

Local Implicit Grid Representations for 3D Scenes (Supplementary Material)

1. Additional implementation details

1.1. Model architecture

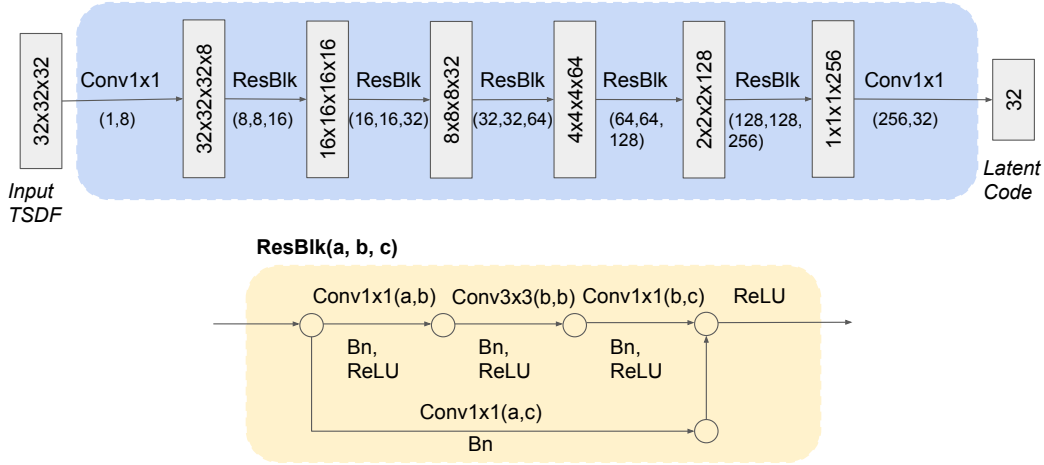


Figure 1: Encoder architecture. The encoder is a simple 3D CNN decorated with residue blocks, that encodes 3D TSDF tensors into latent codes, which can be decoded into implicit surfaces by an implicit network decoder.

We present a schematic of our encoder architecture for our part autoencoder in Fig. 1. The input to the encoder is a normalized TSDF crop of the part to be encoded, and the encoder uses 3D CNNs to encode the input into a latent code of dimensions 32. The encoder is decorated with residue blocks with bottleneck layers for improved performance.

We refer the reader to [1] for the architecture for our refiner. We preserve the architecture of the IM-NET model, but reduce the latent dimension from 128 to 32, and reduce the number of hidden layers in every layer of the model to 1/4 of the original value for improved efficiency, due to the fact that part geometries are easier to learn and represent than entire objects.

1.2. Part autoencoder training

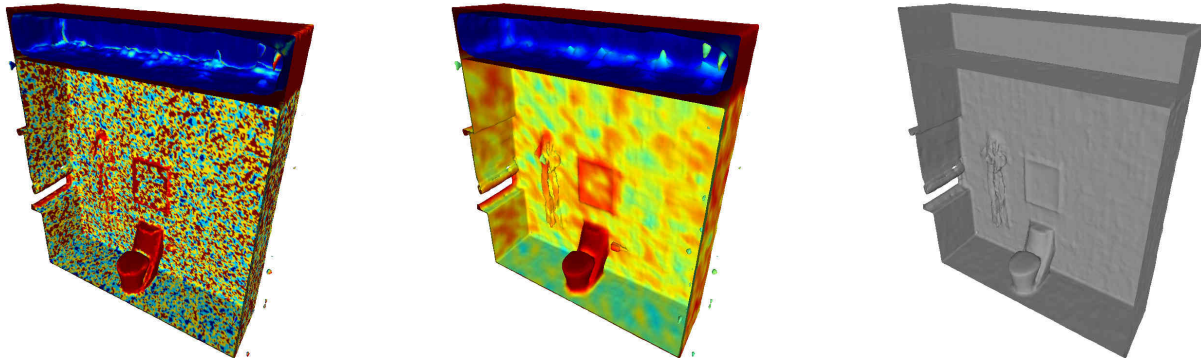
For training the part autoencoder, we use a batch size of 32, and for each shape we sample 2048 point samples. We train with a latent penalty factor $\lambda = 10^{-2}$, learning rate of 10^{-3} . We sample empty volumes with a probability of 10^{-3} to embed empty space. We train the part autoencoder for a total of 10^7 steps.

1.3. Inference

For reconstructing geometries from point samples, for each point sample, we sample 10 points along the point normal with a standard deviation of 1cm. For the Local Implicit Grid, we initialize each cell with Gaussian normal random values with a standard deviation of 0.01. During latent grid optimization, we use 32768 random point samples per batch, and optimize with a learning rate of 10^{-3} . We optimize for a fixed 10000 steps. When extracting the final mesh, we extract the mesh at $1/64m$ resolution.

1.4. Postprocessing algorithm

As discussed in the main text, one undesired side product from assuming all empty LIG grid cells to be “exterior” space is that it results in back-faces enclosed in large volumes. A simple postprocessing algorithm can be devised to remove such artifacts. For every face in the reconstructed mesh, we first compute the centroid of each face, as well as its normal direction. For the centroid of each face, we find the top-k nearest points in the original input oriented point set and compute the dot



(a) Before postprocessing. Color by original mesh normal alignment signal. (b) Before postprocessing. Color by normal alignment signal after Lap. smoothing. (c) Postprocessed Reconstructed Mesh. The back-face artifact in the original reconstructed mesh can be clearly seen in dark blue, and is effectively removed in the postprocessed mesh (c).

product of the normals between the pair of points. As such, back-faces will consistently have the opposite sign, and the exterior face will have the correct sign. This, however, will be noisy and non-robust to thin surfaces (with both sides very close to each other), since approximately half of the time the faces will find an input point on the opposite side as its nearest neighbor (see Fig. 2a). This can be effectively mitigated by using a Laplacian kernel (diffusion coefficient λ , i iterations) to smooth the normal alignment signal, followed by discarding all faces below a certain normal alignment threshold n , and discarding all disconnected components with an area below a .

In all our cases, we used the parameters $k = 3$, $n = -0.75$, $\lambda = 0.5$, $i = 50$, $a = 1$.

2. Additional ablation studies

We perform additional ablation studies on the effects of latent code length on reconstruction performance. See Table 1 and Fig. 3 for reference. With increasing number of latent channels, the reconstruction performance improves with diminishing marginal improvement. Our choice of 32 latent channels strikes a good balance between performance and efficiency.

CL	CD(\downarrow)	Normal(\uparrow)	F-Score(\uparrow)
8	0.018	0.925	0.879
16	0.013	0.944	0.923
32	0.012	0.961	0.957
64	0.012	0.965	0.963

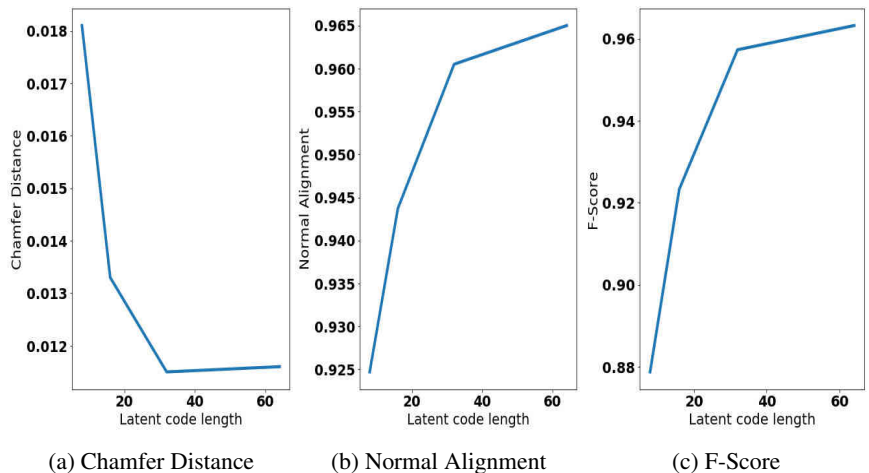


Table 1: Additional ablation study on the effects of latent code length (CL). Reconstruction performance measured on SceneNet reconstruction from 100 point samples / m^2 .

Figure 3: Line plot for Chamfer Distance, Normal Alignment and F-Score versus Latent Code Length.

3. Additional visual results

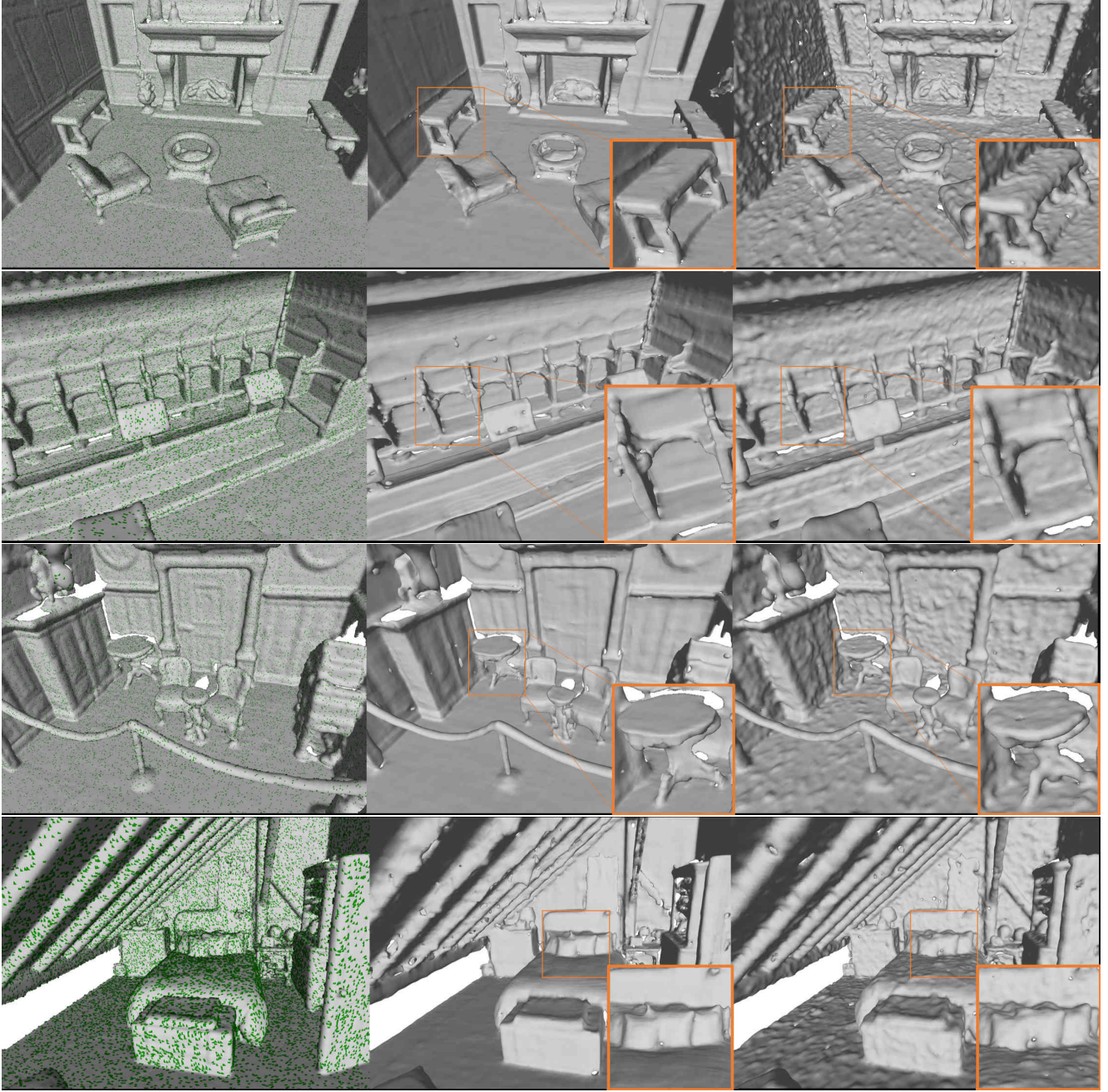


Figure 4: Left: Ground truth mesh overlaid with input point samples; Middle: Our reconstruction; Right: Screened PSR [2] reconstruction. The input are point samples from the Matterport ground truth mesh at a sample density of $500 \text{ points} / m^2$.

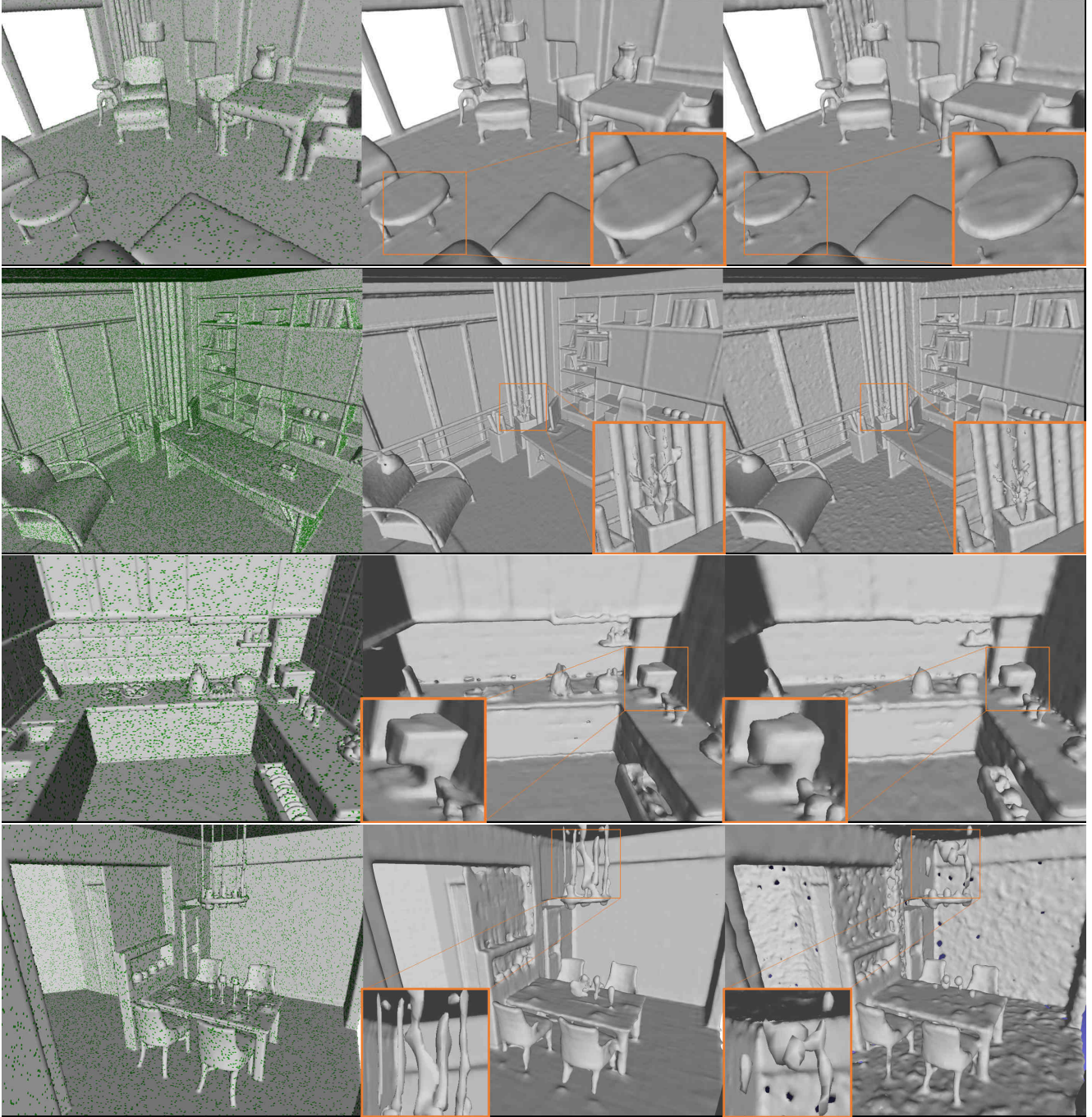


Figure 5: Left: Ground truth mesh overlaid with input point samples; Middle: Our reconstruction; Right: Screened PSR [2] reconstruction. The input are point samples from the SceneNet ground truth mesh at a sample density of $500 \text{ points} / m^2$.

References

- [1] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019. 1
- [2] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):29, 2013. 3, 4