. http://graphics.stanford.edu/data/3Dscanrep/
3. Chabra, R., Straub, J., Sweeney, C., Newcombe, R., Fuchs, H.: Stereodrnet: Dilated residual stereonet. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11786–11795 (2019)
4. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. IEEE transactions on pattern analysis and machine intelligence **40**(3), 611–625 (2017)
5. Handa, A., Whelan, T., McDonald, J., Davison, A.: A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In: IEEE Intl. Conf. on Robotics and Automation, ICRA. Hong Kong, China (May 2014)
6. Mur-Artal, R., Montiel, J.M.M., Tardos, J.D.: Orb-slam: a versatile and accurate monocular slam system. IEEE transactions on robotics **31**(5), 1147–1163 (2015)
7. Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A.J., Kohi, P., Shotton, J., Hodges, S., Fitzgibbon, A.: Kinectfusion: Real-time dense surface mapping and tracking. In: Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on. pp. 127–136. IEEE (2011)
8. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. arXiv preprint arXiv:1901.05103 (2019)
9. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). vol. 1, pp. 519–528. IEEE (2006)
10. Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mur-Artal, R., Ren, C., Verma, S., Clarkson, A., Yan, M., Budge, B., Yan, Y., Pan, X., Yon, J., Zou, Y., Leon, K., Carter, N., Briales, J., Gillingham, T., Mueggler, E., Pesqueira, L., Savva, M., Batra, D., Strasdat, H.M., Nardi, R.D., Goesele, M., Lovegrove, S., Newcombe, R.: The Replica dataset: A digital replica of indoor spaces. arXiv preprint arXiv:1906.05797 (2019)
11. Whelan, T., Goesele, M., Lovegrove, S.J., Straub, J., Green, S., Szeliski, R., Butterfield, S., Verma, S., Newcombe, R.: Reconstructing scenes with mirror and glass surfaces. ACM Transactions on Graphics (TOG) **37**(4), 102 (2018)