A. Appendix

A.1. Simplified Code

We provide a simplified version of the codebase for assessing the efficacy of MergeOcc, which is accessible in https://github.com/Easonxu-01/MergeOcc_simple. By strictly adhering to the guidelines provided in the Readme.md file, users can train a generalized occupancy prediction model showcasing outstanding performance on both the SemanticKITTI dataset [1] and the OpenOccupancy-nuScenes dataset [21].

A more comprehensive version of the codebase will be released in the near future.

A.2. Experiments Settings

A.2.1. Model Architecture

As detailed in the Sec. 4, We use a 3D voxel-based method L-CoNet, whose backbone is ResNet3D and a 2D projection-based method PointOcc, whose backbone is the Swin Transformer. The input points are configured to perform cylindrical partitioning with dimensions $(\mathcal{H}_{\text{in}}, \mathcal{W}_{\text{in}}, \mathcal{D}_{\text{in}}) = (512, 360, 32)$, representing radius, angle, and height, respectively. Notably, to maintain the fairness of experiments, we refrain from adjusting any model's parameters. All training processes are kept as originally intended, meaning we use **the identical model architecture and hyper-parameter set**. Additionally, since PointOcc does not provide available checkpoints, the reproduced results from our own implementation are reported. The occupancy head produces a voxel representation of dimensions $(\mathcal{H}_{\text{out}}, \mathcal{W}_{\text{out}}, \mathcal{D}_{\text{out}}) = (128, 90, 10)$. To enhance performance, we adopt a coarse-to-fine query strategy to upsample the output by a factor of $s = 4$, following the methodology of L-CONet.

A.2.2. Task Description and General Settings

The 3D semantic occupancy prediction has garnered significant attention for autonomous driving, necessitating the assignment of semantic labels to all regions within the spatial domain. Our evaluation is conducted on two widely recognized datasets for autonomous driving occupancy prediction: SemanticKITTI and OpenOccupancy-nuScenes. In SemanticKITTI, the dataset's perceptive field spans the range from $[-72.0m, -72.0m, -3.4m]$ to $[72.0m, 72.0m, 3m]$. However, the annotated region is restricted from $[0m, -25.6m, -3.4m]$ to $[51.2m, 25.6m, 3m]$, with a voxel resolution of $0.2m$. This yields a volumetric representation of $256 \times 256 \times 32$ voxels for occupancy prediction. In contrast, OpenOccupancy-nuScenes spans its perceptual range from $[-51.2m, -51.2m, -5m]$ to $[51.2m, 51.2m, 3m]$, maintaining the same voxel resolution of $0.2m$, resulting in a volumetric grid of $512 \times 512 \times 40$ voxels. Our analysis is confined to the intersection of ground truth ranges, as delineated in Sec. 3.3, specifically from $[0m, -25.6m, -3.4m]$ to $[51.2m, 25.6m, 3m]$.

A.2.3. The Metric of Occupancy Prediction

For 3D semantic occupancy prediction, we use the intersection over the union (**IoU**) of occupied voxels, ignoring their semantic class as the evaluation metric of the scene completion (SC) task and the **mIoU** of all semantic classes for the semantic scene completion (SSC) task.

$$\begin{aligned} \text{IoU} &= \frac{TP}{TP + FP + FN} \\ \text{mIoU} &= \frac{1}{C} \sum_{i=1}^{C} \frac{TP_i}{TP_i + FP_i + FN_i} \end{aligned} \tag{8}$$

where TP, FP, FN indicate the number of true positive, false positive, and false negative predictions. C is the class number.

A.2.4. Optimization, Training and Testing

Notably, throughout all our experiments, we retained the hyper-parameters from the two baselines without modification. This decision partially demonstrates the superior performance of our method.

During training on both datasets, we employ the Adam optimizer complemented by a weight decay of 0.01. A cosine learning rate scheduler initiates with a peak value of $3e^{-4}$, accompanied by a linear warm-up phase for the initial 500 iterations. The occupancy prediction leverages a combination of classic cross-entropy loss, Lovasz-softmax loss [2], and an affinity loss to optimize the geometry and semantic metrics [5] concurrently.

All experimental procedures were executed using the mmdetection3d framework. All models undergo training for 24 epochs, with a batch size of 8, distributed across 8 RTX 3090 GPUs. For the OpenOccupancy-nuScenes benchmark, we utilize multiple (10) LiDAR sweeps as input, adhering to a widely accepted practice. Conversely, for the SemanticKITTI dataset, a single LiDAR sweep is utilized as input.

Moreover, to ensure gradient backward propagation through all network parameters and sustain the training process, we design a balanced distributed group sampler that integrates data from distinct datasets in each batch. To the best of our knowledge, MergeOcc pioneers the application of the scene completion task on mmdetection3d and explores the MDT paradigm within this framework.

A.3. Further Ablation about R.A.

Additionally, ablation experiments indicate that the point cloud range alignment (R.A.) module plays an important role in improving the performance of MergeOcc. To elucidate the underlying causes, we conducted additional ablation experiments focusing on R.A., with results detailed in

Trained on	Method	Tested on OpenOccupancy-nuScenes		Tested on SemanticKITTI	
		Geometric IoU	Semantic mIoU	Geometric IoU	Semantic mIoU
SK+OO-nu	MergeOcc-V (G.A. for all)	37.0	21.2	68.7	14.2
	MergeOcc-V (G.A. for backbone)	38.4(+1.4)	21.4(+0.2)	69.9(+1.2)	14.5(+0.3)

Table 6. Experimental results of different scopes of geometric alignment.

Tab. 7. These findings demonstrate that R.A. is beneficial exclusively within the Multiple Datasets Training (MDT) paradigm. Conversely, under the single dataset training paradigm, R.A. is detrimental due to inherent data limitations. The principal performance improvements are attributed to the **learning afforded by larger and more diverse datasets**, with R.A. **serving as the bridge** to address existing gaps. This outcome highlights the superiority and critical importance of the MDT paradigm.

Model	Dataset	IoU	mIoU
L-CoNet (baseline)	OO-nu	30.9	15.8
MergeOcc-V (w/ `R.A.`)	OO-nu	27.4(−3.5)	12.5(−3.3)
MergeOcc-V (w/ `R.A.`)	Both	38.2(+7.3)	21.3(+5.5)

Table 7. Futher ablation experiments about R.A.

A.4. Scope of Using Geometric Alignment

Additionally, We investigate the optimal range for geometric statistical data alignment, that is, the utilization range of the dataset-specific norm layer. We categorize the settings into two distinct configurations: (a) employing geometric alignment only at the backbone and (b) substituting all norm layers in the network with geometric alignment. The experimental results are delineated in Tab. 6, prompting us to advocate for applying geometric alignment solely on the backbone of the network.

A.5. Coarse to Fine Stage

We use the geometric coarse to fine query to upsample the initial output, inspired by OpenOccupancy [21].

Specifically, the coarse occupancy $O^{\mathcal{M}} \in \mathbb{R}^{\frac{D}{S}\times\frac{H}{S}\times\frac{W}{S}\times c}$ is first generated by the baseline model, where the occupied voxels $V_{\mathrm{o}} \in \mathbb{R}^{N_{\mathrm{o}}\times 3}$ (N_{o} is the number of occupied voxels, and 3 denotes the (x, y, z) indices in voxel coordinates) are split as high-resolution occupancy queries $Q_{\mathrm{H}} \in \mathbb{R}^{N_{\mathrm{o}}8^{\eta-1}\times 3}$:

$$Q_{\mathrm{H}} = \mathcal{T}_{\mathrm{v}\to\mathrm{w}}(\mathcal{F}_{\mathrm{s}}(V_{\mathrm{o}}, \eta)), \tag{9}$$

where $\mathcal{F}_{\mathrm{s}}$ is the voxel split function (i.e., for (x_0, y_0, z_0) in V_{o}, the split indices are $\{x_0 + \frac{i}{\eta}, y_0 + \frac{j}{\eta}, z_0 + \frac{k}{\eta}\}(i, j, k \in (0, \eta - 1))$), η is the split ratio (typically set as 4), and $\mathcal{T}_{\mathrm{v}\to\mathrm{w}}$ transforms the voxel coordinates to the world coordinates. Subsequently, we transform Q_{H} to voxel space to sample geometric features $F^{\mathcal{G}} = \mathcal{G}_{\mathrm{S}}(F^{\mathcal{F}}, \mathcal{T}_{\mathrm{w}\to\mathrm{v}}(Q_{\mathrm{H}}))$ ($\mathcal{G}_{\mathrm{S}}$ is the *grid sample* function, $\mathcal{T}_{\mathrm{w}\to\mathrm{v}}$ is the transformation from world coordinates to voxel coordinates). FC layers then regularize the sampled features to produce fine-grained occupancy predictions:

$$O^{\mathrm{g}} = \mathcal{G}_f(\mathcal{G}_f(F^{\mathcal{G}})), \tag{10}$$

where $F^{\mathcal{G}}$ are FC layers. Finally, O^{g} can be reshaped to the volumetric representation $O^{\mathrm{vol}} \in \mathbb{R}^{\frac{\eta D}{S}\times\frac{\eta H}{S}\times\frac{\eta W}{S}\times c}$:

$$O^{\mathrm{vol}}(x, y, z) = \begin{cases} O^{\mathrm{g}}(\mathcal{T}_{\mathrm{v}\to\mathrm{q}}(x, y, z)) & (x, y, z) \in \mathcal{T}_{\mathrm{w}\to\mathrm{v}}(Q_{\mathrm{H}}) \\ \text{Empty Label} & (x, y, z) \notin \mathcal{T}_{\mathrm{w}\to\mathrm{v}}(Q_{\mathrm{H}}), \end{cases} \tag{11}$$

where $\mathcal{T}_{\mathrm{v}\to\mathrm{q}}$ transforms the voxel coordinates to indices of the high-resolution query Q_{H}. For LiDAR-based CONet that without multi-view 2D features, we only sample Q_{H} from $F^{\mathcal{L}}$.

A.6. More Illustration about Semantic Label Mapping

A.6.1. Semantic Level differences

Different datasets have obvious semantic disparities. For instance, SemanticKITTI distinguishes static and dynamic objects (**e.g.**, 'car' vs. 'moving-car'), which is crucial for scene understanding. Furthermore, it aggregates certain object categories into broader classes (**e.g.**, 'other-vehicle' encompasses buses and rail vehicles) and details categories for road infrastructure ('road', 'parking', 'sidewalk') and natural elements ('vegetation', 'terrain'). Conversely, nuScenes [3] focuses on distinguishing various vehicle types (**e.g.**, 'car', 'truck', 'bus') and explicitly categorizes urban infrastructure elements such as 'traffic-cone'. It also clearly segregates 'driveable surface' from 'sidewalk', establishing a distinct boundary between zones that are navigable by vehicles and those meant for pedestrian use.

Besides, the distribution of point clouds varies significantly across datasets, owing to their collection from diverse geographical locales via different LiDARs, as depicted in Tab. 1. Such heterogeneity engenders pronounced disparities in road topographies and object dimensions, as exemplified by the visualizations presented in Fig. 5 and Fig. 6.

Hence, semantic discrepancies across different datasets, stemming from varying class definitions and annotation granularity, present significant challenges in the MDT paradigm. Specifically, multi-head models may yield duplicate outputs for identical objects that appear in multiple

datasets, leading to uncertainty and redundancy that negatively impact downstream tasks. Therefore, SLM is necessary for aligning initial outputs to a unified label space in the MDT paradigm.

A.6.2. Computation of Label Space Learning Algorithm

The size of our optimization problem scales linearly in the number of potential merges $|\mathbb{T}|$, which can grow exponentially in the number of datasets. To counteract this exponential growth and mitigate the complexity, we propose a greedy algorithm that only considers sets of classes:

$$\mathbb{T}' = \left\{ \boldsymbol{t} \in \mathbb{T} \middle| \frac{c\boldsymbol{t}}{|\boldsymbol{t}|-1} \leq \tau \right\}.$$

For an appropriately aggressive threshold τ, the number of potential merges $|\mathcal{T}'|$ remains manageable. We greedily grow $\mathcal{T}'$ by first enumerating all feasible two-class merges ($|\boldsymbol{t}| = 2$), then three-class merges, and so on. The detailed algorithm diagram is shown in Algorithm 1. The time complexity of this algorithm is $O(|\mathcal{T}'| \max_i |\hat{L}^i|)$, which may remain computationally feasible even as the number of datasets increases.

A.6.3. Sequential Paradigm to Add New Datasets to the Unified Label Space

While the aspiration is to maintain large and comprehensive training domains and label spaces, practical scenarios often necessitate the inclusion of more fine-grained labels or specific testing domains. Upon establishing a unified label space from an existing set of training datasets, a straightforward label space expansion algorithm is employed to facilitate the addition of further datasets and labels after the unified model has been trained.

We adopt a sequential optimization paradigm. Specifically, we execute the unified model on the already merged dataset and train a domain-specific perception model on the new domain. Subsequently, the label space learning algorithm is applied to generate a new unified label space. This approach mitigates the computational overhead associated with merging more than two datasets.

Algorithm 1: Learning a unified label space

Input : $\{\mathbf{o}_i, \hat{\mathbf{l}}_i\}_{i=1}^N$: semantic occupancy grids ground truth and labels for each of the N training datasets
$\{\{\tilde{\mathbf{o}}_i^{(j)}, \tilde{\mathbf{l}}_i^{(j)}\}_{j=1}^N\}_{i=1}^N$: predicted semantic occupancy grids with predicted classes in all datasets for each training dataset
λ, τ: hyper-parameters for algorithm

Output : L: unified label space
$\mathcal{T}$: the transformation from each individual label space to the unified label space

// Compute potential merges and merge cost
$\hat{L} = \bigcup_i \hat{L}_i$ // Short-hand used to simplify notation
$\mathbb{T}_1 \leftarrow \{(l) | l \in \hat{L}\}$ // Set of single labels
Compute $c_{\boldsymbol{t}}$ for all single labels $\boldsymbol{t} \in \mathbb{T}$. // 0 for most metrics
for $n = 2 \ldots N$ **do**
  $\mathbb{T}_n \leftarrow \{\}$
  for $\boldsymbol{t} \in \mathbb{T}_{n-1}$ **do**
    for $l \in \hat{L}$ **do**
      if *l and all labels in $\boldsymbol{t}$ are from different datasets* **then**
        compute $c_{\boldsymbol{t} \cup \{l\}}$.
        if $\frac{c_{\boldsymbol{t} \cup \{l\}}}{n-1} \leq \tau$ **then**
          Add $\boldsymbol{t} \cup \{l\}$ to $\mathbb{T}_n$.
        end
      end
    end
  end
end
$\mathbb{T} \leftarrow \bigcup_{n=1}^N \mathbb{T}_n$
// Solve the ILP.
$\boldsymbol{x} \leftarrow$ ILP_solver$(c, \mathbb{T}, \lambda)$ // Solve equation (8).
Compute $L, \mathcal{T}$ from $\boldsymbol{x}$
Return: $L, \mathcal{T}$

A.7. Challenges of Merging Datasets

A.7.1. Dataset Introduction.

Following the practice of popular scene completion models [21–23] , our experiments are conducted on two prominent LiDAR semantic occupancy prediction benchmarks, namely, OpenOccupancy-nuScenes [21] and SemanticKITTI [1].

SemanticKITTI comprises annotated outdoor LiDAR scans with 21 semantic labels, organized into 22 point cloud sequences. Sequences 00 to 10, 08, and 11 to 21 are designated for training, validation, and testing, respectively. From these, 19 classes are selected for training and evaluation after merging classes with distinct motion statuses and removing classes with sparse points.

As for OpenOccupancy-nuScenes, it is a large-scale occupancy prediction dataset deriving from nuScenes. Since the 3D semantic and 3D detection labels are unavailable in the test set, Wang et al. [21] did not provide dense occupancy labels of the unseen test set. Consequently, we utilize the training set for model training and the validation set for evaluation purposes.

A.7.2. Primary Aspiration and Challenges

In the quest for highly intelligent automated vehicles, scalability and generalizability emerge as pivotal characteristics in perception models. According to the scaling law, data

plays a crucial role in augmenting both performance and generalizability.

However, acquiring vehicle travel data poses significant challenges compared to other forms of visual or textual data. Particularly, obtaining 3D LiDAR data entails substantial expenses. Consequently, existing autonomous driving datasets exhibit limited data volumes, hindering the training of models with requisite scalability and generalization. The conventional single dataset training-and-testing paradigm confines the source data within a delimited domain. Fully exploiting all available 3D data holds promise for mitigating resource expenditure and enhancing performance.

Initially, we have made a lot of attempts to train a vanilla 3D perception model using multiple datasets by directly merging existing 3D datasets, such as merging OpenOccupancy-nuScenes [21] and SemanticKITTI [1]. However, we found that commonly employed 3D perception models failed to perform satisfactorily across both datasets, as shown in Fig. 1. This inadequacy stems from the substantial disparities inherent in 3D point clouds acquired by diverse LiDARs, as shown in Tab. 1, rendering previous 3D models incapable of effectively addressing the significant data shift.

Furthermore, in the architectural design of the models to accommodate diverse label spaces, the incorporation of multiple heads within the network is deemed imperative. To facilitate gradient backward propagation across all network parameters and sustain the training process, a balanced distributed group sampler has been designed. This sampler amalgamates data from disparate datasets within each batch, thereby ensuring that data for each batch is sequentially drawn from distinct datasets.

A.8. Extended Visualization of Occupancy Prediction Results

A visual comparison of the occupancy prediction results generated by the primary methods is illustrated in Fig. 5 and Fig. 6.

For Fig. 5 and Fig. 6, we utilize the proposed MergeOcc-V model trained jointly on OpenOccupancy-nuScenes and SemanticKITTI datasets and showcase the outcomes on the validation sets of these two datasets. These results comprehensively demonstrate our ability to achieve improved occupancy prediction simultaneously for OpenOccupancy-nuScenes and SemanticKITTI datasets using a single perception model.

The proposed method produces results that more closely align with the ground truth in both structural layout and semantic consistency, capturing diverse semantic elements such as roads, vegetation, and buildings with greater clarity and more accurate category boundaries. Compared to L-CoNet and the D.M. method, it achieves a more complete and continuous representation of the scene and performs well simultaneously on both types of LiDAR.

Furthermore, owing to the denser ground truth annotations provided by the SemanticKITTI dataset, the outcomes yielded by MergeOcc on the OpenOccupancy-nuScenes dataset manifest heightened credibility compared to the ground truth in certain areas, such as the drivable surface and the trunk.

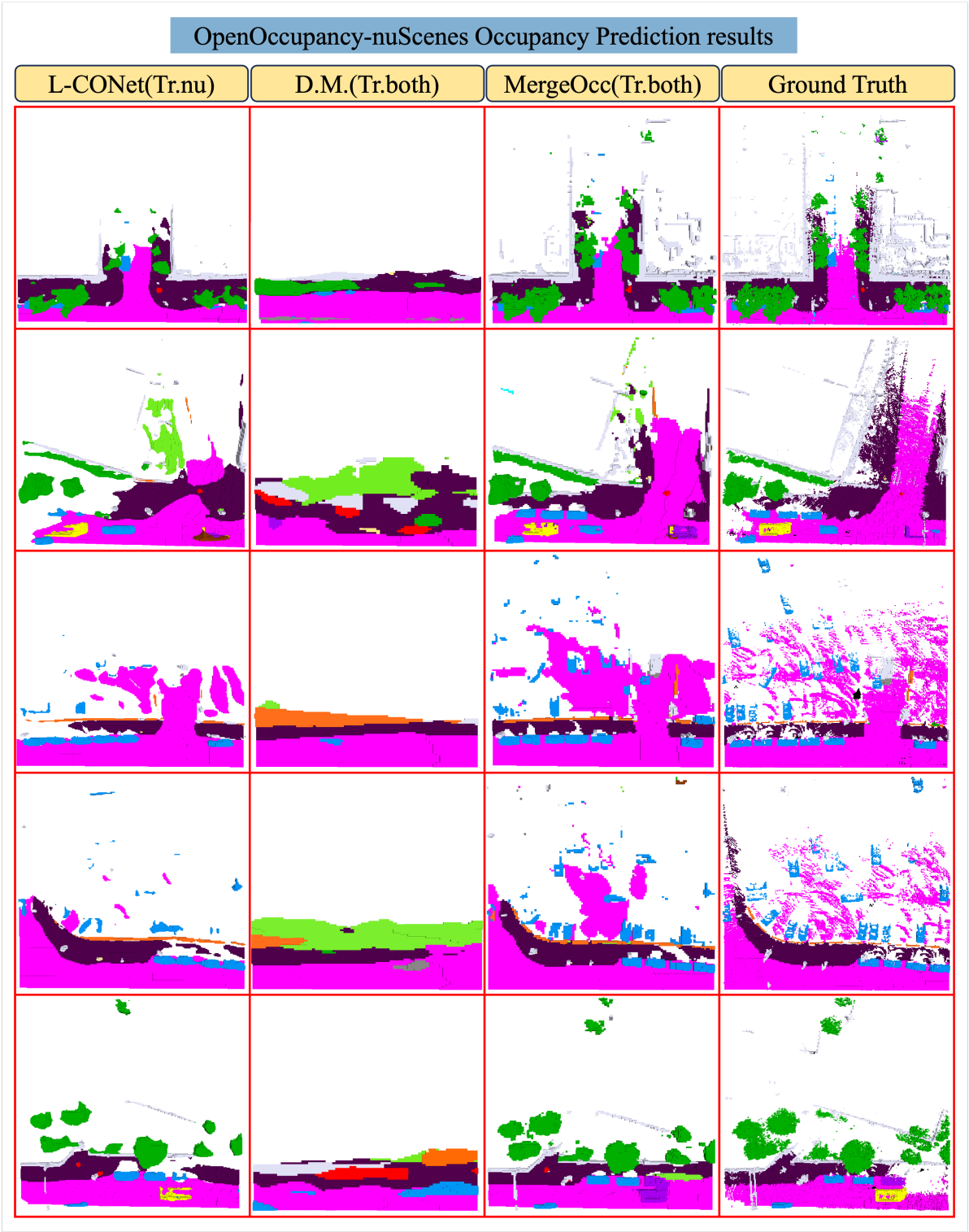

Figure 5. Visualizations of occupancy prediction results on OpenOccupancy-nuScenes.

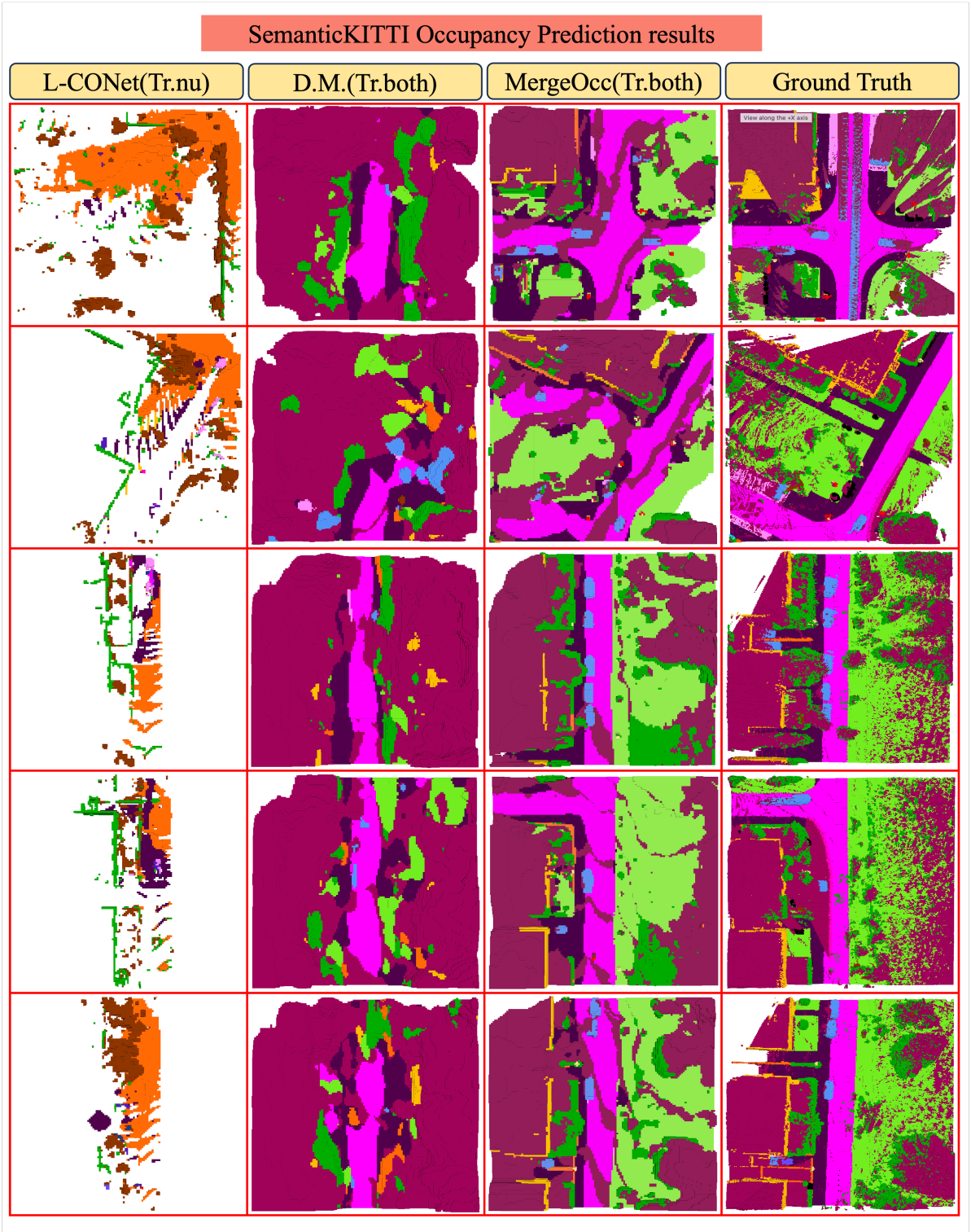

Figure 6. Visualizations of occupancy prediction results on SemanticKITTI.