Proceedings of the Seventh International Conference on
Railway Technology: Research, Development and Maintenance
Edited by: J. Pombo
Civil-Comp Conferences, Volume 15, Paper 14.4
Civil-Comp Press, Edinburgh, United Kingdom, 2026
ISSN: 2753-3239, doi: 10.4203/ccc.15.14.4

LiDAR-Based Coupling Detection for Automatic Shunting: A Simulation-to-Real-World Transfer Study

A. Schotten, A. Von Stillfried and R. Pfaff

Institute for Rail Vehicles (IFS), RWTH Aachen University, Germany

Abstract

This study presents a comprehensive investigation into the transferability of a laser scanning-based coupling detection algorithm from simulation environments to real-world applications in automatic shunting. With the growing implementation of automation in rail transport systems, ensuring efficient and safe coupling mechanisms has become an important consideration. A data processing pipeline was developed within a Gazebo simulation framework to develop the detection system, followed by extensive testing and adaptation on data collected from a real-world shunting vehicle. The results demonstrate that while accurate recognition of coupler positions can be achieved, challenges persist in adapting algorithms due to discrepancies between simulated and actual conditions. The findings in this paper highlight the importance of model fidelity and parameter tuning in ensuring reliable performance under varying environmental factors. This research advances the understanding of automated coupling systems and contributes to the knowledge needed to bridge the gap between simulation and reality in the context of railway automation.

Keywords: railway automation, automated shunting, simulation-to-reality transfer, laser scanning, coupling detection, point cloud processing

1 Introduction

The continuous modernisation of rail transport systems has led to a growing demand for automation in operational procedures, particularly in the context of freight shunting. One critical, yet traditionally manual aspect of this process is the coupling and uncoupling of wagons. Manual coupling remains labour-intensive, time-consuming, and potentially hazardous [1]. Despite the widespread use of mechanical coupling systems such as the Union Internationale des Chemins de fer (UIC) hook across Europe, the integration of automation into this process presents a valuable opportunity for improving both workplace safety and operational efficiency [2].

In recent years, automation in railway shunting has been gaining attention due to the increasing need to improve workplace safety and reduce operational costs [1, 2]. Development is ongoing for remote-controlled and semi-autonomous coupling systems. Also, the Digital Automatic Coupler (DAC) has been proposed as an effective way to sustainably increase the efficiency of rail freight transport in Europe. DAC developments are part of broader European initiatives aimed at modernizing freight operations by enabling automatic mechanical coupling and introducing advanced features such as electro-pneumatic (EP) braking. The inclusion of DAC combined with EP braking has shown potential to improve longitudinal train dynamics and reduce in-train forces, contributing to enhanced operational performance in freight rail operations. [12] Despite these promising developments, fully automated coupling systems are still under development and the coupling of freight wagons remains largely manual in practice. Standard systems such as the UIC hook are still widely used across Europe [3], requiring precise and reliable recognition for any automation effort without replacing the coupler.

For perception systems which are required for automated coupling processes, prototyping and testing is done in simulation environments like Gazebo. These platforms offer a controlled and reproducible setting, making them well suited for prototyping detection systems before deployment. Although these kinds of environments are good for prototyping, the transfer of algorithms designed in these environments to the real world is often challenging due to real world variables like the environment, sensor behaviour, and simulation model accuracy. [8, 10] Simulation environments like Gazebo offer the ability to develop and refine robotic algorithms in a controlled setting. When combined with the Robot Operating System (ROS2), they support modular system design and sensor simulation [6, 7]. However, transferring perception algorithms from simulation to real-world environments remains a known challenge in robotics, often referred to as the simulation-to-reality (sim2real) gap [11]. While the transfer from simulation to real world settings is a very popular topic in mobile robotics, where the simulation-to-reality gap was first addressed, very little attention has been paid to it in the context of railway automation.

The core objective of this study is to systematically analyze this sim2real gap. The challenge in this transition lies in the discrepancies between idealized virtual models and the stochastic nature of real-world railyards. Simulated environments, such as Gazebo, inherently provide an idealised data structure characterized by uniform point density, flawless sensor behavior, and a complete absence of background clutter. Conversely, real-world railway environments introduce severe variables, including sensor noise, mechanical actuation distortion, and complex background geometry. By evaluating how a classical pipeline—utilizing Principal Component Analysis (PCA) and Iterative Closest Point (ICP) algorithms—must adapt its clustering radii and correspondence thresholds, this research quantifies the structural modifications required to bridge the divide between low-fidelity virtual prototyping and real-world deployment on edge hardware.

This work investigates the transferability of a LiDAR-based coupler detection algorithm developed with simulation data to a real-world shunting vehicle. A complete data processing pipeline was initially developed for a model in Gazebo and was later used on the data recorded by a real-world shunting vehicle. The results are evaluated based on classification accuracy and the adaptations required for real-world deployment. LiDAR sensors are commonly used in mobile robotics and autonomous systems for 3D environment perception. Enabling point cloud-based object recognition and pose estimation by employing methods such as PCA and ICP [4, 5, 9]. While these approaches are robust in structured simulated environments, they often require tuning to handle real-world noise and incomplete data.

2 Methodology

The methodology for data acquisition (2.1,2.2) and processing (2.3) is as consistent as can be achieved across both simulation and real-world environments. This approach helps to identify the necessary adaptations when transferring the algorithm for processing data from simulated to real-world applications. (3.2)

2.1 Simulation

For the simulation data a Gazebo Harmonic world was created with the Rotrac and a flat wagon inside. A LiDAR sensor was attached at the end of the Rotrac which was facing the wagon. This sensor was then instructed to record a 3D point cloud from the wagon. It only recorded the wagon and no other parts which were placed in the world like the tracks.

2.2 Real World

In order to obtain real-world data a small road-rail shunting robot, the Rotrac E2, was used in combination with a LiDAR setup. As LiDAR sensor a 2D-sensor paired with a stepper motor was used. The core sensing device is a SICK TiM571-2050101 scanner. The mechanical sweeping motion enabled by the stepper motor introduces a vertical scanning dimension, covering a total angle of 73°. Thus, resulting in a pseudo-3D scan. Data was recorded at a frequency of 20 Hz with a buffer size of 250 points, optimized for capturing the detailed geometry of the coupler.

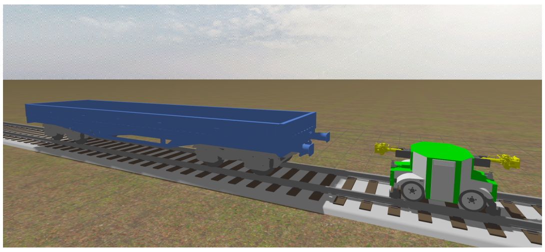

Figure 1: Simulation of shunting robot and flatbed wagon.

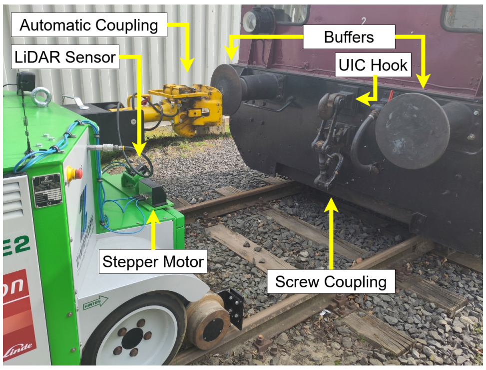

Figure 2: Measurement setup on Railway Tracks.

2.3 Software Pipeline

To process the point cloud and recognise the coupling, a sequential data processing pipeline was implemented. As first steps, noise reduction and downsampling are applied in sequence. Downsampling is performed first, significantly reducing the number of points and thereby decreasing the computational load of subsequent filtering steps. (Fig. 3) To downsample, a voxel-based approach with a grid filter of $1\,\mathrm{cm} \times 1\,\mathrm{cm} \times 1\,\mathrm{cm}$ is used to preserve geometric detail, while streamlining the dataset. To remove isolated points caused by sensor noise, a radius-based outlier filter is employed. (Fig. 3) Specifically, only points with at least six neighbours within a 10 cm radius are retained. This method was selected over statistical filtering due to its robustness in datasets with nonuniform point density and its ability to preserve fine structural details, such as the UIC hook, while discarding sparse anomalies. The implementation is based on the Point Cloud Library (PCL) [13].

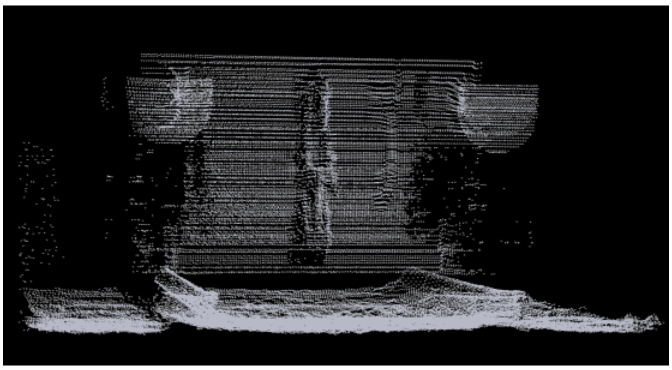

Figure 3: Real-world point cloud after downsampling and outlier removal.

Following outlier removal, the point cloud is segmented into distinct clusters to isolate the coupler components from non-relevant structures, such as the buffers. (Fig. 4) This step is necessary to reduce visual and computational complexity and to focus subsequent analysis on the relevant geometry. Clustering is performed using a Euclidean distance-based approach, where points are grouped based on their spatial proximity.

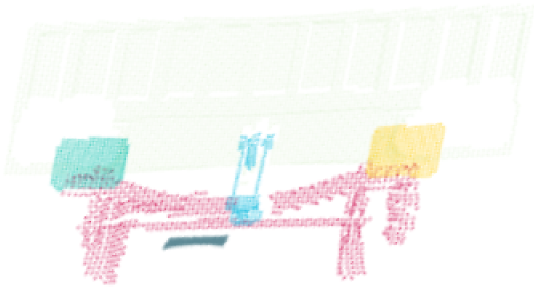

Figure 4: Simulated clustered point cloud visualisation.

To filter out noise and irrelevant groupings, clusters were additionally constrained by size. Only those containing between 100 and 1,000 points were retained. (Fig. 5) This range was found to be effective for capturing coupler elements while excluding small noise clusters and large, unrelated structures. The remaining clusters were merged into a single point cloud and visualised using distinct colours to support manual validation of the segmentation quality. The colour information was not used in subsequent processing steps.

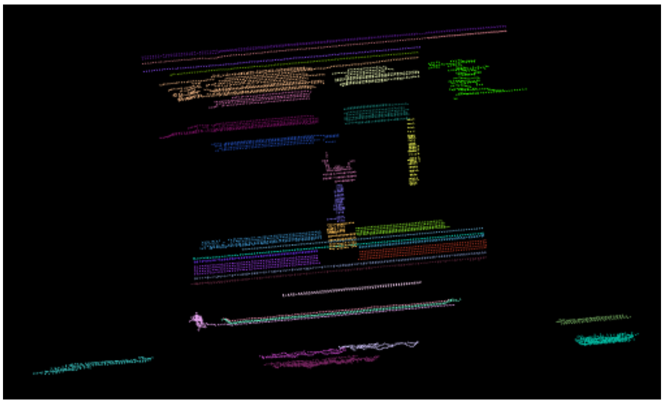

Figure 5: Real-world clustered point cloud visualisation after applying Euclidean distance-based segmentation and removing very small and big clusters.

For the initial setup of the data analysis, the correspondence grouping, the processed point cloud was compared with a set of predefined coupler reference models. These models were created using three different approaches: point clouds generated from simulation scans, point clouds acquired from real-world scans (Fig. 6) of freight couplers, and point clouds directly exported from a CAD model. For each creation method, three characteristic and mechanically stable coupler configurations were considered: an ideal coupling position, a downward-hanging coupler position, and an incorrectly self-coupled configuration. This resulted in a total of nine reference models used for correspondence grouping and evaluation.

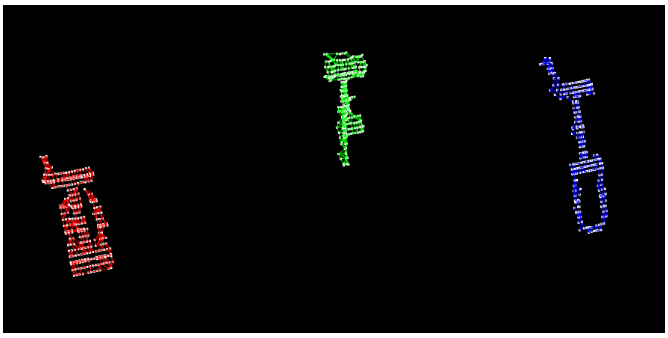

Figure 6: Point cloud models from real-world scans for the three coupler positions with overlaid mesh net for better visualisation (left: ideal coupling position, middle: downward-hanging coupler position, right: incorrectly self-coupled position).

The alignment process begins with PCA, which is applied to the model point clouds to calculate their principal axes. This step normalises the model orientation by aligning it to a common reference frame, reducing the influence of rotation and positioning differences between the models and the recorded scene. After this rough alignment, the refined matching is carried out using the ICP algorithm, which iteratively minimises the Euclidean distance between nearest neighbour points in the two clouds by computing an optimal transformation matrix. The ICP algorithm was strictly configured with a maximum of 50 iterations, a transformation convergence threshold of 1×10^{-8}, and a Euclidean fitness epsilon of 0.01. A model is considered a valid match if the ICP alignment results in a fitness score below 0.05 and the overlap score, defined as the proportion of model points located within 0.01 m of any point in the recorded scene, exceeds a predefined threshold. For visual inspection, the matched model is rendered in red and superimposed onto the recorded scene, which is displayed in white (Fig. 7). This visualisation allows for clear interpretation of the alignment outcome and facilitates validation of the correspondence during the development process of the software. The simulated point cloud could now be matched to the models.

3 Experiments

Following the establishment of the methodology, experiments were conducted to collect data and evaluate the performance of the algorithm. The experiments conducted aim to validate the effectiveness of the developed algorithm in both simulated and real-

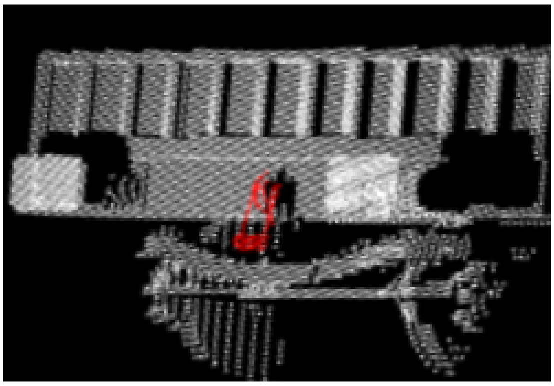

Figure 7: Correctly matched model with simulated clustered point cloud (all clusters visible for easier visualisation).

world settings. This comparison enables an evaluation of the software's effectiveness in its transfer from simulation to real-world operational conditions (3.2).

3.1 Simulation

A 3D static point cloud was recorded of the coupler using the previously described world setup. As the simulation environment does not change, all that was changed was the position of the coupler and a slight difference in distance from the sensor to the coupler. This led to a very small data set. The recorded data was processed using the developed software pipeline designed to identify the coupler and determine its position, thereby providing critical information regarding the safety of coupling to the vehicle.

3.2 Transfer to real world

The real-world data was then recorded. The Rotrac was positioned in front of a vehicle with a screw coupling with distances from the sensor to the coupler ranging from $1\,\mathrm{m}$ to $5\,\mathrm{m}$. Two different vehicles were used (shunting locomotive Koef and regional train LVT) as the recorded coupling to ensure a broader data pool. The coupler was scanned in the same different stable positions as the models used for the coupler recognition. Hereby focusing on the regular ideal coupling position in which the coupling should be before coupling.

During data collection, a critical sim2real discrepancy emerged concerning the mechanical actuation of the sensor. In the Gazebo simulation, a 3D point cloud is generated instantaneously as a static snapshot. However, in the real-world setup, the 2D LiDAR is mechanically pitched up and down by the stepper motor in a continuous looped motion. This temporal-spatial data acquisition introduced slight misalignments between consecutive LiDAR sweeps due to minor shifts in the sensor's physical position and signal variations. When these multiple sweeps were aggregated into a single static point cloud, overlapping regions appeared, causing the structural features of the coupler to become duplicated or 'ghosted'. This overlap severely distorted the point cloud and complicated the initial correspondence matching process, as the same geometric features were evaluated multiple times with slight offsets. This phenomenon underscores a significant limitation in transferring idealized simulated sensors to low-cost, mechanically actuated real-world hardware.

Data of the rear sensor was recorded at a frequency of $20\,\mathrm{Hz}$ with a buffer size of 250 points. While real-time visualisation was available through RViz during the scans, data was simultaneously stored as a rosbag2 file for post-processing. This file had to be converted into a static point cloud before using it in the processing algorithm. Additionally, only points $1.5\,\mathrm{m}$ to each side of the LiDAR sensor were used in order to not get as many unnecessary points from the surrounding. (Fig. 8)

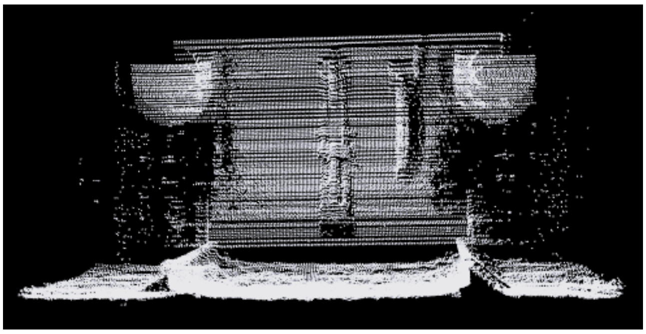

Figure 8: Cropped, aggregated static point cloud

Trying to input the point clouds from the real-world data through the algorithm developed for and successful with the simulation data proved unsuccessful. The points responding to the coupler were already lost during the cluster extraction process. Therefore, this step had to be adapted first. While a maximum point-to-point distance of $0.7\,\mathrm{m}$ had proven effective in the simulation environment, this threshold could not be reused for real-world data, as it frequently grouped nearly the entire point cloud into a single cluster. Instead, a reduced threshold of $0.2\,\mathrm{m}$ was determined through iterative refinement to provide reliable separation of the coupler structure at measurement distances of $1.5\,\mathrm{m}$. Higher thresholds resulted in over-clustering, while lower values risked fragmenting relevant structures. To account for a broader flexibility with the distance to the coupler and accommodate the significantly greater diversity exhibited by the real-world data the parameter could not be fixed at $0.2\,\mathrm{m}$.

To accurately compare the generated models with the recorded data, the coupler clusters must be visible after the cluster extraction, ideally accompanied by a minimal number of extraneous clusters. As this could not be achieved using a single value for the maximum permissible distance between points while still considering them part of the same cluster, a second, and subsequently a third, cluster extraction node was introduced. This resulted in three distinct cluster extraction point cloud datasets, each generated using different parameter settings. To optimise the extraction of the coupler clusters at sensor-coupler distances between $1.5\,\mathrm{m}$ to $2.5\,\mathrm{m}$, the parameters were set to $0.2\,\mathrm{m}$, $0.3\,\mathrm{m}$, and $0.4\,\mathrm{m}$, respectively. The minimum and maximum number of points per cluster were retained at 100 and 1,000, as in the previous version.

The correspondence node was slightly modified to enable comparison of the three models against each of the three processed point cloud files derived from the recorded data. The updated node identifies the best match between the models and the processed datasets. (Fig. 9)

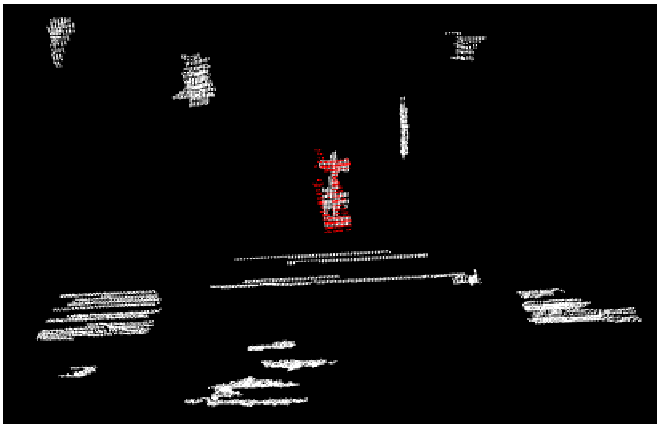

Figure 9: Correctly matched model with clustered point cloud recorded from $2\,\mathrm{m}$ distance.

After running the point clouds through the renewed algorithm, the matches were still not found with the models derived from the simulated coupler scans but also not with the ones directly taken from the CAD models. Only the models created by scanning a real-world coupler would lead to a higher matching success. Additionally, the process was found to be promising to skip the clustering step altogether and using the correspondence grouping algorithm directly after downsampling and noise reduction. (Fig. 10) This way the models only need to be matched against a single point cloud rather than three separate clustered variants. This not only eliminates the clustering step, thereby reducing computational time, but also decreases the overall processing complexity and resource consumption, such as memory usage and intermediate data handling, required for the matching process. Therefore, this method was also used with future real-time decision making in mind.
An attempt was made to mitigate false matches in high-density regions by reducing the voxel grid size used in the downsampling procedure. However, this adjustment resulted in the cluster structures no longer being reliably identified, and it failed to improve model alignment in the non-clustered point clouds.

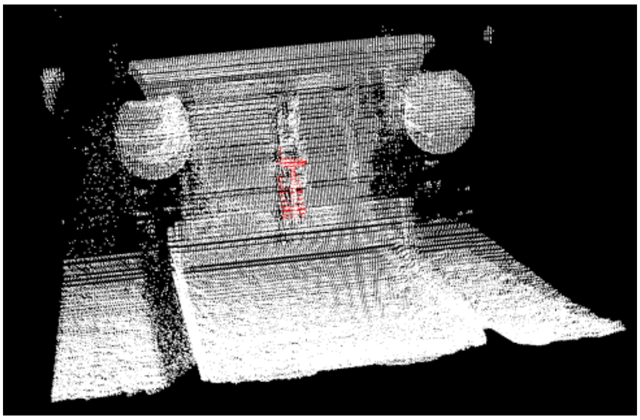

Figure 10: Correctly matched parked model with clustered point cloud recorded from $2\,\mathrm{m}$ distance.

4 Results and Discussion

Developed simulation process from simulated data was used as a general guideline for the real-world data processing. While the coupler in the simulated data was always recognised by the previously defined algorithm (2.3) this was not the case for the coupler from the real-world data. After the above-mentioned (3.2) adaptations were made the results can be observed.

One of the most significant challenges encountered during the real-world evaluation was the handling of environmental clutter, which directly dictated the limits of Euclidean clustering. In the Gazebo simulation, the environment was highly controlled, containing only the target wagon. Consequently, a generous maximum point-to-point clustering threshold of 0.7 m successfully and cleanly segmented the coupler. In stark contrast, the real-world tracks have adjacent geometric noise. Applying the simulated $0.7\,\mathrm{m}$ threshold resulted in massive over-clustering, grouping the coupler, the wagon body, the track bed, and the buffers into a single indistinguishable mass. Even after iteratively reducing the threshold to $0.2\,\mathrm{m}$, the clustering approach struggled to cleanly separate the coupler from highly proximal structures, most notably the brake pipes. This caused the PCA and ICP alignment steps to fail or match to the wrong geometry.

This limitation highlights an architectural oversight in the initial pipeline design: the failure to leverage known mechanical constraints. Freight wagon couplers are consistently centered on the vehicle's longitudinal axis. Future iterations of this sim2real pipeline should incorporate a geometric Region of Interest (ROI) filter prior to clustering. Applying a bounding box based on the known kinematic envelope of the coupler would dramatically reduce the search space, eliminate proximal background noise such as the interfering brake pipes, and significantly optimize the ICP registration process without requiring fragile parameter tuning.

Furthermore, it must be noted that while data was collected at distances up to $5\,\mathrm{m}$, reliable detection peaked strictly within the $1.5\,\mathrm{m}$ to $2\,\mathrm{m}$ range. Beyond $5\,\mathrm{m}$, the point density returned by the LiDAR degraded significantly, causing the coupler geometry to become too sparse for accurate ICP alignment. Below $1.5\,\mathrm{m}$, the sensor's field of view and the mechanical sweep angle were insufficient to capture the complete structural context of the coupler, leading to fragmented point clouds. From an operational perspective in automated shunting, identifying the coupler status at a $1.5\,\mathrm{m}$ distance is sufficient to confirm coupling readiness before the vehicle initiates the final, low-speed mechanical connection. An accurate recognition of the coupling could be seen in distances between $1.5\,\mathrm{m}$ to $2\,\mathrm{m}$. Looking at the algorithm using the three clustering steps within this distance range the coupling itself was recognised in 20 of 21 scans (95.2%) (Fig. 9). The correct position of the coupling was recognised in only 16 of 21 scans (76.2%). However, notably none of the not ideal coupling positions being falsely classified as ideal coupling position, indicating a promising reliability in avoiding false positives. While this sample size is preliminary, this binary distinction is a critical foundation for ensuring safe automated coupling processes in future, large-scale deployments. Now using the algorithm which drops the clustering altogether (Fig. 10) 19 of 21 scans (90.5%) in the distance of $1.5\,\mathrm{m}$ to $2\,\mathrm{m}$ were recognized correctly making it only slightly less accurate. Notably, all recognised couplers were also recognized in the correct positions. The comparison with the non-clustered point clouds also appears to offer greater efficiency, particularly in light of the observation that some of the clusters associated with mismatches are not easily removable if the coupler clusters are to be preserved. For instance, the cluster corresponding to the brake pipes adjacent to the coupler frequently only disappears after the coupler cluster itself has already been eliminated. As the models are, in some instances, mismatched to these pipes (Fig. 11), this suggests that the clustering process does not consistently permit a clear distinction between relevant and irrelevant structures.

Dropping the clustering step with the simulated data did not improve recognition, as the simulated point cloud only contained points from the wagon that were evenly spread and easily categorized. However, in the real-world context, bypassing the clustering step entirely proved to be a highly effective adaptation. By removing the brittle dependency on Euclidean separation, the pipeline avoided the accidental deletion or distortion of coupler points. Furthermore, eliminating the clustering node fundamentally reduces the computational complexity of the pipeline. While specific latency profiling was not the primary focus, dropping the computationally expensive nearest-neighbor searches required for clustering substantially decreases memory consumption and processing time. This reduction in overhead is a critical requirement for deploying the algorithm on the edge-computing hardware of the Rotrac E2 for real-time, low-latency automated shunting.

Using simulations to generate first datasets helped to accelerate the development and design of the processing algorithms as immediate tests were possible without having to record real-world data which can be a major undertaking. The model fidelity should be viewed as an important factor, as the better the simulation mimics real-world situations the closer the data will be to the real-world data. The algorithms parameters are not directly transferable as the real-world does not behave like the perfected isolated simulation environment.

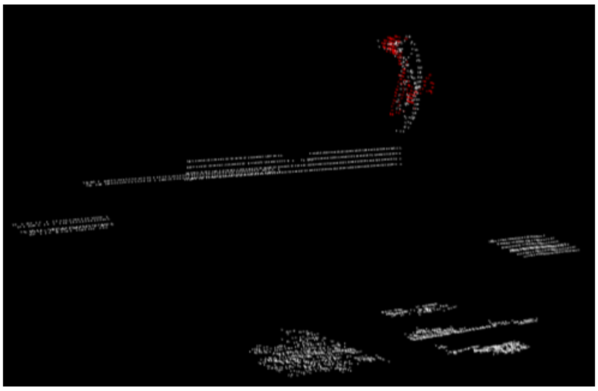

Figure 11: Mismatch cluster model with clustered data - Match found in brake pipes.

5 Conclusion and Outlook

The increasing demand for automation in freight rail operations underscores the necessity for reliable technologies that enhance both efficiency and safety in shunting environments. This study has successfully developed a LiDAR-based coupling detection system, demonstrating its feasibility through simulation and subsequent real-world application. The initial algorithm designed in Gazebo provided a foundational framework that guided the adaptation process for real-world scenarios. While the direct use of the simulated data-based algorithm was not possible it could be used as a general guideline for real world cases. Throughout this research, it was established that accurate identification of coupler positions is essential for ensuring safe and efficient automated operations. The results indicate that with appropriate adaptations, such as tuning clustering parameters and refining the correspondence matching process, reliable coupling detection can be achieved within an operational range of approximately $1.5\,\mathrm{m}$ to $2\,\mathrm{m}$. However, this study also revealed significant challenges associated with transferring algorithms from simulation to real-world applications. The discrepancies between simulated environments and real-world conditions, specifically background clutter, sensor sweep misalignments, and varying point densities, necessitated critical structural adjustments to the pipeline, such as omitting the clustering phase entirely. These findings emphasize the importance of model fidelity; higher fidelity simulations yield better transferability to actual operational contexts.

Looking forward, transferability should be tested on different algorithms and integrating machine learning techniques could improve adaptability to different geometries and dynamic environments. For the developed coupler recognition future work should focus on broadening validation across diverse coupling types and environmental conditions to enhance robustness in various scenarios. Implementing geometric Region of Interest (ROI) filters should also be prioritized to mitigate background noise without relying on fragile parameter tuning. Moreover, refining model generation methods through hybrid strategies combining simulated data with real-world observations will be essential for improving correspondence accuracy under real-world conditions. Simplifying operational modes by utilizing only key models could streamline implementation while ensuring safety remains paramount.

In summary, this study contributes significantly towards advancing autonomous coupling detection systems and understanding how data processing algorithms can be transferred from simulated to real-world data sets. By addressing current challenges and exploring future enhancements, we can pave the way for safer and more efficient automated railway operations.

References

[1] M. Zasiadko, “Digital coupling reduces human exposure to risks”, RailTech, July 24, 2020. Available at: https://www.railtech.com/digitalisation/2020/07/24/digital-coupling-reduces-human-exposure-to-risks/, Accessed on: January 17, 2025.

[2] Encyclopaedia Britannica, Inc., “Automation”, Encyclopaedia Britannica, December 04, 2024. Available at: https://www.britannica.com/technology/automation, Accessed on: January 17, 2025.

[3] S. Kobert, D. Busch, “Zug- und Stoßeinrichtungen”, in “Handbuch Schienenfahrzeuge”, C. Schindler, (Editor), DVV Media Group, Hamburg, Germany, 436–444, 2014.

[4] J. Zhou, “A review of LiDAR sensor technologies for perception in automated driving”, Academic Journal of Science and Technology, Vol. 3, Nr. 3, 2022.

[5] H. Pan, Z. Chen, L. Chen, H. Hong, “Major technical barriers and a high-precision distance measurement method for frequency modulated continuous wave LiDAR”, Optics and Lasers in Engineering, Vol. 183, 2024.

[6] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, W. Woodall, “Robot Operating System 2: Design, architecture, and uses in the wild”, Science Robotics, Vol. 7, Nr. 66, 2017.

[7] S. Macenski, T. Moore, D. V. Lu, A. Merzlyakov, M. Ferguson, “From the desks of ROS maintainers: A survey of modern & capable mobile robotics algorithms in the robot operating system 2”, Robotics and Autonomous Systems, Vol. 168, 2023.

[8] Gazebo Simulation, “ROS Installation for Gazebo”, Available at: https://gazebosim.org/docs/latest/ros_installation/, Accessed on: February 6, 2025.

[9] Z. Nan, W. Liu, G. Zhu, H. Zhao, W. Xia, X. Lin, Y. Yang, “LiDAR-Camera joint obstacle detection algorithm for railway track area”, Expert Systems with Applications, Vol. 275, 2025.

[10] N. Koenig, A. Howard, “Design and use paradigms for Gazebo, an open-source multi-robot simulator”, in “Proceedings of the 2004 IEEE/RSJ International

Conference on Intelligent Robots and Systems", Sendai, Japan, September 28–October 2, 2004.

[11] E. Aljalbout, J. Xing, A. Romero, I. Akinola, C. R. Garrett, E. Heiden, A. Gupta, T. Hermans, Y. Narang, D. Fox, D. Scaramuzza, F. Ramos, "The reality gap in robotics: Challenges, solutions, and best practices", October 2025.

[12] L. Cantone, T. Durand, A. Ottati, G. Russo, R. Tione, "The digital automatic coupler (DAC): An effective way to sustainably increase the efficiency of freight transport in Europe", Sustainability, Vol. 14, 15671, 2022.

[13] R. B. Rusu, S. Cousins, "3D is here: Point Cloud Library (PCL)", in "Proceedings of the IEEE International Conference on Robotics and Automation", May 2011.