CVPR Appendix for: Graph Attention Convolution for Point Cloud Segmentation

Anonymous CVPR submission

Paper ID 4649

A. Overview

This document provides additional details and further analysis of our proposed graph attention convolution (GAC) in the main paper. In Section B we describe more details on the network architectures and training parameters. Section C provides a proof of Theorem 1, and the further analysis of our GAC is shown in Section D. Finally, we show more visualizations of our point cloud segmentation results in Section E.

B. Network Architecture and Training Details

Segmentation Network. Our segmentation network is constructed on the graph pyramid of the point cloud. The input point cloud is first represented as a graph pyramid including 5 scales according to Section 3.3 of our main paper. The subsample ratios for graph coarsening are set to 4-4-4-2, i.e., the finest scale has 4096 vertices, then the coarser scales have 1024, 256, 64, and 32 vertices respectively. Therefore, our segmentation network consists of 9 layers, layers 1-5 consist of our GAC and the graph pooling operations, layers 6-9 consist of the feature interpolation and the skip connection modules. The output dimension of each layer is set to 64-128-256-512-256-256-256-128-128. All layers (except the last layer) are normalized with batch normalization and activated by the ReLU function.

Considering that the S3DIS and Semantic3D datasets contain objects of different sizes, the radius for neighbor searching at each scale for the S3DIS dataset are set to 0.1m, 0.2m, 0.4m, 0.8m, and 1.6m, while they are 0.2m, 0.4m, 0.8m, 1.6m, and 3.2m for the Semantic3D dataset.

Classification Network. The classification network in Section 4.4 of our main paper is built simply by replacing the feature interpolation layers of the segmentation network with a global pooling layer. The graph pyramid for classification contains only 4 scales as the relatively small number of sampling points on each CAD model. The subsample ratios for graph coarsening are 2-4-4, i.e., the finest scale has 1024 vertices, and the coarser scales have 512, 128, and 32 vertices respectively. The output dimension of each layer (including the fully connected layer) is 64-128-256-512-256 (as shown in Figure 1).

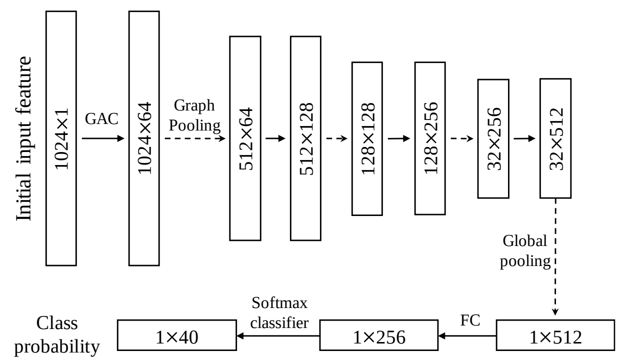

Figure 1. Our classification netework for ModelNet40 shape classification.

Data Augmentation. Before constructing the input point cloud into the graph pyramid, we augment the point cloud on-the-fly by randomly rotating the point cloud along the vertical axis and jittering the coordinates of each point by Gaussian noise $N(0, 0.01)$ truncated to [-0.05, 0.05].

Training Details. The networks are trained with the Adam optimizer and cross-entropy loss with an initial learning rate of 0.001 and momentum of 0.9. For the segmentation task on the S3DIS and Semantic3D datasets, the networks are trained with 50 epochs and batch size 16. For the classification task on the ModelNet40 dataset, the network is trained with 200 epochs and batch size 32.

C. Proof of Theorem 1

For proof convenience, we first prove two lemmas:

- Lemma 1 is a useful fact that any continuous function can be approximated by a multilayer perceptron with a single hidden layer to an arbitrary precision.
- Lemma 2 states that any Hausdorff continuous function can be approximated by the compound of a multilayer perceptron and mean function (similar to [2]).

Lemma 1. *Suppose $f : \mathbb{R}^F \to \mathbb{R}^K$, $K \in \mathbb{Z}$ is continuous function. $\forall \epsilon > 0$ and $x \in \mathbb{R}^F$, $\exists$ a multilayer perception M_{θ_ξ}, such that*

$$\|f(x) - M_{\theta_\xi}(x)\| < \epsilon$$

where θ_ξ is the parameters of multilayer perception M_{θ_ξ}.

Proof. Lemma 1 is a direct corollary of Theorem 2 in [1] to multi-output function. □

Next, we provide the proof of Lemma 2. Following Theorem 1, we denote $\mathcal{X} = \{S : S \subseteq [a, b]^F$ and S is finite $\}$, $f : \mathcal{X} \to \mathbb{R}$ is a continuous set function w.r.t Hausdorff distance $d_H(\cdot, \cdot)$. Then, $\forall \epsilon_1 > 0$, $\exists \delta > 0$, for any $S, S^{'} \in X$, if $d_H(S, S^{'}) < \delta$, we have $|f(S) - f(S^{'})| < \epsilon_1$.

Lemma 2. *Suppose $f : \mathcal{X} \to \mathbb{R}$ is a continuous set function w.r.t Hausdorff distance $d_H(\cdot, \cdot)$. $\forall \epsilon > 0$ and set $S \in \mathcal{X}$, $\exists$ a multilayer perception $M_{\theta_\xi} : \mathcal{X} \to \mathbb{R}^K$, $K \in \mathbb{Z}$, such that*

$$|f(S) - \gamma(\text{Mean}\{M_{\theta_\xi}(x) : x \in S\})| < \epsilon$$

where γ is a continuous function, and $\text{Mean}\{\cdot\}$ is a mean function that takes a set of vectors as input and returns a new vector of their element-wise average value.

Proof. Without loss generalization, we consider S as a one-dimensional finite set, i.e., $F = 1$. Denote $\Omega = [a, b]$, we can evenly split Ω into $K = \lceil \frac{b-a}{\delta} \rceil$ small intervals $[a+(k-1)\Delta, a+k\Delta]$, $k = 1, 2, ..., K$, where $\Delta = \frac{b-a}{K}$.

Define function $m(x) = a + \lfloor \frac{x-a}{\Delta} \rfloor \Delta$ maps x to the lower bound of the interval it lies in. Let $S^{'} = \{m(x) : x \in S\}$, then $|f(S) - f(S^{'})| < \epsilon_1$ as $d_H(S, S^{'}) < \frac{b-a}{K} < \delta$.

Let continuous function $\sigma_k = d_H(x, \Omega \backslash [a + (k-1)\Delta, a + k\Delta])$, and symmetric function $v_k(S) = \text{Mean}\{\sigma_k(x) : x \in S\}$. Denote $\boldsymbol{\sigma} = [\sigma_1, ..., \sigma_K]$ and $\mathbf{v} = [v_1, ..., v_K]$, the value of v_k indicates whether there are points lying in the interval $[a + (k-1)\Delta, a + k\Delta]$, $k = 1, 2, ..., K$.

Therefore, we further define a mapping function $\tau : [0, +\infty) \to \mathcal{X}$ as $\tau(v_k) = \{a + (k-1)\Delta : v_k > 0\}$. It maps the vector $\mathbf{v}$ to a set consisting of the lower bound of the split intervals, which is exactly equals to the set $S^{'}$ we constructed above, i.e., $\tau(\mathbf{v}(S)) = S^{'}$.

Let $\gamma : \mathbb{R}^K \to \mathbb{R}$ be a continious function so that $\gamma(\mathbf{v}) = f(\tau(\mathbf{v}))$, then we have

$$\begin{aligned}
&|f(S) - \gamma(\text{Mean}\{\boldsymbol{\sigma}(x) : x \in S\})| \\
=&|f(S) - f(\tau(\text{Mean}\{\boldsymbol{\sigma}(x) : x \in S\})| \\
=&|f(S) - f(\tau(\mathbf{v}(S)))| \\
=&|f(S) - f(S^{'})| < \epsilon_1
\end{aligned}$$

where

$$\begin{aligned}
&\gamma(\text{Mean}\{\boldsymbol{\sigma}(x) : x \in S\} \\
=&\gamma([\text{Mean}(\sigma_1(x) : x \in S), ..., \text{Mean}(\sigma_K(x) : x \in S)])
\end{aligned}$$

is a symmetric function which is independent of the order of the elements in set S.

Next, we show that the continuous function $\boldsymbol{\sigma}$ can be replaced by a multilayer perceptron. According to Lemma 1, we know that $\forall \epsilon_2 > 0$, $\exists$ a multilayer perception M_{θ_ξ}, such that $\|\boldsymbol{\sigma}(x) - M_{\theta_\xi}(x)\| < \epsilon_2$. Then, we have

$$\begin{aligned}
&\|\text{Mean}\{\boldsymbol{\sigma}(x) : x \in S\} - \text{Mean}\{M_{\theta_\xi}(x) : x \in S\}\| \\
=&\|\text{Mean}\{\boldsymbol{\sigma}(x) - M_{\theta_\xi}(x) : x \in S\}\| \\
<&|S|\epsilon_2
\end{aligned}$$

As S is a finite set, $\forall \delta_1 > 0$, $\exists$ ϵ_2, such that $|S|\epsilon_2 < \delta_1$. Therefore, according to the definition of a continuous function, $\forall \epsilon_3 > 0$, $\exists$ multilayer perception M_{θ_ξ}, such that

$$|\gamma(\text{Mean}\{\boldsymbol{\sigma}(x) : x \in S\}) - \gamma(\text{Mean}\{M_{\theta_\xi}(x) : x \in S\})| < \epsilon_3.$$

Then we have

$$\begin{aligned}
&|f(S) - \gamma(\text{Mean}\{M_{\theta_\xi}(x) : x \in S\})| \\
<&|f(S) - \gamma(\text{Mean}\{\boldsymbol{\sigma}(x) : x \in S\})| \\
&+ |\gamma(\text{Mean}\{\boldsymbol{\sigma}(x) : x \in S\}) - \gamma(\text{Mean}\{M_{\theta_\xi}(x) : x \in S\})| \\
<&\epsilon_1 + \epsilon_3
\end{aligned}$$

Let $\epsilon = \epsilon_1 + \epsilon_3$, we have

$$|f(S) - \gamma(\text{Mean}\{M_{\theta_\xi}(x) : x \in S\})| < \epsilon$$

□

We now restate Theorem 1 and provide its proof.

Theorem 1. *Suppose $f : \mathcal{X} \to \mathbb{R}$ is a continuous set function w.r.t Hausdorff distance $d_H(\cdot, \cdot)$. Denote $S_i = \{h_j : j \in \mathcal{N}(i)\} \in \mathcal{X}$ as the set of neighboring points of vertex $i \in V$ with arbitrary order. $\forall \epsilon > 0$, $\exists K \in \mathbb{Z}$ and parameter θ of GAC, such that for any $i \in V$,*

$$|f(S) - \gamma(g_\theta(S_i))| < \epsilon$$

where γ is a continuous function, and $g_\theta(S_i) \in \mathbb{R}^K$ is the output of our GAC.

Proof. We show that there exists parameter θ that can represent our GAC function g_θ as a mean operator (including the MLP in Lemma 2), then Theorem 1 can be proved according to Lemma 2.

As described in Section 3.1 of the main paper, the parameter θ of our GAC consists of two parts, i.e., $\theta = \{\theta_M, \theta_\alpha\}$

, θ_M is the parameter of the applied MLP for feature transformation and θ_α indicates the parameter of the attention mechanism. Obviously, the mean function is a special case of our attention mechanism when assigning even attentional weights to all the neighbors as $\frac{1}{|S_i|}$. In addition, let $\theta_M = \theta_\xi$ in Lemma 2, we have

$$\begin{aligned} g_\theta(S_i) &= \frac{1}{|S_i|} \sum_{x \in S_i} M_{\theta_\xi}(x) \\ &= \mathrm{Mean}\{M_{\theta_\xi}(x) : x \in S_i\} \end{aligned}$$

As $S_i \in \mathcal{X}$, according to Lemma 2, we have

$$\begin{aligned} &|f(S) - \gamma(g_\theta(S_i))| \\ =&|f(S) - \gamma(\mathrm{Mean}\{M_{\theta_\xi}(x) : x \in S_i\})| < \epsilon \end{aligned}$$

□

D. Further Analysis of GAC

The proof in Section C states that, in the worst case, we can convert the neighboring space into a volumetric representation. The accuracy of the volumetric representation is related to the output dimension K. In this section, we provide more analysis of the effect of the output dimension K on both our GAC and the the mean/max operator (including the MLP) [2].

Similar to Section C, we still consider a one-dimensional finite set $\{h_1, h_2, ..., h_M\}$ contains the $M > 1$ neighbos of vertex $i \in V$. When $K \geq M$, according to the proof of Theorem 1, there exists an MLP that maps each feature to a K-dimension feature space as $\{h_1^{'}, h_2^{'}, ..., h_M^{'}\} \in \mathbb{R}^K$, where $h_i^{'}$ is a K-dimension vector where the i-th element equals h_i and the rest equal zero. Then, the outputs of the mean/max operator and GAC are $o_{mean} = \frac{1}{M}[h_1, h_2, ..., h_M, 0, ...]$, $o_{max} = [h_1, h_2, ..., h_M, 0, ...]$, and $o_{gac} = [\alpha_1 h_1, \alpha_2 h_2, ..., \alpha_M h_M, 0, ...]$ respectively, where α_i is the attentional weight of GAC. In this condition, both of them can entirely encode the input information and reconstruct them.

When $K < M$, e.g., $K = 1$. Then $\{h_1^{'}, h_2^{'}, ..., h_M^{'}\} \in \mathbb{R}$, $h_i^{'} \in \mathbb{R}$ is a one-dimensional value. In this case, the outputs of the mean/max operator and our GAC are $o_{mean} = \frac{1}{M}\sum_{i=1}^{M} h_i^{'}$, $o_{max} = \mathrm{Max}\{h_1^{'}, h_2^{'}, ..., h_M^{'}\}$, and $o_{gac} = \sum_{i=1}^{M} \alpha_i h_i^{'}$. It can be seen that neither the max nor the mean operator can reconstruct the input information. However, the attentional weight α_i of our GAC is dynamically generated by the attention mechanism $\alpha(p_j - p_i, h_j^{'} - h_i^{'})$. Without loss generalization, considering α as a linear system, we have

$$\begin{cases} w_1 h_1^{'} - w_1 h_i^{'} + b_1 = \alpha_1 \\ w_2 h_2^{'} - w_2 h_i^{'} + b_2 = \alpha_2 \\ \qquad \cdots \\ w_M h_M^{'} - w_M h_i^{'} + b_M = \alpha_M \end{cases}$$

, where w_i is the learned weights and b_i is a added term corresponding to $p_j - p_i$, which is independent of $\{h_1^{'}, h_2^{'}, ..., h_M^{'}\}$. Denote the weight matrix

$$W = \begin{pmatrix} w_1 & 0 & \cdots & -w_1 & \cdots & 0 \\ 0 & w_2 & \cdots & -w_2 & \cdots & 0 \\ 0 & 0 & \cdots & -w_3 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -w_M & \cdots & w_M \\ \alpha_1 & \alpha_2 & \cdots & \alpha_i & \cdots & \alpha_M \end{pmatrix}$$

, $c = [\alpha_1 - b_1, ..., \alpha_M - b_M, o_{gac}]^T$, $h = [h_1^{'}, h_2^{'}, ..., h_M^{'}]^T$. The input information of our GAC can be reconstructed as $h = W^\dagger c$, where $W^\dagger$ is the pseudo-inverse matrix of W. The attention mechanism of our GAC acts as an encoder which maps the neighboring features into the attentional weight space. Thus, our GAC is capable of representing the entire neighboring information even though the output dimension K is not sufficiently large.

Notably, the max and mean operator can be seen as two special cases of our GAC as "max attention" and "mean attention" respectively. The max operator tends to capture the most "special" points, while the mean operator is their average description blurring the valuable points. Both of them damage the structural connections between points of an object and result in poor object delineation. Comparatively, our proposed GAC aggregates the information by assigning the neighboring points specific attentional weights, maintaining the structure of the objects which is helpfull towards fine-grained segmentation of point cloud.

E. More Visualizations

In this section, we provide more qualitative segmentation results on the S3DIS and Semantic3D datasets. For the S3DIS dataset, we show our segmentation results from five different types of rooms, their corresponding input data and the ground truth in Figure 2. For the Semantic3D dataset, due to the lack of public ground truth for the testing sets, we only provide the input data and our segmentation results in Figure 3.

References

[1] K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, pages 251–257, 1991. 2

[2] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, pages 77–85, 2017. 1, 3

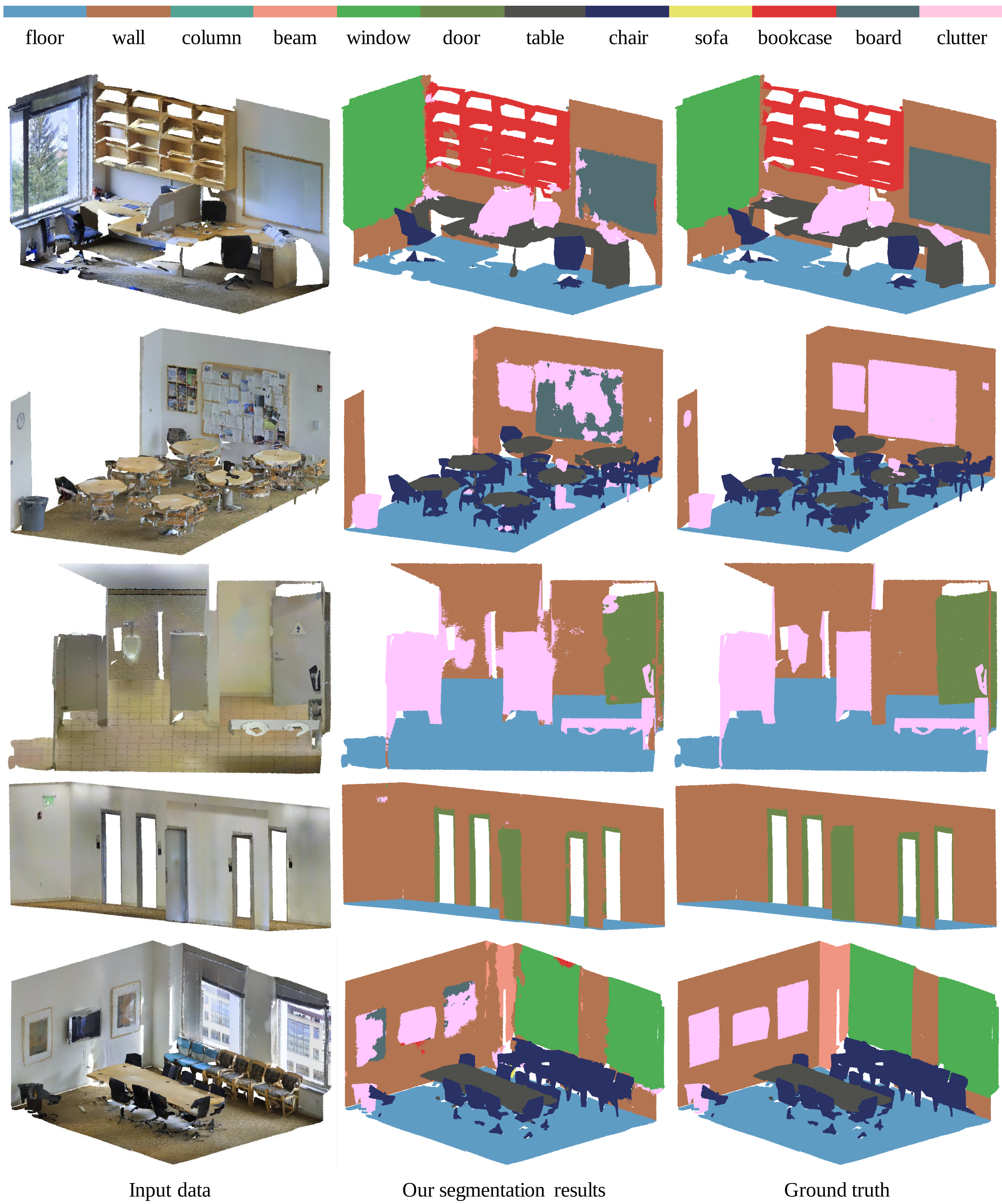

Figure 2. Example visualizations on the S3DIS dataset. The first column is the input point cloud, the second and third columns represent our segmentation results and the ground truth. The ceiling and part of the wall are removed for visualization convenience. We can see that the board is easily confused with the cluster which includes some posters and papers. In addition, the column which has no significant color and local feature difference is also difficult to predict.

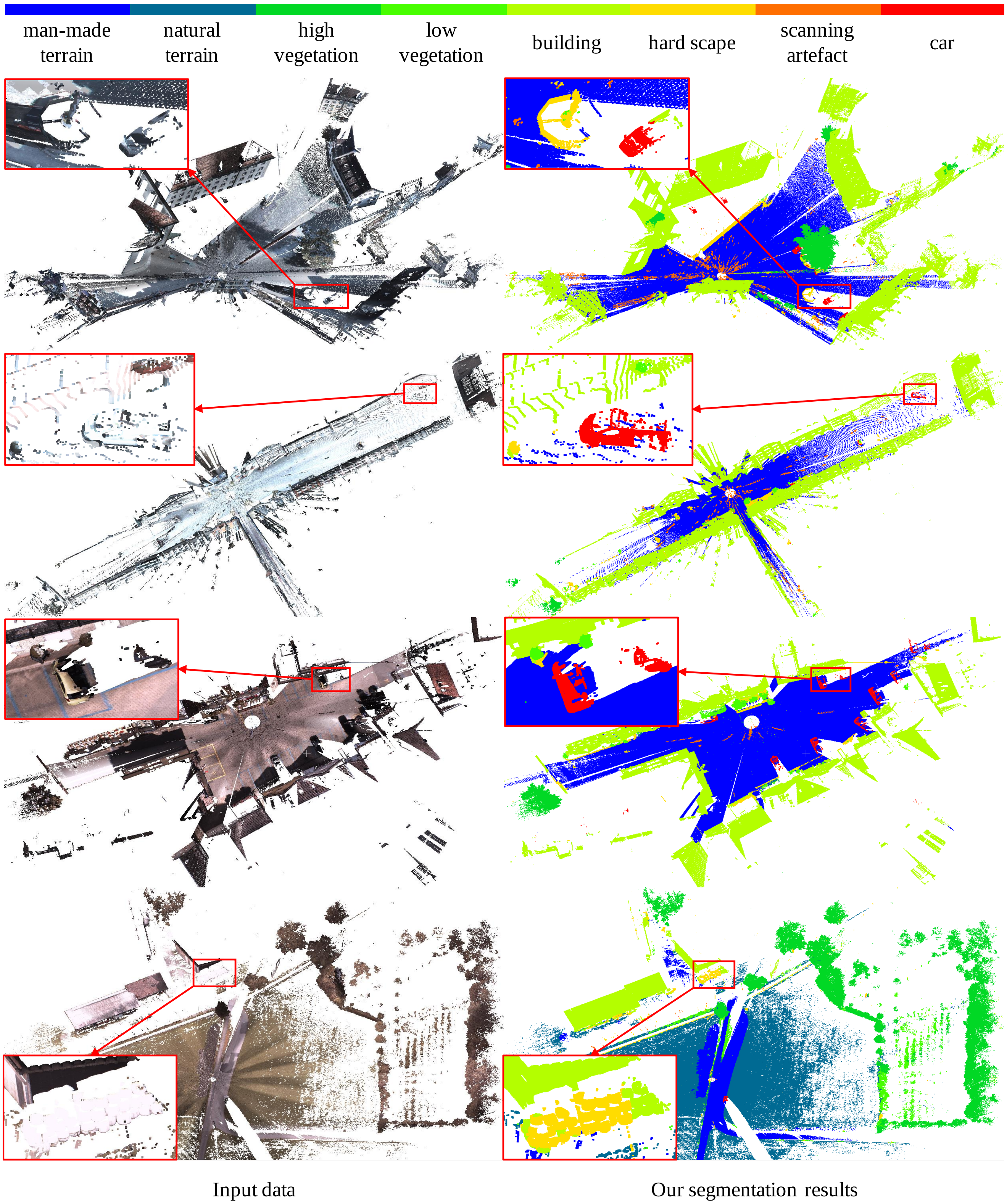

Figure 3. Segmentation results on the Semantic3D dataset. The first column is the input point cloud, and the second column represents our segmentation results. The hard scape is easily confused with the buildings as they include similar artificial signs.