Supplementary Material for VMNet: Voxel-Mesh Network for Geodesic-Aware 3D Semantic Segmentation

Zeyu HU*[1], Xuyang Bai[1], Jiaxiang Shang[1], Runze Zhang[2], Jiayu Dong[2], Xin Wang[2], Guangyuan Sun[2], Hongbo Fu†[3], and Chiew-Lan Tai[1]

[1]Hong Kong University of Science and Technology
[2]Lightspeed & Quantum Studios, Tencent
[3]City University of Hong Kong

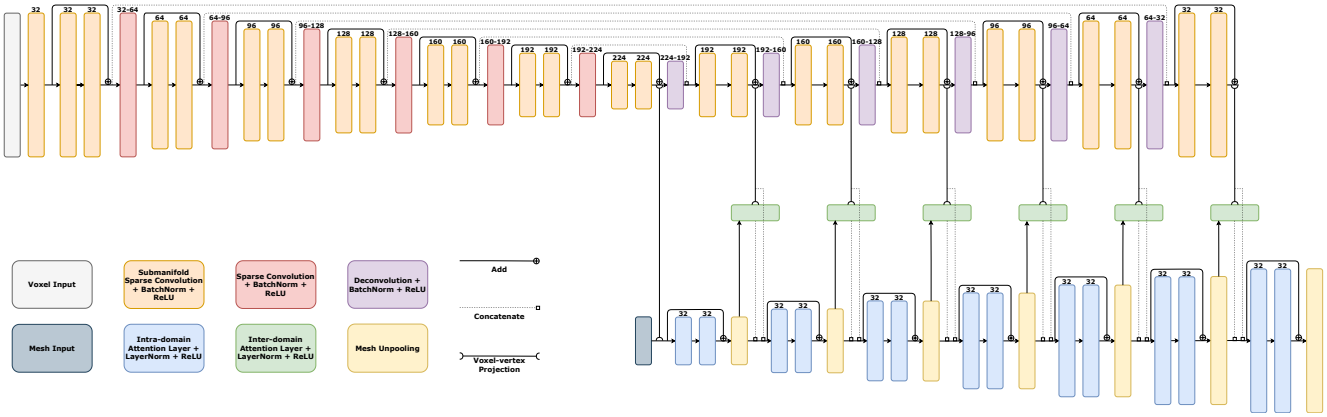

Figure 1. **Detailed network structure of VMNet.** We use a voxel-based 3D U-Net [6] as the contextual feature extractor, consisting of an encoder and a decoder. Afterwards, at each level of the decoder, the aggregated contextual features are first projected from the Euclidean domain to the geodesic domain, and then processed by the intra-domain attentive aggregation modules and the inter-doamin attentive fusion modules defined over triangular meshes, yielding distinctive per-vertex features enriched with both the Euclidean and geodesic information. The number above each layer indicates the feature channel.

Abstract

This supplementary document is organized as follows:

- *Section A depicts the detailed network structure of VMNet.*
- *Section B provides image illustrations of the mesh simplification methods used in VMNet.*
- *Section C shows more visualization results on the ScanNet [2] and Matterport3D [1] datasets.*
- *Section D presents more complexity comparisons of VMNet against other SOTA methods.*
- *Section E conducts an ablation study on Multi-level Feature Refinement.*
- *Section F discusses the design choice of the proposed inter-domain attentive module in VMNet.*

A. Detailed Network Structure

The network structure adopted in VMNet is illustrated in Fig. 1. VMNet consists of two branches, in which one operates on the voxel representation and the other operates on the mesh representation. In the upper branch (Euclidean branch), taking the voxels as input, we employ the widely used U-Net [6] style network for contextual feature aggregation. The network is mainly built upon submanifold sparse convolution layers and sparse convolution layers, both of which are originally introduced by Graham et al. [4]. In total, there are 7 levels of sparse voxels $(\mathcal{S}^0, ..., \mathcal{S}^l, ..., \mathcal{S}^6)$. At each level, there is a skip connection between the encoder and decoder. In the lower branch (geodesic branch), for each level of sparse voxels $\mathcal{S}^l$, we prepare a simplified triangular mesh $\mathcal{M}^l$, which is generated from the original mesh and has similar numbers of vertices to those of the corresponding sparse voxels $\mathcal{S}^l$. At level l, the aggregated contextual features are extracted from the decoder of the Euclidean branch and then projected from voxels $\mathcal{S}^l$ to mesh vertices $\mathcal{M}^l$ through voxel-vertex projection. On the mesh $\mathcal{M}^l$, the projected Euclidean fea-

*intern at Tencent Lightspeed & Quantum Studios
†corresponding author

tures are adaptively fused with the geodesic features utilizing the inter-domain attentive fusion modules. The fused features are then refined through the intra-domain attentive aggregation modules. The distinctive per-vertex features on the last mesh level $\mathcal{M}^0$ are used for semantic prediction.

B. Mesh Simplification

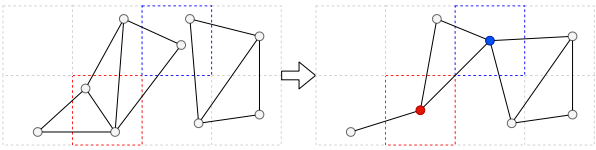

Figure 2. **Illustration of Vertex Clustering for mesh simplification.** Vertices falling in the same cell are merged to form a new vertex. The resulting mesh might be non-manifold (**red cell**) or have its topology changed (**blue cell**).

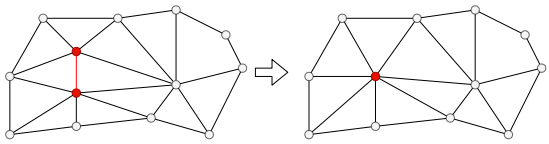

Figure 3. **Illustration of Quadric Error Metrics based edge collapse for mesh simplification.** The edge between two red vertices is collapsed and the resulting mesh is re-triangulated with its topology preserved.

As described in Section 3.5 of the main paper, to construct a mesh hierarchy for multi-level feature learning, we adopt two well-known mesh simplification methods from the geometry processing domain: Vertex Clustering (VC) [7] and Quadric Error Metrics (QEM) [3]. In order to facilitate readers' understanding, we prepare the illustrations of the two methods in Fig. 2 and Fig. 3.

C. Qualitative Visualization

In this section, we present more qualitative comparisons on the ScanNet [2] and Matterport3D [1] datasets. As shown in Fig. 6 and Fig. 7, our results are compared with those by SparseConvNet [4], which operates on the Euclidean domain solely and has a more complex network structure than VMNet. Our results generally show a better capacity of dealing with complex geometries, as well as produce less ambiguous features on spatially close objects.

D. More Complexity Comparisons

With the same settings as in paper L. 681-701, we report more complexity comparisons of our network against other representative methods in Table 1. While achieving the highest mIoU, VMNet is largely comparable to other representative methods, in terms of both inference time and parameter size.

Method	Conv Category	Params (M)	Latency (ms)	mIoU(%)
MVPNet[5]	2D-3D	24.6	95	64.1
PointConv[10]	PointConv	21.7	307	66.6
KPConv[9]	PointConv	14.1	52	68.4
DCM-Net[8]	GraphConv	0.76	151	65.8
VMNet (**Ours**)	Sparse+Graph Conv	17.5	107	**74.6**

Table 1. **Comparisons additional to Table 3 in the paper.**

E. Ablation: Multi-level Feature Refinement

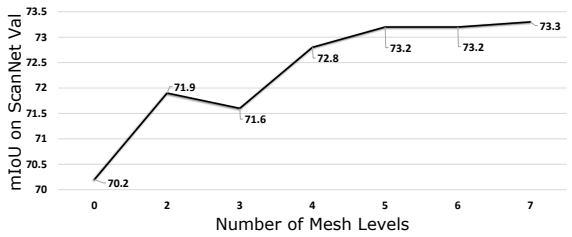

Figure 4. **Ablation study:** Multi-level feature refinement.

To measure the effects of individual geodesic feature refinement levels, we successively add the aggregation and fusion modules to the overall architecture. Except for the baseline with no geodesic branch, we start with the outermost mesh levels $\mathcal{M}^0\&\mathcal{M}^1$ to retain one fusion module and two aggregation modules. Next, along with each added mesh level, one fusion module and one aggregation module are added. The results are presented in Fig. 4. We witness that the first four levels bring the most performance gain, indicating the higher importance of finer-level meshes for geometric learning. We will add this experiment in the revision and explore networks focusing on fine levels in the future work.

F. Design Choice of Inter-domain Attention

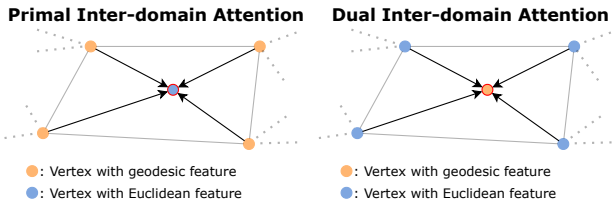

Figure 5. **Illustration of primal and dual inter-domain attention. (Left)** The primal inter-domain attention generates query vectors from the Euclidean features and aggregates the neighboring geodesic features. **(Right)** The dual inter-domain attention generates query vectors from the geodesic features and aggregates the neighboring Euclidean features.

As described in Section 3.4 of the main paper, we proposed an inter-domain attentive module for adaptive feature fusion. The module takes both the Euclidean features and the geodesic features as input and utilizes the attention mechanism, in which the attention weights are conditioned

on features from both the domains. To build such an inter-domain attentive module, there are two design choices. As shown in Fig. 5, we denote the one used in VMNet as the primal inter-domain attention and denote the other one as the dual inter-domain attention. We empirically find that the primal inter-domain attention yields better results than the dual one (73.3% vs 72.8% in mIoU on ScanNet Val). It may be caused by the different importance of the Euclidean features and the geodesic features in the task of indoor scene 3D semantic segmentation.

References

[1] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3D Vision (3DV)*, 2017. 1, 2, 5

[2] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1, 2, 4

[3] Michael Garland and Paul S Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 209–216, 1997. 2

[4] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9224–9232, 2018. 1, 2

[5] Maximilian Jaritz, Jiayuan Gu, and Hao Su. Multi-view pointnet for 3d scene understanding. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2

[6] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 1

[7] Jarek Rossignac and Paul Borrel. Multi-resolution 3d approximations for rendering complex scenes. In *Modeling in computer graphics*, pages 455–465. Springer, 1993. 2

[8] Jonas Schult, Francis Engelmann, Theodora Kontogianni, and Bastian Leibe. Dualconvmesh-net: Joint geodesic and euclidean convolutions on 3d meshes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8612–8622, 2020. 2

[9] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6411–6420, 2019. 2

[10] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9621–9630, 2019. 2

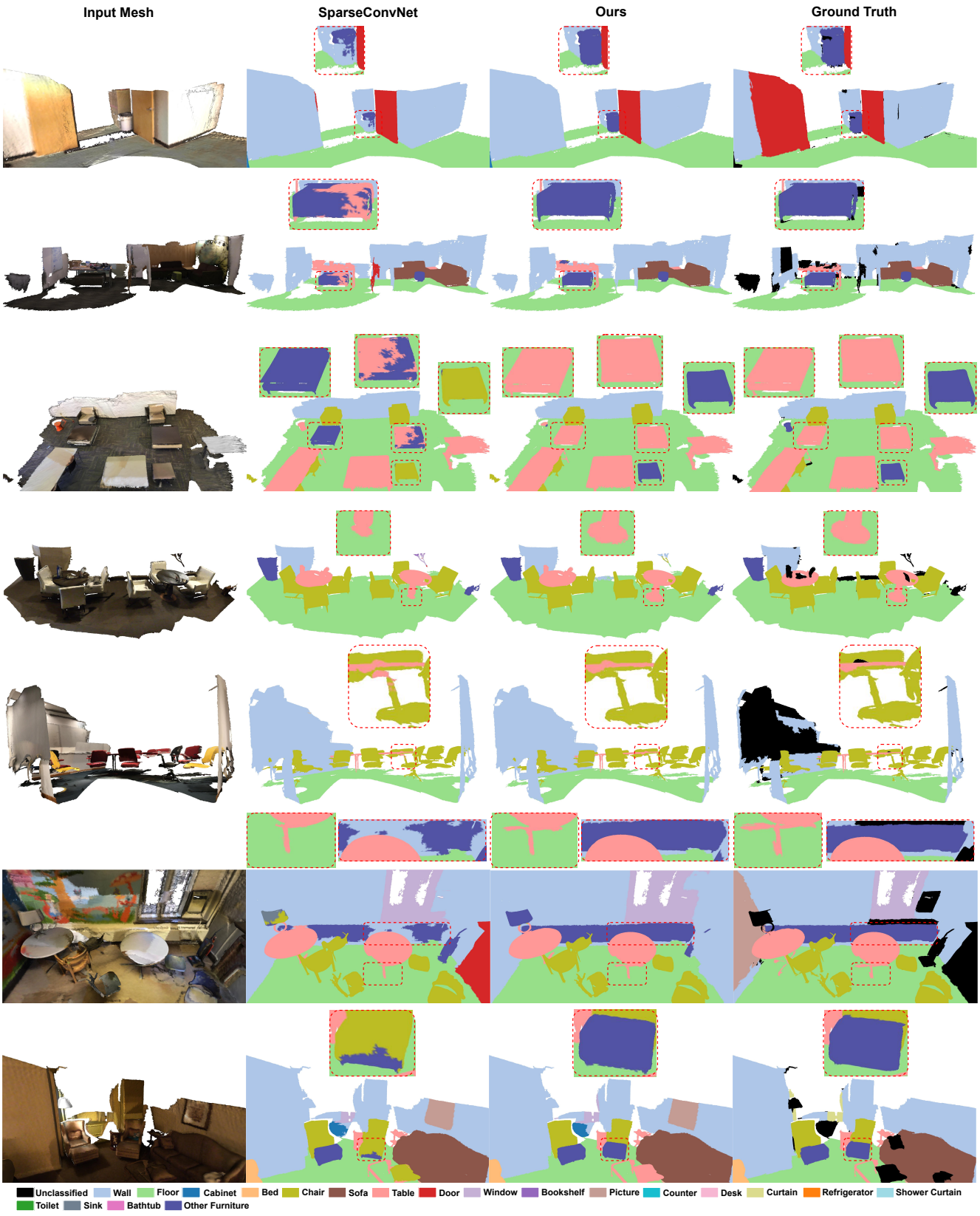

Figure 6. **More qualitative results on ScanNet Val [2].** The key parts for comparison are highlighted by dotted red boxes.

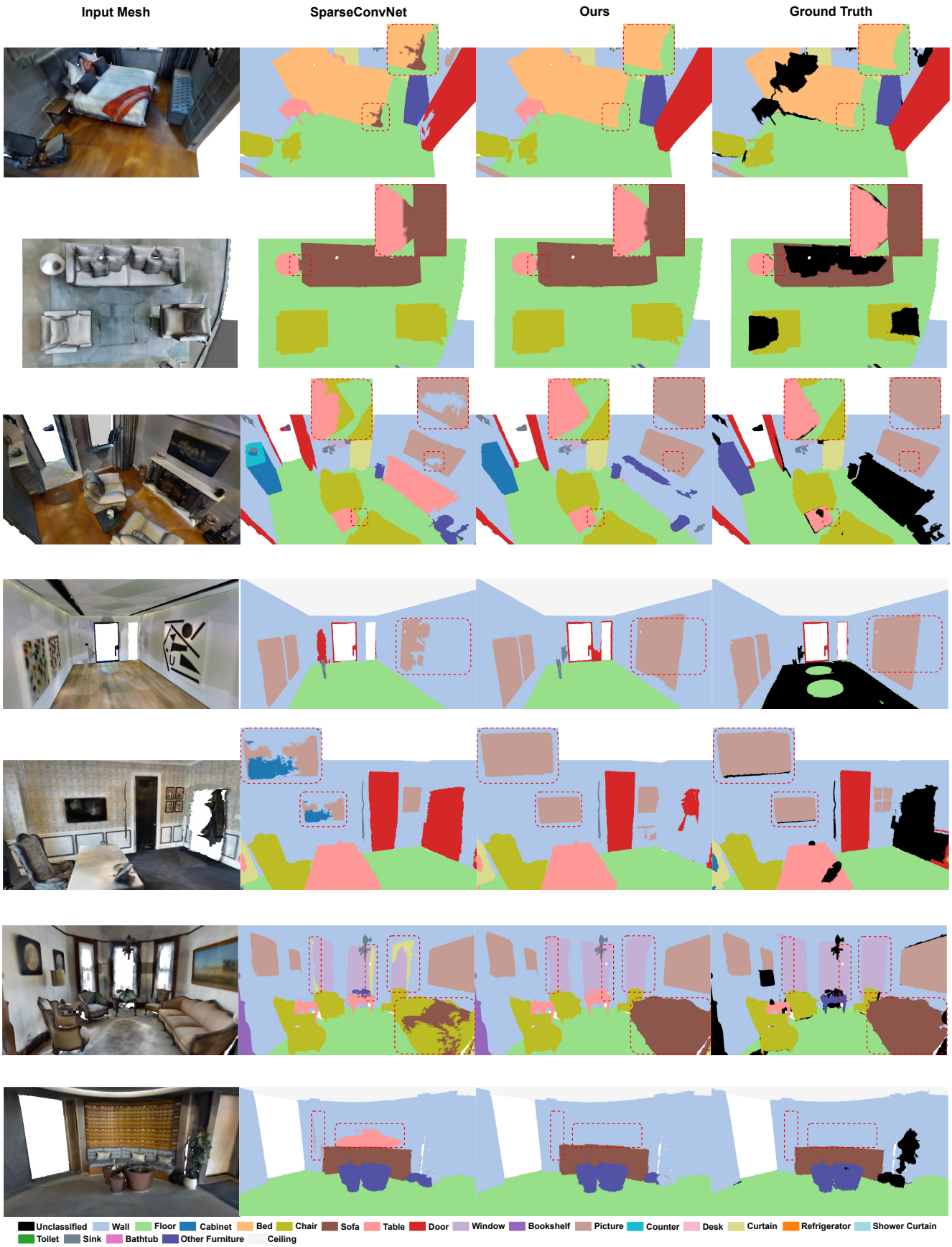

Figure 7. **Qualitative results on Matterport3D Test [1].** The key parts for comparison are highlighted by dotted red boxes.