

## A APPENDIX

### A.1 MORE DETAILS

**Volume sparsify.** The ground-truth occupancy volumes are generated based on the ground-truth TSDF volumes. The voxels with TSDF value of  $[-1, 1]$  are regarded as occupied and set to 1, otherwise set to 0. In the training or the inference, after the occupancy volume is predicted in the coarser level, the voxels of occupancy value less than 0.5 are regarded as empty and discarded, while the remaining voxels are regarded as occupied and transmitted to the next level. The sparse volume is stored in a hash table, where the key of the table is the hash value of the voxel, and the value of the table stores the corresponding feature. In the coarsest level, all voxels are regarded as non-empty and stored in the hash table, which do not consume much memory because the size of the volume is small.

	Method	Per Frame Time	Per Scene Time	FPS
	Atlas (Murez et al., 2020)	71 ms	840 ms	14
	NeuralRecon (Sun et al., 2021)	30 ms	0 ms	33
	TransformerFusion (Bozic et al., 2021)	130.5 ms	243.3 ms	7
	VoRTX (Stier et al., 2021)	71.4 ms	231.7 ms	14
	Ours	13.3 ms	286.4 ms	75

Table 5: Efficiency experiments.

### A.2 EFFICIENCY

The runtime analysis is presented in Table 5. For a fair comparison to previous methods, the time is tested on a chunk of size  $1.5 \times 1.5 \times 1.5 \text{ m}^3$  with an Nvidia RTX 3090 GPU. Our framework consists of two parts: one is the per-frame part, including the feature extraction of the 2D images; the other one is the per-scene part, including the feature fusion, 3D feature processing, and mesh generation. The per-frame model runs for every keyframe, i.e., it keeps running whenever a new keyframe comes. Differently, the per-scene model runs only once for generating a mesh reconstruction of a scene, i.e., it only works after all frames are fed, or when we need to output a mesh. Therefore, the online speed of a normal running is 75 FPS, which only performs the mesh generation once at the end.

Metrics	Definition
Abs Rel	$\frac{1}{n} \sum  d - d^*  / d^*$
Abs Diff	$\frac{1}{n} \sum  d - d^* $
Sq Rel	$\frac{1}{n} \sum  d - d^* ^2 / d^*$
RMSE	$\sqrt{\frac{1}{n} \sum  d - d^* ^2}$
$\delta - 1.25^i$	$\frac{1}{n} \sum (\max(\frac{d}{d^*}, \frac{d^*}{d}) < 1.25^i)$
Acc	$\text{mean}_{p \in P} (\min_{p^* \in P^*}   p - p^*  )$
Comp	$\text{mean}_{p^* \in P^*} (\min_{p \in P}   p - p^*  )$
Chamfer distance	$\frac{\text{Acc} + \text{Comp}}{2}$
Prec	$\text{mean}_{p \in P} (\min_{p^* \in P^*}   p - p^*   < 0.05)$
Recall	$\text{mean}_{p^* \in P^*} (\min_{p \in P}   p - p^*   < 0.05)$
F-score	$\frac{2 \times \text{Prec} \times \text{Recall}}{\text{Prec} + \text{Recall}}$

Table 6: Metric definitions.  $n$  denotes the number of pixels with both valid ground truth and prediction,  $d$  and  $d^*$  denote the predicted and the ground-truth depths,  $p$  and  $p^*$  denote the predicted and the ground-truth point clouds.

### A.3 METRICS

The definitions of the 2D metrics and 3D metrics used for evaluation are explained in Table 6.

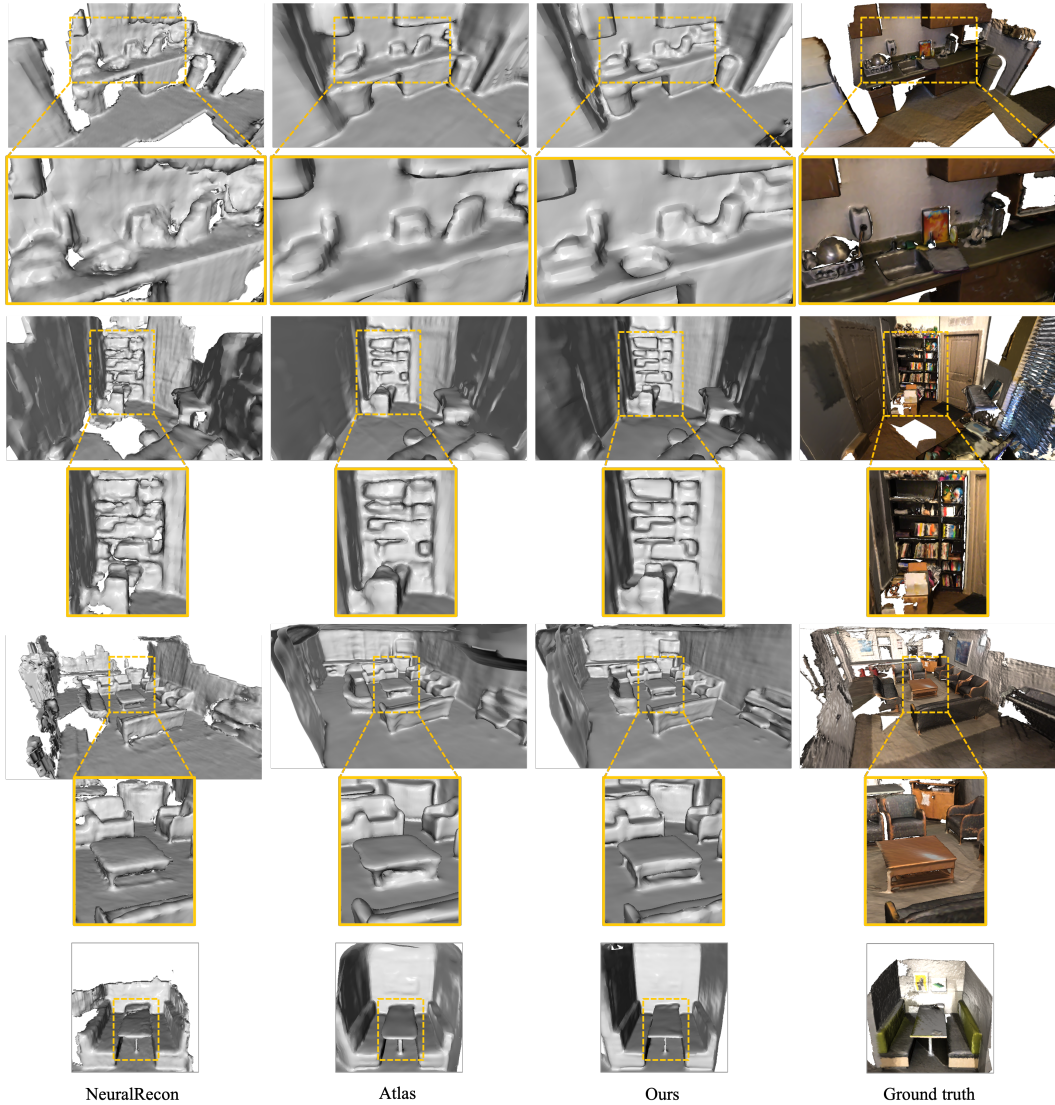


Figure 6: More qualitative results.

#### A.4 LIMITATIONS

Due to the volume representation, our framework is limited by the trade-off between the resolution of the volume and the memory consumption. A smaller voxel size would cost much more memory. The voxel size is set to 4 cm, such that the geometry details less than 4 cm are hard to be recovered.

#### A.5 MORE RESULTS

More results are presented in Figure 6 and Figure 7.

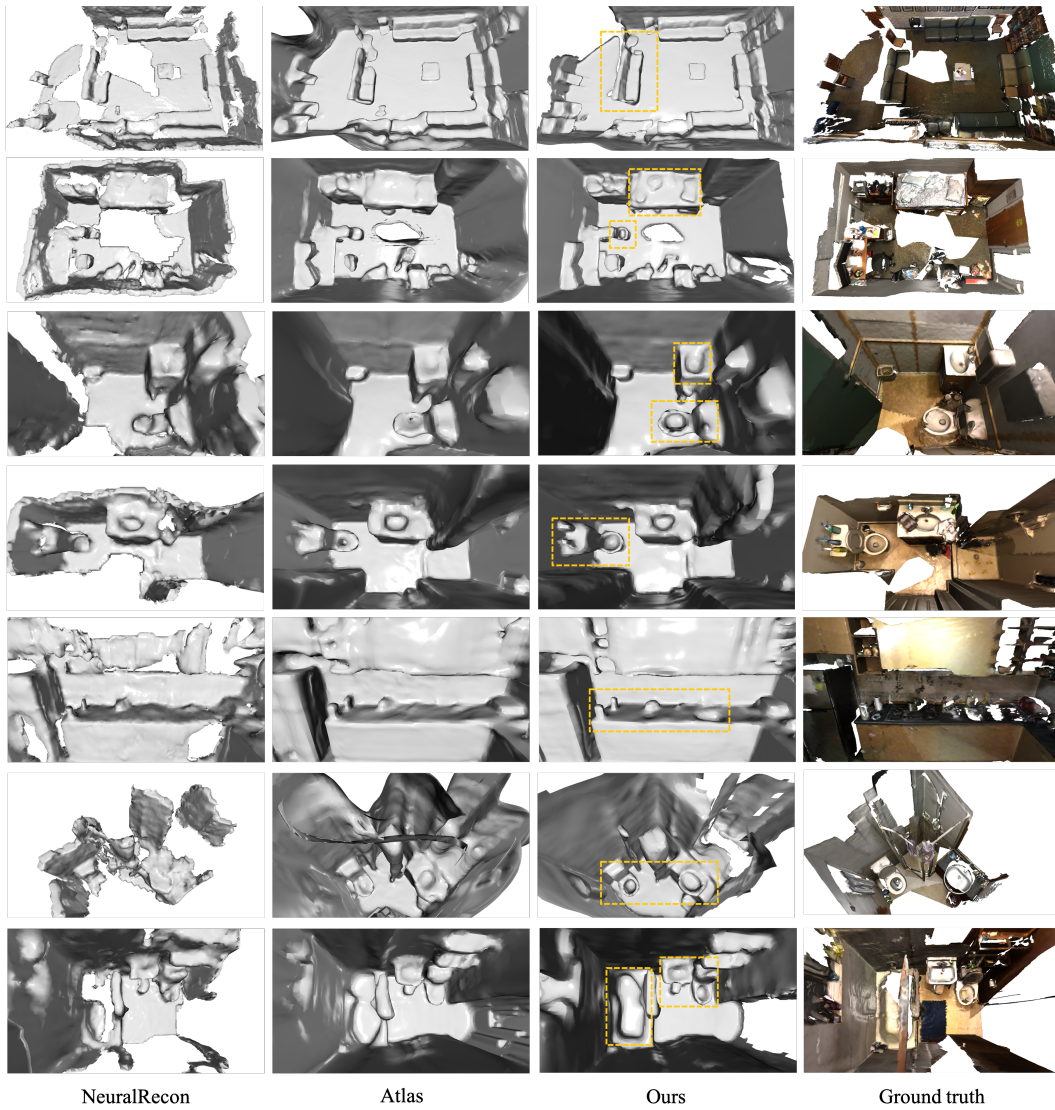


Figure 7: More qualitative results.