28^{th} *Computer Vision Winter Workshop*
Graz, Austria, February 12–14, 2025

DOI: 10.3217/978-3-99161-022-9-001

A Data-Centric Approach to 3D Semantic Segmentation of Railway Scenes

Nicolas Münger Max Ronecker Xavier Diaz Michael Karner
SETLabs Research GmbH
Elsenheimerstraße 55, 80687 München, Germany
{firstname.lastname}@setlabs.de

Daniel Watzenig
Graz University of Technology
Inffeldgasse 16/II, 8010 Graz, Austria
daniel.watzenig@tugraz.at

Jan Skaloud
EPFL - Swiss Federal Technology Institute of Lausanne
GR A2 392 (Bâtiment GR) , 1015 Lausanne , Switzerland
jan.skaloud@epfl.ch

Abstract

LiDAR-based semantic segmentation is critical for autonomous trains, requiring accurate predictions across varying distances. This paper introduces two targeted data augmentation methods designed to improve segmentation performance on the railway-specific OSDaR23 dataset. The person instance pasting method enhances segmentation of pedestrians at distant ranges by injecting realistic variations into the dataset. The track sparsification method redistributes point density in LiDAR scans, improving track segmentation at far distances with minimal impact on close-range accuracy. Both methods are evaluated using a state-of-the-art 3D semantic segmentation network, demonstrating significant improvements in distant-range performance while maintaining robustness in close-range predictions. We establish the first 3D semantic segmentation benchmark for OSDaR23, demonstrating the potential of data-centric approaches to address railway-specific challenges in autonomous train perception.

1. Introduction

Rail transport offers a sustainable alternative to other transportation modes, emitting significantly lower carbon emissions [11]. Its continued development, especially through autonomous train operation (ATO), is critical to achieving climate goals like those in the European Union's Green Deal. ATO, defined from GoA0 (manual) to GoA4 (fully automated) [24], addresses labor shortages, increases operational flexibility and reliability, and optimizes service frequency. The Lausanne metro M2 line, a GoA4 system, demonstrates these benefits through higher frequency and adaptability. However, while fully automated systems work well in controlled settings, such as metro lines, implementing GoA3–4 in open rail networks is challenging due to unpredictable obstacles and the absence of physical barriers. Ensuring safety in open rail ATO is therefore a key research area.

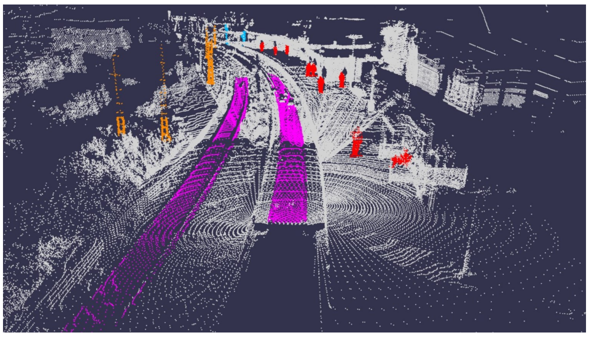


Figure 1. Example of a segmented pointcloud from the OSDaR23 dataset [30]

Robust perception systems are essential for obstacle detection and hazard identification in ATO. LiDAR (Light Detection and Ranging) suits these tasks by providing rich 3D geometric information [21]. LiDAR semantic segmentation, assigning a class to each 3D point, enables detailed environmental understanding. In autonomous driving, 3D object detection [6, 7, 18, 23, 25, 26] and semantic segmentation [1, 17, 19, 27, 38] are well-studied across many modalities. However, applying these techniques to autonomous train operation has received less attention, partly due to limited public datasets. The OSDaR23 dataset [30] addresses this gap by providing data for various railway perception tasks (Fig. 1). This paper applies deep learning-based 3D semantic segmentation to LiDAR point clouds in the railway domain using OSDaR23. We focus on safety-critical classes, emphasizing long-range segmentation accuracy due to trains' substantial braking distances. We also adopt a data-centric approach, introducing domain-specific data augmentations to improve robustness and performance.

Contributions

This paper introduces targeted data augmentation methods for LiDAR semantic segmentation in the railway domain, evaluated on the real-world OSDaR23 dataset.

1. Comprehensive evaluation of a state-of-the-art 3D semantic segmentation network on OSDaR23, including dataset analysis.
2. A person instance pasting augmentation method to enhance pedestrian segmentation at distant ranges.
3. A track sparsification augmentation method to improve track segmentation by redistributing point density.
4. Report the first 3D semantic segmentation results on the OSDaR23 dataset.

2. Background

This background section provides a general overview of point cloud segmentation, followed by segmentation and augmentation techniques specific to the railway domain.

2.1. Point cloud semantic segmentation

Semantic segmentation assigns a class label to each element of the input. While image-based segmentation assigns labels to pixels, point cloud segmentation must handle unordered, unstructured 3D points. Deep learning has become the standard approach, surpassing traditional techniques [35]. Methods are typically categorized into view-based, voxel-based, and point-based approaches, each imposing structure onto the raw data differently.

View-based methods

View-based methods project the point cloud into one or multiple 2D images, leveraging established image-based segmentation. SnapNet [4] generates RGB-depth snapshots from various viewpoints, applies a CNN for labeling, and back-projects labels to 3D. CENet [8] uses spherical projection and channels (x, y, z, d, r) for each pixel. Larger image widths improve performance but slow inference. However, these methods lose some 3D geometric fidelity due to projection.

Voxel-based methods

Voxel-based methods discretize the point cloud into a volumetric grid and apply 3D CNNs. PVKD [14], for example, builds on Cylinder3D [41] and employs a teacher-student framework, achieving similar accuracy at lower latency. Despite structuring the data, voxelization introduces resolution limits and can demand high memory.

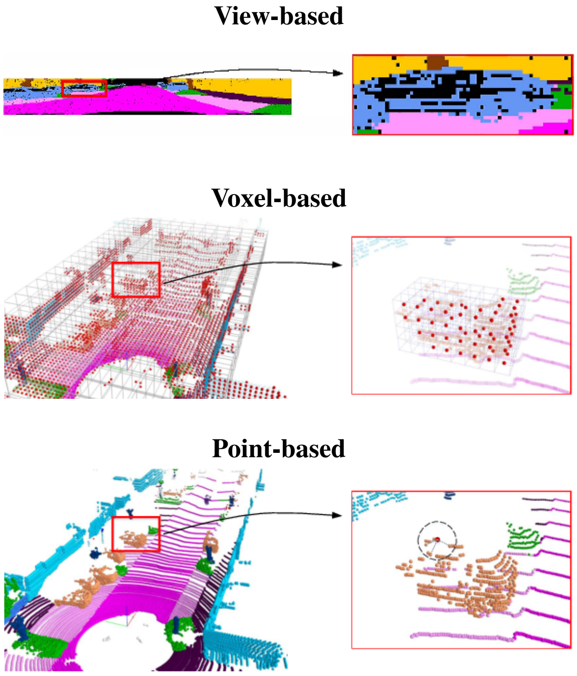


Figure 2. Schematic representation of three main deep learning-based methods for semantic segmentation of point cloud data. Adapted from [36].

Point-based methods

Point-based methods directly process points without explicit restructuring. PointNet [21] introduced MLP-based features and max-pooling for permutation invariance. Transformers, as in Point Transformer [39] and its improved PTV3 [33], leverage self-attention for robust performance. This preserves data fidelity but can be slower.

In summary, view-based and voxel-based methods effectively impose structure at the cost of fidelity, while point-based methods maintain full data integrity but may be computationally more demanding.

Fig. 2 shows an example for each of the three approaches.

2.2. Railway-domain focused segmentation

Prior work on railway point cloud segmentation focused mainly on infrastructure inspection. [28] segmented tunnel scenes into ground, lining, wiring, and rails using KPConv [31] and PointNet [21]. Similarly, [13] employed a PointNet++ [22]-based architecture to classify rails, cables, and traffic signals. These efforts used non-public datasets and older architectures, and did not target autonomous train operation.

In contrast, the automotive field has benefited from large-scale, publicly available datasets like Waymo Open Dataset [29], nuScenes [5], and SemanticKITTI [2]. Comparable resources remain scarce in the railway domain. Existing sets, such as WHU-Railway3D [12] and Rail3D [15], focus on infrastructure and rely on multi-frame reconstructions, not reflecting real-time conditions. OSDaR23 [30] addresses this gap with single-frame Li-

DAR data and classes relevant to autonomous rail operation, enabling models tailored to open railway environments.

2.3. Data augmentation methods for point clouds

Data-centric AI aims to enhance model performance by improving data quality and diversity rather than solely refining architectures. In point cloud segmentation, data augmentation (DA) introduces variations—such as rotations, translations, and sparsifications—to enrich training data and improve generalization [9, 20, 40].

Part-aware augmentation [10] applies transformations to specific object regions (e.g., sparsifying parts of cars or pedestrians), reducing reliance on dense shapes and aiding recognition at longer distances. PolarMix [34] integrates entire LiDAR scans by angular swapping or instance-level rotate-pasting, increasing variability at both scene and object levels. Both methods have demonstrated notable performance gains in 3D tasks and inspire the DA techniques explored in this work.

3. Initial Analysis

In this section, we evaluate the baseline performance of Point Transformer V3 (PTV3) on the OSDaR23 dataset. Since the dataset has seen limited use in prior research, its suitability for semantic segmentation tasks, along with potential performance bottlenecks, remains unclear. This analysis aims to establish a baseline understanding of the model's strengths and limitations, highlighting key challenges such as class imbalance and long-range prediction issues. These findings will guide subsequent efforts to enhance model performance through targeted improvements.

3.1. Baseline

For our baseline, we require a modern, high-performing semantic segmentation model suited for LiDAR point clouds. Point Transformer V3 (PTV3)[33] is the current top performer on the SemanticKITTI benchmark, demonstrating strong segmentation accuracy with reasonable inference speed. Although relatively new and less cited, it builds on the widely adopted Point Transformer[32, 39] architecture, making it a robust choice for our experiments.

3.2. Dataset and Experiment setup

We conduct our experiments on OSDaR23 [30], a single-frame, multi-sensor LiDAR dataset collected in various railway scenarios. As shown in Table 1, OSDaR23 has a higher average point density per frame than popular automotive datasets [2, 5, 29], but covers fewer total frames and primarily captures the forward view of the locomotive instead of a full 360° surround.

Although OSDaR23 provides 22 annotated classes, several contain few points, resulting in class imbalance (Fig. 3a). To address this, we merge or discard certain classes (Table 2) and remove overlapping annotations (e.g., *switch* on *track*). Figure 3b shows the resulting distribution after class mapping.

All experiments follow the official train, validation, and test splits. We adapt data augmentations to the forward-facing LiDAR viewpoint, limiting large rotations/flips and applying sensor-specific intensity normalization. We train Point Transformer V3 (PTV3) with a learning rate of 0.001, using both cross-entropy and Lovász-Softmax loss [3].

3.3. Baseline Performance

We begin by examining the baseline model's overall segmentation performance on the validation set. As shown in Table 3, the model (PTV3) achieves a mean IoU (mIoU) of 74.49%, indicating solid overall accuracy across classes. However, this summary metric masks performance issues at longer ranges.

Fig. 4 shows the recall map for the class track. For each planar grid cell of 1x1 meter, the recall is computed. The values are obtained over all frames of the validation set, providing an overview of the performances given the spatial location. In the ranges close to the sensor the recall is generally high. Beyond x=60m, however, the recall quickly degrades. This means the network has good capabilities at identifying the track points at close range but misses points further.

Similarly, person segmentation suffers at longer ranges, as reflected in the range IoU (rIoU) results (Table 4). Although performance is strong at mid-range (40–60 m), it drops significantly beyond 60 m. This decline correlates with fewer training samples at longer distances, indicating that data scarcity limits long-range accuracy.

In summary, while the baseline model performs well overall, it struggles to maintain performance at longer distances for key classes like track and person. Insufficient training data in these ranges is a likely contributor to weaker performance, motivating the need for data augmentation and other strategies to improve long-range segmentation results.

4. Methodology

This section outlines the data-centric strategies developed to address the dataset-related limitations identified in the baseline analysis. Our methodology focuses on two key augmentations: track sparsification and person instance pasting, tailored to the characteristics of the OSDaR23 dataset.

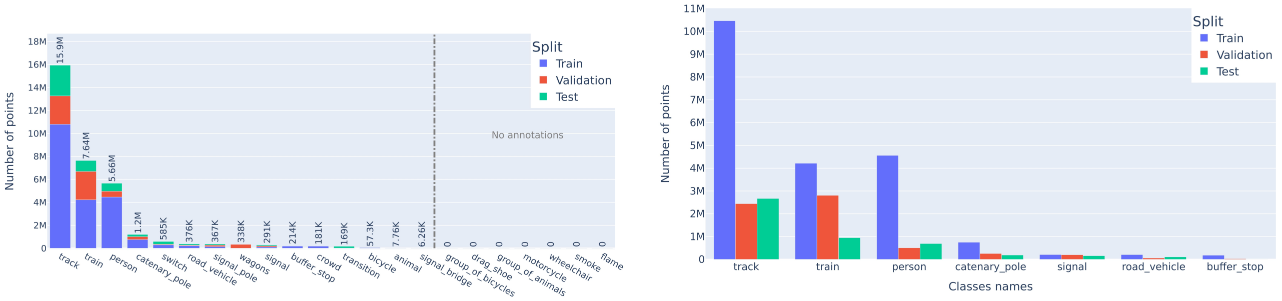
4.1. Tracks sparsification

Building on the part-aware data augmentation method [10], a new strategy was developed to improve track prediction accuracy at farther ranges.

Dense parts of track instances are sparsified by adapting the number of points per range for each track instance. The goal is to equalize point density by reducing points near the sensors to match the density farther away. This is

Table 1. Comparison of OSDaR23 to other popular autonomous driving point cloud datasets.

	SemanticKITTI [2]	NuScenes [5]	Waymo [29]	OSDaR23 [30]
Avg. Points/Frame	120K	34K	177K	204K
Ann. LiDAR frames	15K	40K	230K	1.5K
# LiDAR sources	1	1	5	6
360° field of view	✓	✓	✓	✗



(a) Points per class of the OSDaR23 dataset before mapping.

(b) Points per class of the OSDaR23 dataset after mapping (background omitted).

Figure 3. Comparison of OSDaR23 class distributions before and after mapping.

Table 2. Class mapping for OSDaR23.

Original classes	**Mapped class**
person, crowd	person
train, wagons	train
bicycle, animal, signal_bridge	background
transition, track	track
road_vehicle	road_vehicle
catenary_pole	catenary_pole
signal_pole, signal	signal
buffer_stop	buffer_stop
switch	discarded

achieved by evaluating the number of points within a window of width W at a distance d from the origin, where C_{max} represents the point count in the farthest range. Closer ranges are then randomly downsampled to match C_{max}, ensuring uniform density.

Let $P_{\text{track},i[d-W,d]}$ denote the set of points belonging to the i^{th} track instance in the planar distance range $[d-W,d]$,. The variables W (window width) and C_{max} can be adjusted based on sensor specifications and use case requirements. The pseudocode for the transformation is provided in Algorithm 1.

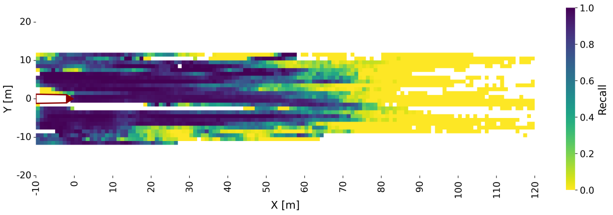


Figure 4. Recall for the class track across the validation set. High recall is observed close to the sensor, with performance decreasing beyond 60 m.

This procedure is applied to all track instances in a frame. Fig. 5 shows a point cloud before and after the transformation. In this example, the window width W is set to 10 meters, and C_{max} is set to 80 meters. The desired density is determined within the range $[C_{\text{max}} - W, C_{\text{max}}]$ (70–80 meters). Points beyond 70 meters remain unchanged, while those closer than 70 meters are significantly downsampled.

4.2. Person Instance Pasting

Inspired by PolarMix [34], we developed a methodology to paste person instances from one frame into another during training. This approach diversifies pedestrian samples by increasing their population. Unlike PolarMix, where

Algorithm 1 Track Instance Sparsification

```
Input: $P_{t,i}$ (points of track $i$), $d_{max}$ (upper range), $W$ (window width)
Output: Downsampled $P_{t,i}$
$D_i \leftarrow$ planar distances from origin for $P_{t,i}$
$d_{max} \leftarrow \min(d_{max}, \max(D_i))$
$C_{max} \leftarrow$ count points in $[d_{max} - W, d_{max})$
while $d_{max} > 0$ do
    $d_{max} \leftarrow d_{max} - W$
    $C \leftarrow$ count points in $[d_{max} - W, d_{max})$
    if $C > C_{max}$ then
        Remove $C - C_{max}$ points from $P_{t,i}$
    end if
end while
Return $P_{t,i}$
```

Table 3. Summary results for the baseline experiment on the validation dataset.

IoU (validation set)								mIoU
background	**person**	**train**	**road vehicle**	**track**	**catenary pole**	**signal**	**buffer stop**	**Overall**
96.84	69.65	86.39	70.09	82.89	47.40	48.80	93.86	74.49

Table 4. Baseline range-based IoU for the person class and approximate number of training instances.

Distance range	IoU [%] (Val)	#Instances (Train)
0–20 m	80.40	≈ 5900
20–40 m	69.73	≈ 3500
40–60 m	81.23	≈ 500
60–80 m	31.36	≈ 350
80–100 m	45.17	≈ 100

objects are rotated around the vehicle without individual transformations, our method accounts for the forward-facing point clouds in OSDaR23, which differ from the 360-degree coverage in datasets like SemanticKITTI. A simple rotation would place instances outside the field of view, necessitating significant adaptation of the original methodology.

As in PolarMix, Scan A denotes the frame undergoing transformation, and Scan B denotes the randomly selected frame from the training set containing at least one person instance.

Each instance of Scan B goes through a set of individual transformations, applied in this order:

1. Flipping along the X axis with 0.5 probability.
2. Random rotation around the instance's center along the Z axis, within the range [-180°, 180°].
3. Random shift along the Y axis, within the range [-2m, 2m].
4. Random shift towards the back of the scene, along the X axis.
5. Shifting along the Z axis so as to be at a realistic height.

An example for scan A and B and the produced result is shown in Fig. 6.

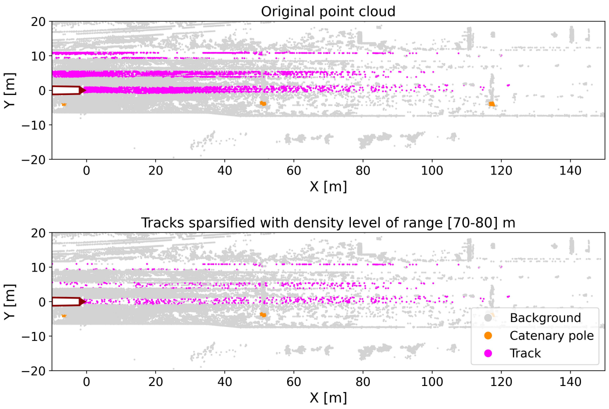


Figure 5. Effect of the tracks sparsification transformation on scene 3_fire_site_3.1, frame 58 from the OSDaR23 dataset.

For the X-axis shift, instances are translated further from the sensor to balance the distribution, with density adjustments based on the histogram of points per instance. The instance is downsampled to match the expected point count N, sampled randomly within $[N{-}0.1N, N{+}0.1N]$.

For the Z-axis shift, instances are adjusted to align with the ground. The ground height is estimated as the mean height of points in Scan A under the instance's bounding box. Special cases include estimating the ground height from railway tracks when no points overlap or ignoring unrealistic heights (e.g., above 150 cm).

After applying the transformation, the augmented dataset shows a more balanced distribution of person instances across ranges, particularly in previously sparse areas as shown in Fig 7.

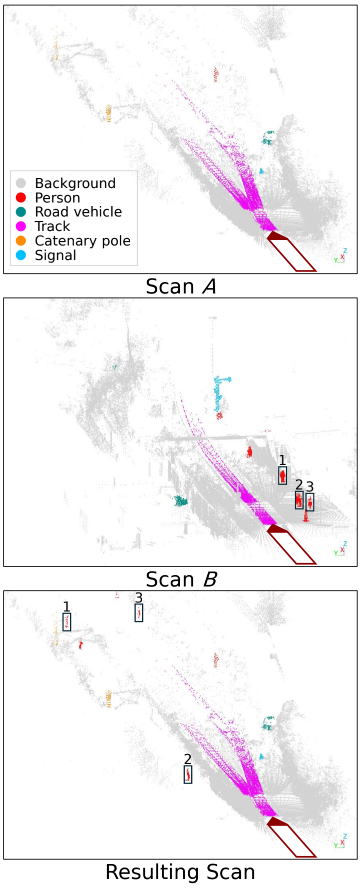


Figure 6. Visualisation of the person instances pasting transformation. Best viewed zoomed in.

5. Results

This section presents the results of applying the data augmentation (DA) methods during training, with varying proportions of affected samples. Models are first evaluated on the validation set to select the best for each task, which are then tested on the test set.

To reduce the foreground bias of IoU, we propose the mean range IoU (mean rIoU), which assigns equal importance to IoUs across all ranges. Let rIoU_i represent the range IoU for bin i. The mean rIoU is defined as:

$$\text{mean rIoU} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{rIoU}_i \tag{1}$$

where rIoU_i is computed for points in the range $[r_{min,i}, r_{max,i}[$, with $r_{min,i}$ and $r_{max,i}$ as bin boundaries, and N as the number of bins.

5.1. Track sparsification

This section evaluates the impact of the track sparsification DA method, tested with two density selection distances (DSD): 70-80m and 40-50m. The augmentation was applied with varying probabilities (p) during training, with range IoUs computed at 20m intervals from 0-100m. The baseline corresponds to $p = 0$ (no augmentation), while $p = 1$ applies the transformation to all training samples.

The ablation study identifies the best augmentation probabilities as $p = 0.6$ for DSD 70-80m and $p = 0.9$ for DSD 40-50m. Table 5 summarizes the results. The model with DSD 40-50m at $p = 0.9$ achieves the highest mean rIoU (59.49%), improving performance in ranges 40-60m and 60-80m by over 7 percentage points compared to the baseline. Both augmented models show improvements in the farthest range (80-100m), while maintaining strong performance near the origin. The baseline achieves the highest rIoU in 0-20m but with minimal difference (0.01 percentage points).

The selected model (DSD 40-50m, $p = 0.9$) also improves recall at farther distances, as shown in Fig. 8, while maintaining comparable performance closer to the origin. The results demonstrate that the track sparsification DA method effectively enhances performance at greater distances when applied with the identified optimal probabilities.

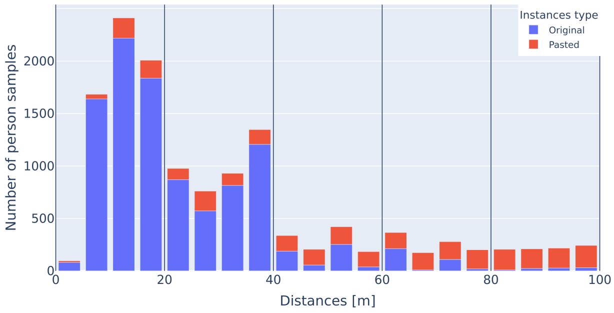


Figure 7. New distribution of samples with the person instance pasting DA applied on all frames from the train set.

5.2. Person instances pasting

This section evaluates the impact of the person instances pasting DA method using two approaches: online augmentation and offline dataset inflation. Online augmentation applies transformations to training samples in real-time, modifying data on-the-fly during training. Offline augmentation pre-processes the dataset by adding transformed samples, increasing its size before training. For person instance pasting, online augmentation randomly pastes instances during training, while offline augmentation generates augmented frames beforehand and incorporates them into the dataset.

In online augmentation, the probability (p) determines the likelihood of applying transformations to a sample during each training iteration. Higher p dynamically increases the number of augmented samples in each epoch.

In offline augmentation, the dataset size is expanded by adding transformed samples, controlled by the augmentation ratio (α). For instance, $\alpha = 1.0$ doubles the dataset by adding a transformed version of each sample, while $\alpha = 0.5$ increases the size by 50%.

Again an ablation study is conducted to determine the optimal values for p and α. The best models are selected based on mean rIoU: $p = 0.8$ for online DA and $\alpha = 0.1$ for offline DA. Table 5 compares these models with the baseline. Both approaches show significant improvements in the farthest ranges (60-100m). The online method achieves an 18.56 percentage-point increase in range 60-80m and a 12.59-percentage-point increase in range 80-100m over the baseline. Similarly, the offline method improves range 80-100m by 13.53 percentage-point. For closer ranges (0-60m), the differences are minimal, with variations below 3 percentage points. The online DA trained model ($p = 0.8$) achieves the highest mean rIoU (66.99%) and is the overall best model for this task.

5.3. Results on Test Set

The best-performing models identified during validation were evaluated on the test set to assess their generalization to new data.

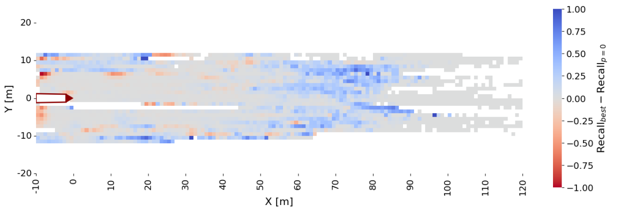


Figure 8. Recall difference between the best model and model with no augmentation on the validation set.

Table 5. Summary metrics for baseline and best models of track sparsification and person instance pasting (validation set).

	mean rIoU	r0-20	r20-40	r40-60	r60-80	r80-100
Track Sparsification (Density Selection Distances)						
baseline	56.52	**86.76**	82.05	64.98	40.98	7.82
70-80m (best)	58.01	86.64	81.61	64.74	43.15	**13.93**
40-50m (best)	**59.49**	86.75	**82.19**	**66.70**	**48.29**	13.50
Person Instances Pasting						
baseline	61.57	80.40	69.73	**81.23**	31.36	45.17
online (best)	**66.99**	78.66	**70.12**	78.49	**49.92**	57.76
offline (best)	66.77	**80.98**	**70.12**	79.46	44.59	**58.70**

Table 6. Summary of test set results. TS: track sparsification, PIP: person instance pasting (online). For each method, the best-performing model from the validation set is used.

	IoU (test set)								**mIoU**
	background	person	train	road vehicle	track	catenary pole	signal	buffer stop	
Baseline	**97.09**	**77.98**	**59.87**	72.06	81.29	71.01	**56.83**	0.53	64.58
TS (best)	97.03	77.27	57.33	73.51	80.60	75.67	53.27	0.29	64.37
PIP online (best)	97.02	77.21	57.47	**77.33**	**81.34**	**75.83**	52.25	**0.81**	**64.91**

5.3.1. Class Track

The best model for track sparsification (TS, DSD 40-50m, $p = 0.9$) improves rIoUs in ranges beyond 40m, with a 5 percentage-point increase in 80-100m compared to the baseline. However, a slight decrease in the 0-20m range is observed, attributed to the network focusing on sparsified far-range points during training, potentially neglecting the dense close-range regions. Recall maps show significant gains in 60-90m, reflecting better far-range detection, while closer ranges see some localized recall reduction on the locomotive's sides.

5.3.2. Class Person

The best model for person instance pasting (PIP online, $p = 0.8$) achieves substantial improvements in distant ranges, with increases of 11.42 and 12.59 percentage points in 60-80m and 80-100m, respectively. However, a drop of 11.58 points in the 40-60m range is linked to low diversity in the test set for this range, dominated by repetitive samples of a single stationary human instance. These repetitive samples, while well-segmented across frames, contribute to cumulative small errors, reducing the rIoU.

5.3.3. Other Classes

Table 6 summarizes the IoUs across all classes. The baseline model performs best overall for the person class, while the PIP online model achieves the highest track IoU. These results highlight that the methods are tailored to improve distant-range performance, leading to trade-offs in close-range inference. For the buffer stop class, all models show a near-complete IoU drop (from 93.86% on validation to <1% on the test set), due to overfitting to similar training-validation point clouds and poor generalization to the sparse test set.

5.3.4. Discussion of Results

The TS method enhances far-range performance while minimally impacting close-range inference, demonstrating its effectiveness in handling sparsified regions. Future work could explore variable DSDs for improved adaptability.

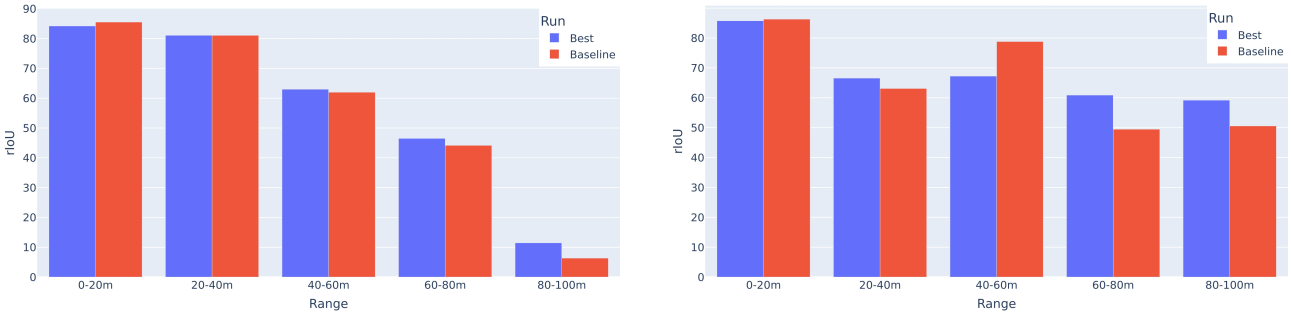
The PIP online method significantly boosts distant-range rIoUs but struggles in low-diversity regions such as 40-60m. Future improvements could include adapting the intensity field and creating a more diverse instance registry to enhance generalization.

6. Conclusion

The experiments on OSDaR23 validate the effectiveness of the proposed targeted data augmentations in improving segmentation performance at distant ranges, with minimal impact on close-range accuracy. The track sparsification and person instance pasting methods address key challenges in LiDAR-based semantic segmentation for autonomous trains.

Future work could integrate additional sensor data, such as RGB images, to leverage color information and enhance performance. Incorporating temporal data, as demonstrated in methods like MemorySeg [16], could further improve predictions by capturing motion and context. Additionally, exploring the inverse of track sparsification—densifying distant point clouds using techniques like [37]—offers another avenue for enhancing segmentation in sparse regions.

These methods provide a solid foundation for advancing multimodal, temporal, and augmentation-driven approaches in semantic segmentation for autonomous train systems.



(a) Class track: Range IoUs for baseline and TS model.

(b) Class person: Range IoUs for baseline and PIP model.

Figure 9. Comparison of range IoUs on the test set for baseline and the best-performing models: (a) Track sparsification (TS), (b) Person instance pasting (PIP online).

Acknowledgements

This work has received funding from the German Federal Ministry for Economic Affairs and Climate Action (BMWK) under grant agreement 19I21039A.

References

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(12):2481–2495, 2017. 1

[2] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9297–9307, 2019. 2, 3, 4

[3] Maxim Berman, Amal Rannen Triki, and Matthew B. Blaschko. The Lovasz-Softmax Loss: A Tractable Surrogate for the Optimization of the Intersection-Over-Union Measure in Neural Networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4413–4421, Salt Lake City, UT, 2018. IEEE. 3

[4] Alexandre Boulch, Bertrand Le Saux, and Nicolas Audebert. Unstructured Point Cloud Semantic Labeling Using Deep Segmentation Networks, 2017. 2

[5] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving, 2020. 2, 3, 4

[6] R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation . In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, Los Alamitos, CA, USA, 2017. IEEE Computer Society. 1

[7] Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. Monocular 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[8] Hui-Xian Cheng, Xian-Feng Han, and Guo-Qiang Xiao. Cenet: Toward Concise and Efficient Lidar Semantic Segmentation for Autonomous Driving. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, pages 01–06, 2022. 2

[9] Shuyang Cheng, Zhaoqi Leng, Ekin Dogus Cubuk, Barret Zoph, Chunyan Bai, Jiquan Ngiam, Yang Song, Benjamin Caine, Vijay Vasudevan, Congcong Li, Quoc V. Le, Jonathon Shlens, and Dragomir Anguelov. Improving 3D Object Detection through Progressive Population Based Augmentation, 2020. 3

[10] Jaeseok Choi, Yeji Song, and Nojun Kwak. Part-Aware Data Augmentation for 3D Object Detection in Point Cloud, 2021. 3

[11] Deutsche Bahn. Metropolitan Network: A strong European railway for an ever closer union, 2023. 1

[12] Zhen Dong, Fuxun Liang, Bisheng Yang, Yusheng Xu, Yufu Zang, Jianping Li, Yuan Wang, Wenxia Dai, Hongchao Fan, Juha Hyyppä, and Uwe Stilla. Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark. *ISPRS Journal of Photogrammetry and Remote Sensing*, 163:327–342, 2020. 2

[13] Javier Grandio, Belén Riveiro, Mario Soilán, and Pedro Arias. Point cloud semantic segmentation of complex railway environments using deep learning. *Automation in Construction*, 141:104425, 2022. 2

[14] Yuenan Hou, Xinge Zhu, Yuexin Ma, Chen Change Loy, and Yikang Li. Point-to-Voxel Knowledge Distillation for LiDAR Semantic Segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8469–8478, New Orleans, LA, USA, 2022. IEEE. 2

[15] Abderrazzaq Kharroubi, Zouhair Ballouch, Rafika Hajji, Anass Yarroudh, and Roland Billen. Multi-Context Point Cloud Dataset and Machine Learning for Railway Semantic Segmentation. *Infrastructures*, 9(4):71, 2024. 2

[16] Enxu Li, Sergio Casas, and Raquel Urtasun. Memory-Seg: Online LiDAR Semantic Segmentation with a Latent Memory. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 745–754, Paris, France, 2023. IEEE. 7

[17] Yanwei Li, Xinze Chen, Zheng Zhu, Lingxi Xie, Guan Huang, Dalong Du, and Xingang Wang. Attention-Guided Unified Network for Panoptic Segmentation . In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7019–7028, Los Alamitos, CA, USA, 2019. IEEE Computer Society. 1

[18] Zhijian Liu, Haotian Tang, Alexander Amini, Xingyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfu-

sion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023. 1

[19] Rohit Mohan and Abhinav Valada. Efficientps: Efficient panoptic segmentation. *International Journal of Computer Vision*, 129:1551 – 1579, 2020. 1

[20] Jiquan Ngiam, Benjamin Caine, Wei Han, Brandon Yang, Yuning Chai, Pei Sun, Yin Zhou, Xi Yi, Ouais Alsharif, Patrick Nguyen, Zhifeng Chen, Jonathon Shlens, and Vijay Vasudevan. StarNet: Targeted Computation for Object Detection in Point Clouds, 2019. 3

[21] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation, 2017. 1, 2

[22] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 2

[23] Charles R. Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J. Guibas. Frustum PointNets for 3D Object Detection from RGB-D Data . In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 918–927, Los Alamitos, CA, USA, 2018. IEEE Computer Society. 1

[24] Giuseppe Rizzi. Automated metros. *UITP*, Accessed: 5 Sept. 2024. 1

[25] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1

[26] Andrea Simonelli, Samuel Rota Bulò, Lorenzo Porzi, Peter Kontschieder, and Elisa Ricci. Are we missing confidence in pseudo-lidar methods for monocular 3d object detection? In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3205–3213, 2021. 1

[27] Kshitij Sirohi, Rohit Mohan, Daniel Büscher, Wolfram Burgard, and Abhinav Valada. Efficientlps: Efficient lidar panoptic segmentation. *IEEE Transactions on Robotics*, 38 (3):1894–1914, 2022. 1

[28] M. Soilán, A. Nóvoa, A. Sánchez-Rodríguez, B. Riveiro, and P. Arias. Semantic segmentation of point clouds with pointnet and kpconv architectures applied to railway tunnels. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, V-2-2020:281–288, 2020. 2

[29] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Sheng Zhao, Shuyang Cheng, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in Perception for Autonomous Driving: Waymo Open Dataset, 2020. 2, 3, 4

[30] Rustam Tagiew, Martin Köppel, Karsten Schwalbe, Patrick Denzler, Philipp Neumaier, Tobias Klockau, Martin Boekhoff, Pavel Klasek, and Roman Tilly. OSDaR23: Open Sensor Data for Rail 2023. In *2023 8th International Conference on Robotics and Automation Engineering (ICRAE)*, pages 270–276, 2023. 1, 2, 3, 4

[31] Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, Francois Goulette, and Leonidas J. Guibas. KPConv: Flexible and Deformable Convolution for Point Clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6411–6420, 2019. 2

[32] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point Transformer V2: Grouped Vector Attention and Partition-based Pooling, 2022. 3

[33] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point Transformer V3: Simpler, Faster, Stronger, 2023. 2, 3

[34] Aoran Xiao, Jiaxing Huang, Dayan Guan, Kaiwen Cui, Shijian Lu, and Ling Shao. PolarMix: A General Data Augmentation Technique for LiDAR Point Clouds. 2022. 3, 4

[35] Yuxing Xie, Jiaojiao Tian, and Xiao Xiang Zhu. Linking Points With Labels in 3D: A Review of Point Cloud Semantic Segmentation. *IEEE Geoscience and Remote Sensing Magazine*, 8(4):38–59, 2020. 2

[36] Jianyun Xu, Ruixiang Zhang, Jian Dou, Yushi Zhu, Jie Sun, and Shiliang Pu. RPVNet: A Deep and Efficient Range-Point-Voxel Fusion Network for LiDAR Point Cloud Segmentation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16004–16013, Montreal, QC, Canada, 2021. IEEE. 2

[37] Jihwan You and Young-Keun Kim. Up-Sampling Method for Low-Resolution LiDAR Point Cloud to Enhance 3D Object Detection in an Autonomous Driving Environment. *Sensors*, 23(1):322, 2023. 7

[38] Jiaying Zhang, Xiaoli Zhao, Zheng Chen, and Zhejun Lu. A Review of Deep Learning-Based Semantic Segmentation for Point Cloud. *IEEE Access*, 7:179118–179133, 2019. 1

[39] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip H. S. Torr, and Vladlen Koltun. Point Transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16259–16268, 2021. 2, 3

[40] Qinfeng Zhu, Lei Fan, and Ningxin Weng. Advancements in Point Cloud Data Augmentation for Deep Learning: A Survey. *Pattern Recognition*, 153:110532, 2024. 3

[41] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and Asymmetrical 3D Convolution Networks for LiDAR Segmentation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9934–9943, Nashville, TN, USA, 2021. IEEE. 2