# Rail to Digital automated up to autonomous train operation

## D7.5 – Development of Data-Factory toolchain and perform ML/AI model training

Due date of deliverable: 30/09/2025

Actual submission date: 30/09/2025

Leader/Responsible of this Deliverable: Dr. Philipp Neumaier | DB InfraGO AG

Reviewed: Y

| **Document status** | | |
|---|---|---|
| Revision | Date | Description |
| 01 | 01/09/2025 | First preparation of document |
| 02 | 24/09/2025 | Revision shared with Partners from WP7 |
| 03 | 30/09/2025 | Revision for TMT Process |
| 04 | 31/10/2025 | Exec. Summary: clarified curated vs raw data; Fig. 1: unified y-axis range; Görlitz Dataset: added distribution figure; Bicycle class: explanation added; Baseline Model: clarified COCO pretraining; Typo fixed (“his”→“this”); Confidence threshold explained; Tables 1–2: simplified, specific values removed from the graphs |
| 05 | 02/12/2025 | Minor change. Removed a comment that caused wrong pdf export |
| 06 | 20/03/2026 | Added Chapter 2.5 (Extended Evaluation using D7.4 Dataset); revised Executive Summary and Conclusions. |
| 07 | 20/04/2026 | Corrected minor part of Chapter 2.5 |


| **Project funded from the European Union’s Horizon Europe research and innovation programme** | | |
|---|---|---|
| **Dissemination Level** | | |
| **PU** | Public | x |
| **SEN** | Sensitive – limited under the conditions of the Grant Agreement | |

Start date: 01/06/2024 Duration: 16 months

## ACKNOWLEDGEMENTS

This project has received funding from the Europe's Rail Joint Undertaking (ERJU) under the Grant Agreement no. 101102001. The JU receives support from the European Union's Horizon Europe research and innovation programme and the Europe's Rail JU members other than the Union.

## REPORT CONTRIBUTORS

| Name | Company | Details of Contribution |
|---|---|---|
| Dr. Philipp Neumaier | DB | Content creator |
| Dr. Volker Eiselein | DB | Content creator |
| Kameliya Taneva | DB | Content creator |
| Sebastian Dubiel | DB | Content creator |
| Louis Delvig | SNCF | Expert Review |
| Philippe Bourdenet | SNCF | Expert Review |
| Dr. Cedrick Lelionnais | SNCF | Expert Review |
| Michele Bruzzo | Hitachi | Expert Review |
| Hjalmar Boulouednine | SMO | Expert Review |
| Saro Thiyagarajan | FT/Wabtec | Expert Review |
| Mark Koopman | NS | Expert Review |
| Adapa Saikrishna | Hitachi | Expert Review |
| Daniel Blanco Martin | Adif | Expert Review |
| Jose Antonio Arcos Mena | Adif | Expert Review |

**Disclaimer**

The information in this document is provided “as is”, and no guarantee or warranty is given that the information is fit for any particular purpose. The content of this document reflects only the author’s view – the Joint Undertaking is not responsible for any use that may be made of the information it contains. The users use the information at their sole risk and liability.

# EXECUTIVE SUMMARY

The European rail sector is undergoing a decisive technological shift toward higher levels of automation (Grades of Automation 3 and 4). At the forefront of this transformation stands the Rail to Digital Automated up to autonomous Train Operation (R2DATO) project. This report (D7.5) marks a critical milestone in the project, serving as a bridge between the conceptual design of the so-called Data Factory and its operational implementation. It details the successful development and validation of an integrated machine learning (ML) toolchain capable of transforming annotated sensor data into high-performance, domain-specific artificial intelligence (AI) functions—essential components for robust perception systems in automated train operations.

The methodological foundation of this work is a scenario-based development framework that systematically links operational requirements with data acquisition, simulation, and model training. To evaluate this approach, a well-established object detection model (Faster R-CNN) was adopted as a baseline and subsequently fine-tuned using two publicly available datasets: Open Sensor Data for Rail 2023 (Tilly, et al., 2023) and the Görlitz Rail Test Center CV Dataset 2024 (Rustam Tagiew, 2025). Crucially, to rigorously validate the pipeline's operational readiness against the most recent project outputs, an extended evaluation was conducted using newly acquired, domain-specific data originating from Deliverable D7.4. Model performance was assessed using standardized evaluation metrics, including Intersection over Union (IoU), Precision, Recall, and Mean Average Precision (mAP), providing a rigorous and transparent benchmark of its effectiveness.

The findings from both the original and the extended D7.4 evaluations demonstrate a substantial improvement in model performance across key target classes, clearly surpassing the generic baseline, especially for less common object classes. This outcome serves as a robust proof-of-concept for the Data Factory toolchain, validating its capacity to generate high-quality AI models from multimodal sensor data for safety-critical rail applications. The pronounced gains observed for domain-specific assets—most notably, the fine-tuned model's ability to successfully detect trains in highly challenging D7.4 test scenarios where the pre-trained baseline completely failed (mAP 0.0)—underscore the vital importance of domain-specific training data.

Conversely, the results show slightly inferior performance compared to the baseline for the standard 'person' class. This indicates that for specific, highly common classes where the pre-trained model already possesses sufficiently robust training data, adding more data does not necessarily lead to further improvements and may even introduce slight trade-offs.

Simultaneously, the evaluation exposed the vulnerabilities caused by extreme class imbalances, such as the failure to detect bicycles due to critically low instance numbers (only 2 instances in the D7.4 dataset). These dual insights strongly support the strategic vision of a standardized, shared toolchain, along with standardized, well-curated datasets, as a cornerstone for a collaborative, pan-European Data Factory—particularly in a field where isolated data sources are rarely sufficient to cover all scenarios.

Future research should focus on extending the dataset portfolio to include a broader spectrum of operational scenarios, adverse weather conditions, and non-regular situations (NRS), including the use of synthetic data to mitigate the aforementioned extreme class imbalances. Furthermore, establishing standardized procedures for validation and certification of AI models is essential for their deployment in safety-critical contexts.

# ABBREVIATIONS AND ACRONYMS

| | |
|---|---|
| **AI** | Artificial Intelligence |
| **AP** | Average Precision |
| **ATO** | Automatic Train Operation |
| **bbox_mAP** | Bounding box mean Average Precision |
| **bbox_mAP_50** | Bounding box mean Average Precision at IoU of 0.5 |
| **bbox_mAP_75** | Bounding box mean Average Precision at IoU of 0.75 |
| **BG** | Background (as a class of error) |
| **CC0** | Creative Commons Zero (public domain dedication) |
| **COCO** | Common Objects in Context (dataset) |
| **CV** | Computer Vision |
| **D7.5** | Deliverable 7.5 |
| **DB** | Deutsche Bahn (Company name) |
| **DTP** | Data Touchpoint |
| **DZSF** | German Centre for Rail Traffic Research (Deutsches Zentrum für Schienenverkehrsforschung) |
| **ERJU** | Europe's Rail Joint Undertaking |
| **Faster R-CNN** | A robust object detection model that uses a two-stage process involving a Region Proposal Network (RPN) to generate object proposals and a Fast R-CNN network to refine them |
| **FN** | False Negative |
| **FP** | False Positive |
| **GoA3/4** | Grade of automation 3 / 4 |
| **HPC** | High-Performance Computing |
| **IoU** | Intersection over Union |
| **IR** | Infrared |
| **KPIs** | Key Performance Indicators |
| **LiDAR** | Light detection and ranging |
| **Loc** | Localization (as a class of error) |
| **mAP** | Mean Average Precision |
| **mAP_l** | mAP for large objects |
| **mAP_m** | mAP for medium objects |
| **mAP_s** | mAP for small objects |
| **ML** | Machine Learning |
| **NRS** | Non-Regular Situations |
| **ODD** | Operational Design Domain |
| **OSDaR23** | Open Sensor Data for Rail 2023 |
| **PEDF** | Pan-European Data Factory |

| | |
|---|---|
| **PR-Curves** | Precision-Recall Curves |
| **PTP** | Precision Time Protocol |
| **PU** | Public (dissemination level) |
| **R2DATO** | Rail to Digital automated up to autonomous train operation |
| **RADAR** | Radio detection and ranging |
| **RailGoerl24** | Görlitz Rail Test Center CV Dataset 2024 |
| **RGB** | Red-Green-Blue (visible spectrum colorspace) |
| **RoI** | Region of Interest |
| **RPN** | Region Proposal Network |
| **S3** | Simple Storage Service (AWS) |
| **SEN** | Sensitive (dissemination level) |
| **Sim** | Similar object (as a class of error) |
| **TN** | True Negative |
| **TP** | True Positive |
| **TRL** | Technology Readiness Level |
| **VDL** | Vehicle Data Logger |
| **WP7** | Work Package 7 |

# TABLE OF CONTENTS

## LIST OF FIGURES



## LIST OF TABLES

# 1 INTRODUCTION

## 1.1 PROJECT CONTEXT AND OBJECTIVES

The European rail industry is undergoing a major technological transformation to reach higher levels of automation (Grades of Automation 3 and 4). Leading this effort is the *Rail to Digital Automated up to autonomous Train Operation* (R2DATO) project. This report (D7.5) marks a decisive phase in the project's progression, linking the conceptual design of the so-called *Data Factory* to its operational implementation. It details the successful development and validation of an integrated machine learning (ML) toolchain capable of leveraging annotated sensor data to train high-performance, domain-specific artificial intelligence (AI) functions—critical components for perception systems in safe and reliable automated rail operations.

Work Package 7 (WP7) was specifically established to address this challenge by developing both the conceptual foundations and parts of the technical infrastructure for a Data Factory. This ecosystem is designed to support the systematic collection, processing, annotation, simulation, and distribution of rail sensor data. The idea is to enable a collaboration among stakeholders to overcome the data scarcity that hinders individual actors in the development of AI solutions. Previous project deliverables laid the groundwork for this endeavour: D7.1 defined the architectural and functional requirements of the Data Factory; D7.2 addressed the legal, organizational, and governance frameworks necessary for cross-border data sharing; D7.3 focused on the simulation of rare but safety-critical scenarios through synthetic data; and D7.4 documented the acquisition and annotation of real-world sensor data from operational train environments.

Deliverable D7.5 focuses on training and refining object detectors using publicly available datasets. The goal is to demonstrate the applicability of machine learning methods to rail-specific use cases, particularly for detecting relevant objects in sensor data. Model performance is evaluated using standard metrics, showing that retraining with additional domain-specific data leads to measurable improvements.

## 1.2 D7.5's Place in the R2DATO Data Factory

Deliverable D7.5 builds on the prior work of Work Package 7 by using existing components of the Data Factory to train and refine object detection models. The work relies on publicly available datasets and demonstrates how rail-relevant sensor data can be used to develop AI models tailored to specific detection tasks. This approach lays the groundwork for Deliverable D7.6, which will provide a curated and validated open dataset to support further research and development.

Rather than presenting a new, fully integrated pipeline, D7.5 contributes to the practical validation of previously defined concepts. The focus lies on applying available tooling and infrastructure to train models, evaluate their performance using standard metrics, and analyse the effects of retraining on task-specific improvements.

An important aspect of this work is its iterative structure. Insights from model evaluation—such as weaknesses in certain conditions or object classes—can be used to guide future data collection (D7.4) or simulation efforts (D7.3). This feedback-oriented approach supports the continuous improvement of training data and model performance, aligning with the overall goal of enabling safe, reliable AI applications in rail automation.

## 1.3 Contributions to Prior Work and Novelty

The work detailed in this report distinguishes itself from previous research projects in the railway domain, such as X2Rail-4 and TAURO. While these projects identified the need for and the feasibility of data-driven perception systems, D7.5 moves beyond theoretical frameworks to provide concrete, applied demonstrations. This report contributes several novel aspects:

**Experimental Evaluation with Quantifiable Results:**

D7.5 provides a detailed, experimental evaluation of ML model performance using shared, publicly available datasets as an example of the data which can be used within the Data Factory. The report presents specific, quantifiable results, such as performance gains in mAP, for models refined on railway-specific classes.

**Demonstrating the Practical Value of the Data Factory:**

Deliverable D7.5 provides a practical demonstration of how the Data Factory enables the use of diverse data sources—including publicly available and, in principle, proprietary datasets—for the training and refinement of object detection models in the rail domain. It showcases the feasibility of applying standardized methods to build domain-specific AI systems, even under constraints typical for safety-critical environments.

A key achievement lies in the detailed performance analysis of the trained models. The results allow not only for quantitative benchmarking, but also for the systematic identification of weaknesses—such as reduced accuracy under specific conditions or for certain object classes. This insight is crucial for guiding future data collection and model improvement efforts.

D7.5 is embedded in an iterative, forward-looking development process toward GoA4 automation. While the Data Factory provides the technical foundation, the work also highlights a broader challenge: high-quality, annotated data remains the limiting factor for progress. Sustained investment—especially at the European level—is essential to scale up data acquisition, annotation, and curation, and to unlock the full potential of AI-driven automation in the rail sector.

## 1.4 THE DATA FACTORY TOOLCHAIN FOR MACHINE LEARNING WORKFLOWS

**ML Platform:** This subsystem is the computational core of D7.5. It is designed to offer a comprehensive environment for ML model development, encompassing a range of functions such as the generation of model architectures, model training, and the evaluation of model performance. The ML Platform manages the entire model lifecycle, from versioning and comparison to integration with other platforms. It leverages high-performance computing (HPC) clusters to provide the significant computational resources required for training complex models on large datasets, as demonstrated in the experiments documented here.

**Simulation Platform:** While the ML training experiments in this report primarily focus on real-world datasets, the Simulation Platform, as described in D7.3, plays a critical complementary role. This subsystem is dedicated to generating synthetic sensor data and modelling Non-Regular Situations (NRS) that are too hazardous or infrequent to capture in real life. The ability to create a vast, labelled pool of synthetic data for these edge cases is a strategic necessity for developing AI models that are robust enough to operate safely under all possible conditions. The seamless integration of synthetic data with real-world data is a key enabler for achieving the necessary reliability for GoA4 operations.

**Integration Platform:** The machine learning workflows presented in D7.5 are executed within the existing high-performance computing (HPC) environment of the Data Factory. This compute infrastructure provides the necessary resources and interactive tools for data analysts and ML engineers to carry out model training, evaluation, and refinement. The workflow includes integrated data access, processing, and experimentation capabilities, enabling data set preparation and performance analysis. This cohesive toolchain forms the basis of this data ecosystem. This technical capability provides the foundation for a future Pan-European Data Factory (PEDF) as a collaborative and interoperable platform. This is significant because it shows that the technical work directly supports the legal and strategic vision outlined in D7.2, where the PEDF is framed not as a monopolist, but as a coordinating body that defines common standards and interfaces. The successful model training in D7.5 shows that this coordinated and standardized approach can lead to good results. This may convince potential partners to participate in the PEDF.

## 1.5 Data Inputs and Preparation

The foundation of any high-performing AI model is the quality and diversity of the data used for training.

The training data used in D7.5 was prepared and annotated with high precision, based on the methodology established in Deliverable D7.4. Annotations were created using the *RailLABEL* schema, a railway-specific extension of the ASAM OpenLABEL standard. While D7.5 focuses exclusively on camera-based data, the chosen annotation format is designed to support multimodal sensor inputs and thus forms a scalable foundation for analyses in other modalities within the Data Factory and developments towards fully automated trains.

To ensure data integrity and reliability, several quality assurance measures are implemented. All video streams can be synchronized using the Precision Time Protocol (PTP) to ensure temporal alignment. Objects were assigned unique identifiers to allow consistent tracking across frames, and additional attributes—such as occlusion status and pose—were included to increase the semantic richness of the annotations. This structured approach to data preparation is a key enabler of the model performance achieved in D7.5 and demonstrates the relevance of standardized, domain-specific annotation practices for the development of safety-critical AI systems in the rail sector. For practical GoA4 development work, of course, solid requirements for object detection and tracking across different modalities are needed which can then shape the tasks implemented in the data factory.

## 1.6 Broader Implications and Contribution to European Rail

**Impact on Railway Automation**

The findings of this report constitute an important step toward the realization of safe and reliable perception systems for GoA3/4 operations. The demonstrated ability to train a model to accurately detect critical objects like "person" and "train" in diverse visual environments provides a foundational capability for the autonomous train. The evaluation results and the limitations identified point the way for future developments and demonstrate the need for more data, especially for edge cases. This work moves the project from a theoretical discussion of AI's potential in rail to a practical demonstration of its feasibility. Key in this practical applicability is the iterative approach from data provision over data curation to data analysis which enables feedback loops over time and systematic development of a GoA3/4 system in the context of new or changing requirements.

**Advancing the Pan-European Data Factory**

The technical success of D7.5 strengthens the strategic vision for the Pan-European Data Factory (PEDF). The legal and strategic analysis in D7.2 raised questions about the necessity and potential risks of a centralized data entity, with some partners expressing concern that it might become a data monopolist or disrupt existing business models. The work in this report directly addresses these concerns.

By successfully training a model using publicly available, standardized datasets and a toolchain, this report demonstrates that the PEDF's purpose is not to monopolize data, but to act as a coordinating vehicle that enables collaboration and standardization. The improved performance of the refined model shows that this collaborative approach yields a tangible, competitive advantage for all participants.

**Dependencies and Connections to Other Work Packages**

The output of D7.5 has implications for the R2DATO project. The trained models and performance data will be used to inform related project activities. The models and the insights gained from their evaluation will be provided to WP11, Prototype Development of Perception Systems, for use in their work.

Furthermore, the curated, validated datasets used in the experiments will form the basis for the D7.6, Open-Dataset, which will be released to the public. The success of this work demonstrates the value of the holistic, interconnected strategy of R2DATO, where each deliverable builds on the last to create a cohesive, long-term impact.

## 1.7 Structure of the Document

The document is structured as follows:

- **Chapter 2** summarises the objectives of D7.5 in detail.
- **Chapter 3** outlines the methodology for dataset preparation, model training, and evaluation.
- **Chapter 4** presents the experimental results, including performance metrics and comparative analysis.
- **Chapter 5** discusses the implications of the results, identifies limitations, and outlines open research questions.
- **Chapter 6** provides conclusions and recommendations for subsequent work.

By combining requirements, datasets, and AI experiments, D7.5 provides a critical step towards a Pan-European capability for data-driven rail automation. It underlines the necessity of shared infrastructures like the Data Factory to enable robust, safe, and certifiable AI solutions for GoA3/4.

## 1.8 THE ROLE OF ML IN THE DATA FACTORY

The Data Factory forms the technological backbone of the R2DATO project, enabling future AI-based train operations. Developed within Work Package 7 (D7.1–D7.6), its phased structure builds a coherent, scalable ecosystem designed to meet the strict safety and performance demands of autonomous rail systems.

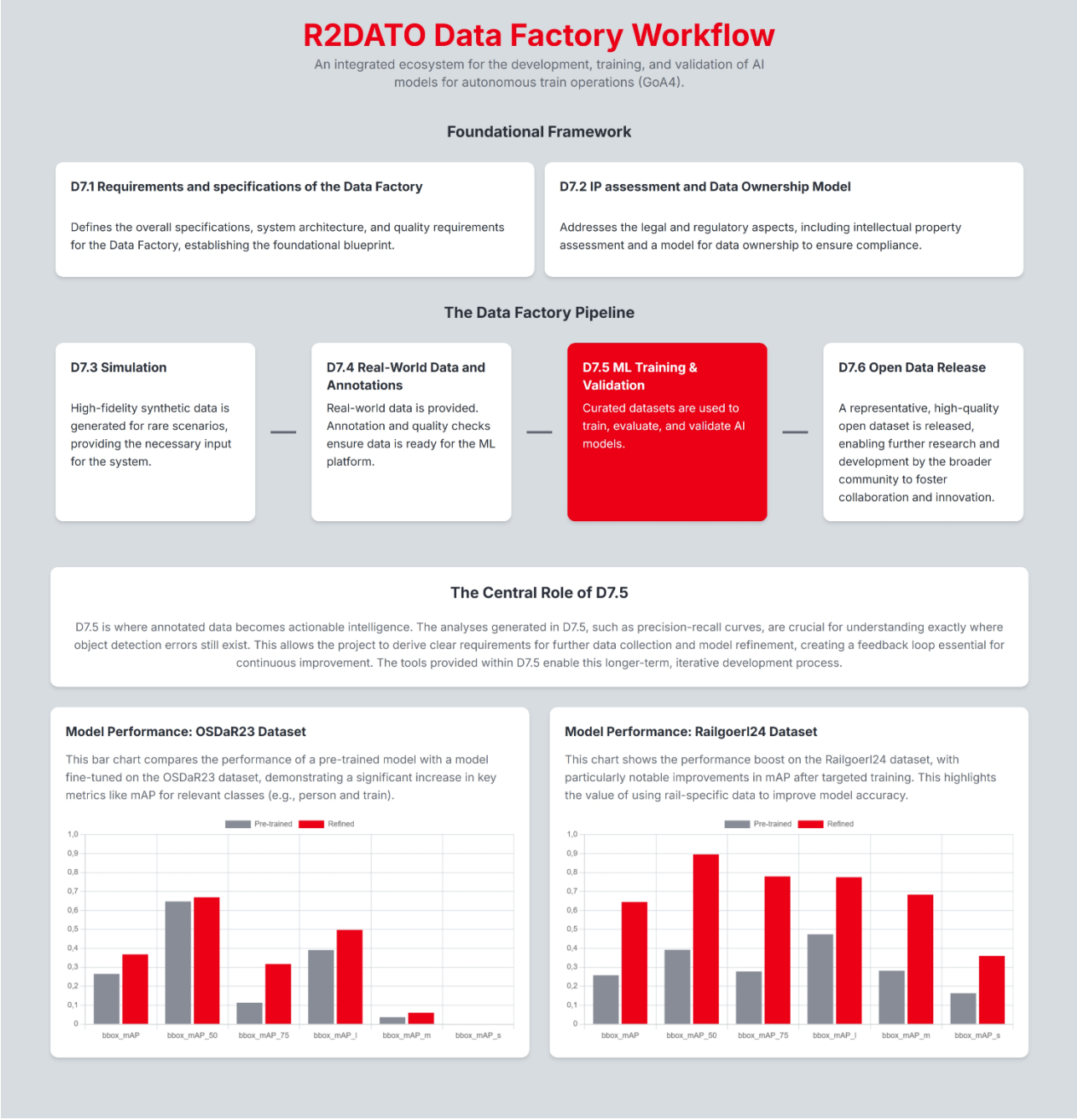


**Figure 1: D7.5 in the Data Factory Context and Workflow**

# 2 MAIN PART

## 2.1 METHODOLOGY ML TRAINING AND DEVELOPMENT

### 2.1.1 Context of this work and reasoning of general approach taken

ATO in the sense of fully automated (i.e. driverless) GoA4[1] train operation can be considered an at least partially autonomous robotics task in a highly regulated and safety-critical domain. As such, it requires high performance standards in the safety of operation because especially in mainline scenarios, driverless trains will need to implement a safe decision-making within the situation-specific degrees of freedom for the train. This means that the generally allowed safe path for a train to avoid collisions with other trains is in the future still likely to be organized and restricted by the interlocking, but there is the need for the train to identify hazardous situations in its immediate surroundings – a task which cannot be taken over by interlockings or other rail operators.

A usual approach derived from similar work in the automotive domain proposes a development alongside the definition of an operational design domain (ODD) (Eichenbaum, 2024) (J. Erz, 2022). In this sense, as part of the architecture specification, the classes of obstacles expected according to the requirements are noted down, and their expected behaviour as well as the driving situations in which they might become relevant all become part of the system specification. The situations might thus comprise properties such as e.g. weather, objects, driving situation, environment etc. which together with their detailed description are usually referred to as “scenarios”, thus enabling methodological approaches like scenario-driven development or scenario-driven testing.

Looking at the example of scenario-driven testing, the method would thus involve going through all scenarios one-by-one and vary the relevant scenario parameters (e.g. weather, object number or position…) in order to derive specific test cases. The coverage of these tests within the scenario is then used as a basis to argue the safety of the final product. While this approach is generally intuitive and appears methodologically straight-forward, it still contains pitfalls such as the vast number of possible scenarios and the specific reasoning about how tests are sampled across or within scenarios. Resolving these questions appears unsolved at the moment and they thus need to be clarified in close collaboration with authorities and regulation bodies for developing a fully automated train or obstacle detection components for it. Nonetheless, scenario-based methods appear to be one of the most promising development approaches in the GoA4 domain.

Within this scenario-based framework, it appears natural to handle the scenarios according to their respective needs. While it would e.g. in principle be feasible to train a generic anomaly detector on all possible hazardous situations, we think a more focused approach considering the specific needs of each scenario is more promising. Its main advantage to generic anomaly detection methods is that it allows to understand specifically the requirements in every scenario and also provides a direct link to suitable training and test data which often supports finding a technological solution. This is

[1] The term “Grade of Automation 4” (GoA4) describes the absence of all personnel on-board a train, i.e. no driver or train attendant is on board and the train can handle all regular and irregular situations according to the specification. It is often also referred to as “unattended driving” of trains.

even more important when you take into consideration the higher amount of data needed for training generic or class-agnostic ML solutions. As another advantage, the approach of detecting specific, different classes allows both the training of a single detector for multiple classes or the usage of multiple detectors for even only one specific class each. In this sense, the framework is also easily extendable in case of new requirements across a system's lifecycle.

With this deliverable, we show the overall usefulness of DB's Data Factory for the aforementioned tasks during the development of fully automated trains. For the training of an obstacle detection system, we consider the case of class-specific object detection. The types of classes considered should be part of the requirements derived for a fully automated train and can thus here only be shown in an exemplary manner.

However, the experiments shown in this deliverable are carefully chosen to show the applicability of the developed toolchain within DB's Data Factory for object detection tasks including training and inference as well as dedicated testing within the development of functions for fully automated trains.

## 2.2 Evaluation metrics for Object Detection: State-of-the-art description

Object detection is a fundamental task in computer vision, with applications ranging from autonomous driving and surveillance to medical imaging and robotics. To evaluate the performance of object detection models, researchers and practitioners typically rely on a set of standardized metrics. These metrics help quantify how well a model can identify and localize objects within images or video frames. Below, the most important concepts and how they relate to each other are explained.

### 2.2.1 Basic Concepts: True/False Positives and Negatives

At the core of object detection evaluation, there are defined four basic terms:

- **True Positive (TP):** A detected object that matches a ground truth object in both class and location (as determined by a bounding box overlap threshold).
- **False Positive (FP):** A detected object that either does not match any ground truth object or does not meet the overlap threshold.
- **False Negative (FN):** A ground truth object that was not detected by the model.
- **True Negative (TN):** Typically, not used in object detection, as the background is usually ignored.

These terms are highly dependent on specific detector settings and it is thus common sense to incorporate them into more complex metrics that describe the performance of an object detection model within a broader range of situations.

### 2.2.2 Bounding Box Matching: Intersection over Union (IoU)

To determine whether a detected bounding box is a true or false positive, we use the **Intersection over Union (IoU)** metric. IoU measures the overlap between a predicted bounding box and a ground truth bounding box:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

- The **Area of Overlap** is the area where the predicted and ground truth boxes intersect.
- The **Area of Union** is the total area covered by both boxes.

A detection is considered a true positive if the IoU exceeds a predefined threshold (commonly 0.5). If the IoU is below this threshold, the detection is counted as a false positive.

### 2.2.3 Matching Procedure

The evaluation per class is conducted in the following steps:

- The inputs are ground truth bounding boxes for all classes and predictions for the respective class
- For every predicted result bounding box, the following steps are conducted:
    - If it overlaps with a ground truth bounding box of the correct class, it is considered a **true positive** (in case of sufficient overlap) or a **localization error** (in case of too small overlap)
    - If it overlaps with a bounding of another class, it is considered a **similar object** (in case of the same object supercategory) or **"other" object** (in case of different supercategory)
    - If there is no overlapping ground truth object, the detection is considered a **false positive** ("background").

### 2.2.4 Precision and Recall

**Precision** and **Recall** are two fundamental metrics derived from the counts of true positives, false positives, and false negatives:

- **Precision** measures the proportion of correct detections among all detections made by the model:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

High precision means that when the model predicts an object, it is likely correct.

- **Recall** (also called sensitivity) measures the proportion of ground truth objects that were correctly detected:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

High recall means that the model misses only few actual objects.

There is often a trade-off between precision and recall: increasing one may decrease the other. For example, lowering the confidence threshold for detections can increase recall (fewer false negatives) but may also increase false positives, thus reducing precision.

### 2.2.5 Precision-Recall Curve and Average Precision (AP)

To visualize the trade-off between precision and recall, we use the **Precision-Recall Curve**. This curve is generated by varying the confidence threshold of the model and plotting the corresponding precision and recall values.

- The **Average Precision (AP)** is the mean over the precision-recall curve for the class(es) tested. It summarizes the model's performance across all confidence thresholds.

### 2.2.6 Mean Average Precision (mAP)

The **mean Average Precision (mAP)** is the average of AP values across all classes. It provides a single-number summary of a model's performance, making it easier to compare different models or configurations.

- **COCO mAP:** The COCO dataset ("Common Objects in Context", (Tsung-Yi Lin, 2014)) uses an extended version of mAP, averaging AP over IoU thresholds from 0.5 to 0.95, in steps of 0.05. This provides a more robust evaluation across different levels of localization accuracy.
- **AP@IoU=0.5:** This is the AP calculated at a single IoU threshold of 0.5, which was the standard in the PASCAL VOC challenge.

### 2.2.7 Relationships Between Metrics

- **Accuracy** is generally not used in object detection because it does not account for the imbalance between the number of objects and the background.
- **Precision** focuses on the quality of positive detections, while **recall** focuses on the quantity of detected objects.
- **mAP** combines both precision and recall across all classes and IoU thresholds, providing a balanced measure of overall performance.

### 2.2.8 Practical Example

Suppose we have a model detecting pedestrians in a video:

- If the model detects 90 pedestrians, but only 80 are correct (TP) and 10 are incorrect (FP), and there are 100 actual pedestrians (so 20 are missed, FN), then:
    - Precision = 80 / (80 + 10) = 0.89
    - Recall = 80 / (80 + 20) = 0.80

The precision-recall curve would show how these values change as the confidence threshold is adjusted. The area under the curve formed by varying the confidence threshold then finally relates to the mAP score that combines the different measures into a single performance value.

## 2.3 DATASETS USED IN THIS DELIVERABLE

It must be noted that developments in the field of digitalization of the rail domain currently suffer from an enormous lack of data. Especially tasks such as object detection and digitalization of infrastructure tasks require large visual datasets for training and testing the systems developed. This problem is even more severe for multi-modal datasets containing not only camera but also other modalities such as LiDAR or RADAR. Unfortunately, due to the high costs and efforts needed to record, curate and annotate such data accurately, there is currently no “gold standard” dataset available to develop perception functions on realistic rail data.

In this work, we focus our evaluation thus on publicly available data and emphasize the need to foster initiatives providing more and better data within the rail domain.

### 2.3.1 Open Sensor Data for Rail 2023 (OSDaR23)

The Open Sensor Data for Rail 2023 (OSDaR23) dataset (Tilly, et al., 2023) is the first publicly available multi-sensor dataset specifically annotated for railway environments. It was created through a joint research project by the German Centre for Rail Traffic Research at the Federal Railway Authority (DZSF), Digitale Schiene Deutschland / DB Netz AG (now DB InfraGO), and FusionSystems GmbH. The dataset is designed to support the development of AI algorithms for automated train operation, particularly for object detection and environment perception in mainline railway contexts.

Key features of OSDaR23 are:

- **Multi-sensor setup**: Includes cameras, LiDAR, radar, and position/acceleration sensors, providing a comprehensive view of the railway environment.
- **Annotated data**: Contains a variety of object classes relevant to railway safety, such as pedestrians, vehicles, infrastructure, and other potential obstacles.
- **Real-world scenarios**: Recorded in Hamburg, Germany, covering diverse railway situations and environments.
- **Data format**: Annotations are provided in JSON format and are published under the CC0 1.0 license, allowing for broad use in research and development.

OSDaR23 is intended to address the lack of publicly available railway datasets and to advance the development of automated driving and safety systems for railways. It is available for download on the official FID move platform https://data.fid-move.de/dataset/osdar23.

Some sample RGB images for OSDaR23 are shown in Figure 2, we refrain from showing other modalities such as IR camera, LiDAR or RADAR because these are not relevant for this deliverable. Contrast for OSDaR is visibly rather high, also, due to constant exposure settings across the data, some areas in images can appear very bright while in other images, darker parts are more difficult to perceive.

An overview of the classes contained in OSDaR23 can be found in Figure 3. Generally, the dataset shows a lot of variation which is very positive. On the other hand, the annotations include also objects which are partially occluded (e.g. pedestrians behind a column or in a crowd) and thus hard to use as training samples for ML models. In our evaluation, we have thus omitted annotations with a degree of occlusion > 0.5. Super-categories are not provided in OSDaR23, meaning that the different object classes are not grouped into super-categories which will be relevant for evaluation (all objects are of type “object”). On this dataset we evaluate on the classes “person” and “train” (motorcycles not used for a very small number of samples).

The dataset contains 45 scenes. For the tests we use 38 for training and 7 for testing/validation.

**Figure 2: Sample images (OSDaR23)**

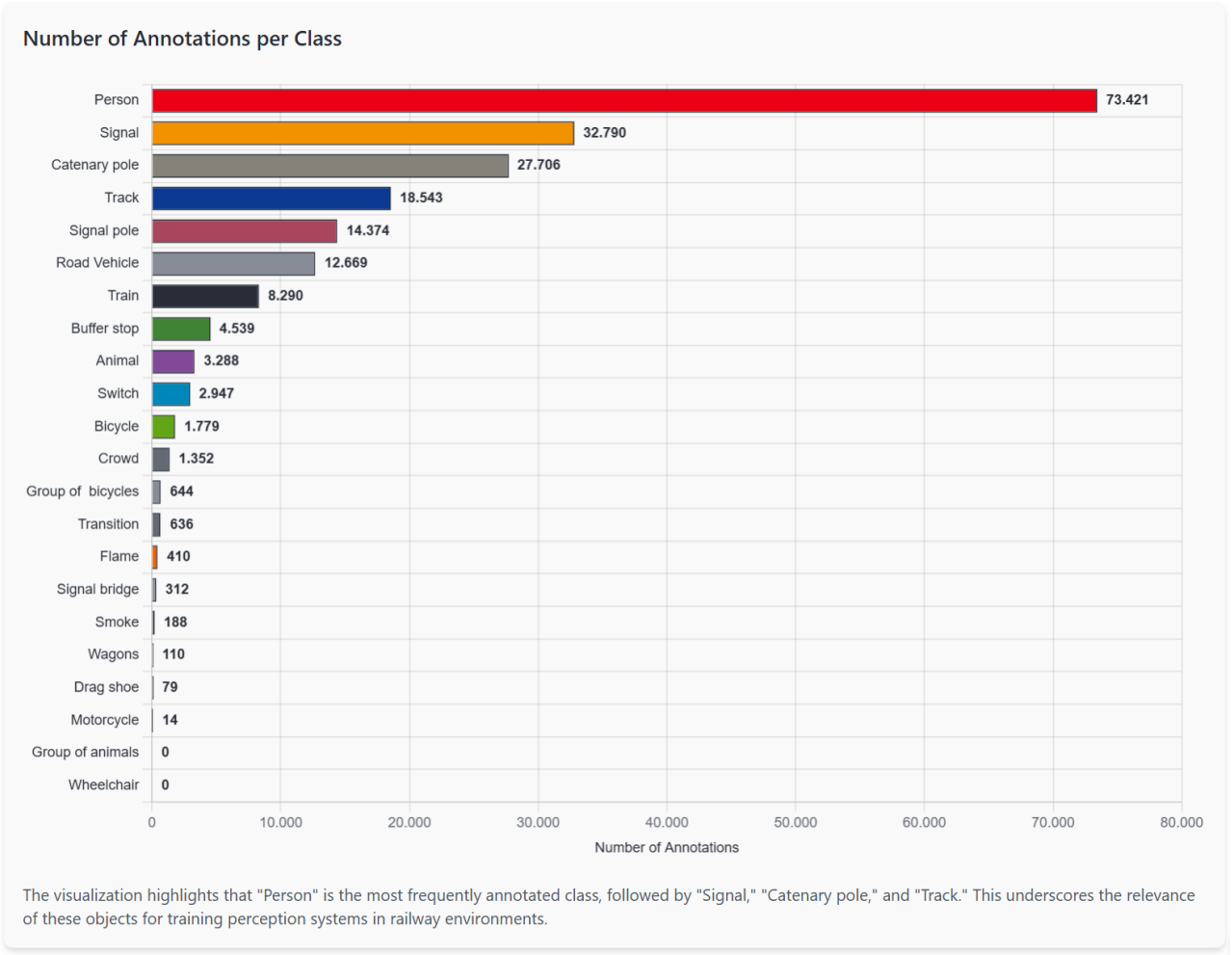


**Figure 3: Distribution of object classes (OSDaR23)**

### 2.3.2 Görlitz Rail Test Center CV Dataset 2024 (RailGoerl24)

The Görlitz Rail Test Center CV Dataset 2024 (RailGoerl24) dataset (Rustam Tagiew, 2025), available at https://data.fid-move.de/dataset/railgoerl24, is an on-board visual light Full HD camera dataset specifically designed to support the development of machine learning-based person detection systems for railway applications.

It consists of 12,205 frames and includes 33,555 box-wise annotations for the object class "person" and 298 for the class “bicycle”. The dataset was recorded at the Görlitz Rail Test Center in Germany and is intended to facilitate tasks such as collision prediction, automated security surveillance, and occupational safety in both mainline and urban railway environments.

Key features of RailGoerl24 include:

- It is one of the few open annotated datasets recorded on-board for mainline railways.
- The dataset includes a terrestrial LiDAR scan covering parts of the area used to acquire the RGB data.
- Faces of recorded actors are not blurred or altered so pedestrian detection algorithms are trained and tested on representative images.
- The data is collected from a reversing train for safety reasons, so it is not suitable for forward-moving train scenarios in sequence-based computer vision applications.

This dataset seems particularly valuable for developing systems aimed at detecting trespassers and ensuring safety in automated railway operations. Some exemplary images from the dataset are shown in Figure 4.

As for the OSDaR23 case, the LiDAR data available for some sequences is not shown because it is not used in this work. It is visible that the color and contrast in the images are somewhat different from OSDaR23 but we used the images as they are without any post-filtering.

Another issue of RailGoerl24 is the rather small number of object annotations. For the object class “bicycle”, only two video sequences are available – leading to the fact that the variation of objects in this class is very limited. One of the major differences is that in one sequence, the bike is standing on the tracks while in the other one it is lying on the rails. This of course is a slight variation but the remaining parameters of the videos are very similar.

While we still incorporate this object class for our evaluation for transparency reasons and to show the overall usefulness of the approach taken, further data would be beneficial to improve the expressive value of the analysis. As for OSDaR23, super-categories are not provided in Railgoerl24, all objects are of type “object”. For our tests, similar to OSDaR23, we use two classes, i.e. person and bicycle even if the sample count for bicycle is very low. We think it is important to show results not only on a single class of objects, however, this underscores again the need for more publicly available datasets in the rail domain.

The dataset contains 61 scenes. For the tests we use 49 for training and 12 for testing/validation.


**Figure 4: Sample images from RailGoerl24 dataset**

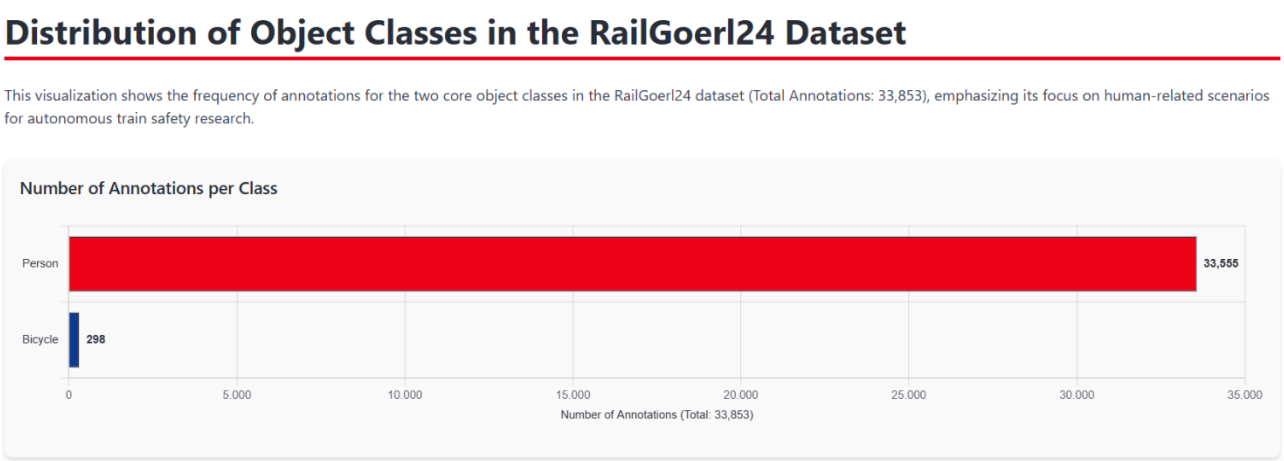


**Figure 5: Distribution of Object Classes in the RailGoerl24 Dataset**

## 2.4 RESULTS AND PERFORMANCE ASSESSMENT

In this section a closer look at the experiments and outcomes of the ML training is taken. We start using a pre-trained model and refine it on the respective datasets which allows us to assess the performance gains for the detection of different object classes on each dataset.

### 2.4.1 Baseline model

As a baseline model, we use a Faster R-CNN_R50 architecture (Shaoqing Ren, 2015). Its architecture is based on a region proposal network (RPN): In order to reduce computational complexity, a fully convolutional network that simultaneously predicts object bounds and objectness scores at each position generates anchors (reference boxes) at multiple scales and aspect ratios to handle objects of various sizes.

The actual detection is done in a Fast R-CNN detection network (Girshick, 2015) which takes the proposals from the RPN, extracts features using RoI (Region of Interest) pooling, and performs classification and bounding-box regression. To simplify the architecture and to reduce computational overhead, shared convolutional features are used, i.e both the RPN and the Fast R-CNN detection network use the same convolutional layers, thus making the system efficient and fast. The entire network is trained jointly, allowing the RPN and Fast R-CNN to share features and be optimized together, which improves both accuracy and speed.

While the model itself was published already in 2015, it still gives good results. Especially when considering training/execution time and computational complexity, its architecture which significantly reduces the computational overhead of region proposal generation, leads to near real-time object detection performance – a requirement which can be safely assumed also for driverless trains.

For our experiments, we use a model which is pretrained on COCO (Tsung-Yi Lin, 2014). Again, while we would have preferred using rail-specific data here, this dataset can be considered a standard dataset for visual object detection tasks and is often used also e.g. in automotive or other perception and object detection domains. It contains photos of 91 object types and a total of 2.5 million annotated samples. However, many of the object classes are not relevant for the rail domain, so we do not use them in his deliverable but focus on classes we have in RailGoerl24 and OSDaR23.

### 2.4.2 Refining the model on OSDaR23 dataset

The evaluation on OSDaR23 is done using the classes person and train in order to show performance on moving objects with high relevance for railway scenarios and with a high numerical significance (i.e. a high number of samples). Unfortunately, appearance of road vehicles in OSDaR23 is often irrelevant for fully-automated driving because there is no risk for collision/near-collision. On the other hand, persons e.g. on platforms are always relevant for decision-making on the vehicle and trains run always on tracks and could thus pose a risk for collision.

Numerical results can be seen in Table 1. Figure 6

The mAP values are based on bounding box matching (“bbox_mAP”) are averaged over both classes “person” and “train”. “bbox_mAP_50” and “bbox_mAP_75” indicate the values at an IOU of >=0.5 and >=0.75, respectively.

An additional evaluation is done based on the size of objects: The table reports values for large ("mAP_l", area A >= 9216 pixels) and medium-sized objects ("m_AP_m", 1024 <= A < 9216 pixels). Small objects ("m_AP_s") have a size of A < 1024 pixels and are virtually non-existent in this dataset.

This explains that for small objects, no performance improvement is obtained. If the pre-trained model does not detect them and no such objects are in the training data, no significant progress can be expected after refinement.

It is visible that the refinement of the model on the OSDaR23 data yields a big improvement over the pre-trained model. In every category, the values have improved compared to the baseline version. Bigger gains can be reported for IOU >= 0.75 and for large objects. This is in line with the expectations because the dataset contains mostly larger objects.

## Object Detection Metrics: Refined vs. Pre-trained Model (OSDaR23)

The table below compares key object detection metrics for a model on the OSDaR23 dataset. It highlights the performance of a pre-trained model versus a refined model, showcasing the impact of dedicated, domain-specific training. The final column quantifies the improvement.

| Metrics | Pre-trained | Refined | Difference (=Refined - Pre-trained) |
|---|---|---|---|
| bbox_mAP | 0.265 | 0.368 | **0.103** |
| bbox_mAP_50 | 0.647 | 0.669 | **0.022** |
| bbox_mAP_75 | 0.113 | 0.317 | **0.204** |
| bbox_mAP_l | 0.391 | 0.497 | **0.106** |
| bbox_mAP_m | 0.037 | 0.060 | **0.023** |
| bbox_mAP_s | 0.000 | 0.000 | **0.000** |

**Table 1: Object detection metrics and improvement for refined vs. pre-trained model (OSDaR23)**

# Object Detection Metrics on OSDaR23 Dataset

This visualization shows the performance improvement of a model after being refined on the OSDaR23 dataset, compared to its pre-trained state. The data illustrates how fine-tuning a model with a domain-specific dataset can lead to significant increases in Mean Average Precision (mAP) for railway-relevant object classes.

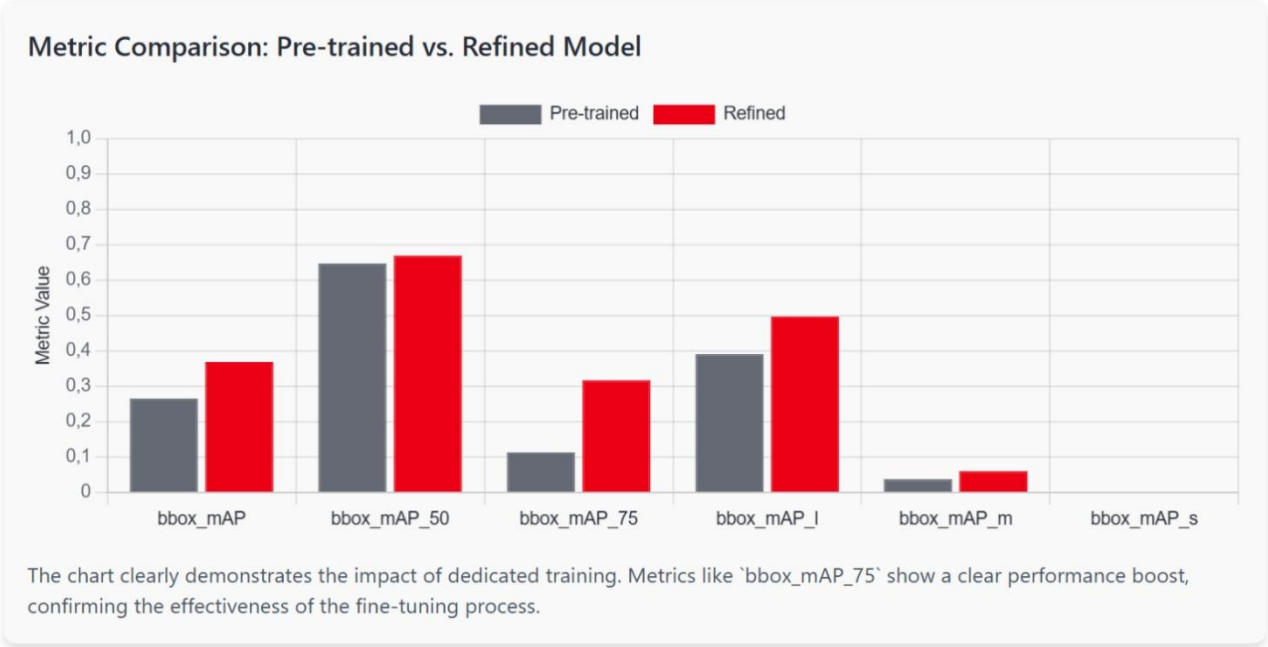


**Figure 6: Object detection metrics and improvement for pre-trained vs. refined model (OSDaR23, all classes)**

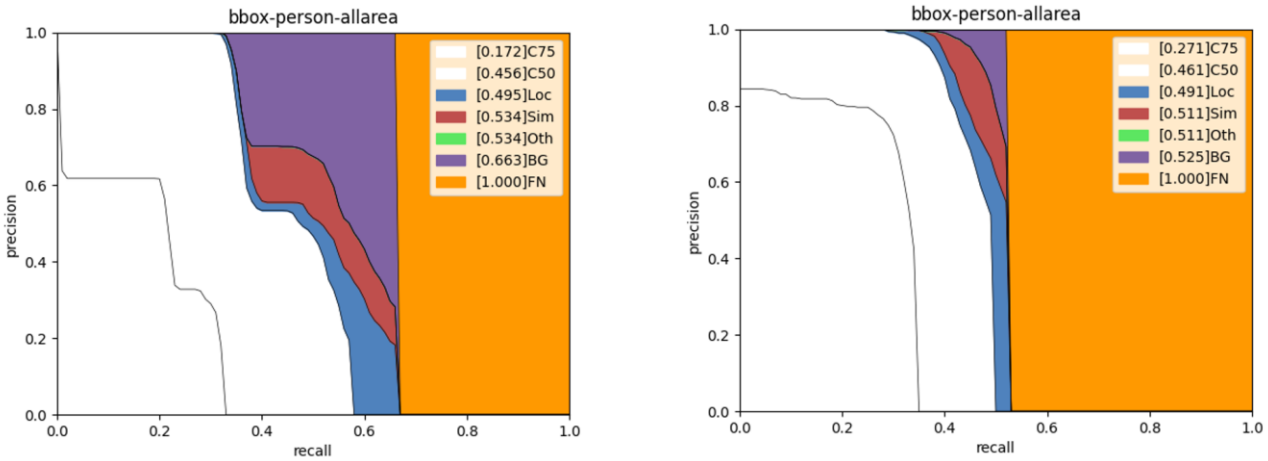


**Figure 7: PR-Curves - Pre-trained model (left) vs. model refined on data (right), person class**

Further results are visualized in Figure 7. The stacked plot shows different metrics in one chart:

- **C75**: AP metric result at IOU threshold of 0.75
- **C50**: AP metric result at IOU threshold of 0.5
- **Loc**: AP metric when ignoring localization errors (-> there is an overlap to the ground truth but too small to be considered a match)

- **Sim**: AP metric when ignoring false positives of the same object supercategory (due to only one supercategory in the data, this is for our experiments the same as “Oth”)
- **Oth**: AP metric when ignoring additionally class confusions across supercategories
- **BG**: AP metric when ignoring additionally all background (= false positive) detections
- **FN**: AP metric when ignoring additionally all false negatives

The plot allows for an assessment of improvements according to the errors in the different categories. It is part of DB Data Factory’s toolset integrated to inspect datasets and models and enables a developer to quickly understand in an intuitive manner where problems with a ML model are: The bigger the respective coloured areas in the image, the more prominent the respective error in the system. Bigger white areas indicate a better performance for the respective metric. When reading the diagram, please consider that the values in the legend are cumulative, therefore the size e.g. of the “BG error area” in Figure 7 (left) is 0.495 - 0.456 = 0.039.

In this evaluation, it is visible that the share of class confusions (“Oth”) among the errors is negligeable both for the baseline and the refined model. Given that bounding boxes of trains can cover large parts of the image, there is the problem of proper evaluation of other objects in these areas. In this evaluation procedure, e.g. a false-positive person could be associated with a train ground truth and would then lead to a higher “Sim” error and not a higher “BG” error.

After training, the BG (false-positive) detections are greatly reduced, and the effect of localization issues is slightly smaller. AP values both for IOU>=0.5 and IOU>=0.75 are higher than in the pre-trained model which again indicates a general improvement after training. The share of errors related to false negative detections is generally higher after training which means that the overall performance was improved especially by reducing false positive detections.

In summary, this visualization is a very helpful tool for analysing the outcome of the training. We would like to emphasize that this evaluation is not about training a perfect machine learning model for perception in the rail domain. Instead, we show with this evaluation that within DB’s Data Factory, the tools are available to understand in depth where the errors of the system are and how they can be visualized to understand the next steps needed for improvement.

It should also be mentioned that in a typical system setup, camera object detection would only be one function among others and obstacle detection in other modalities would certainly be needed

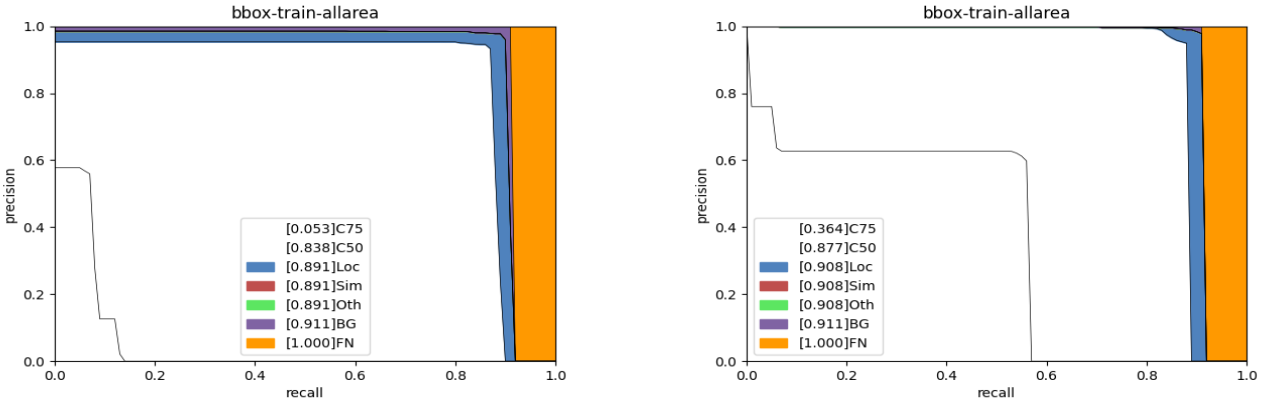


**Figure 8: PR-Curves - Pre-trained model (left) vs. model refined on data (right), train class**

already for cases with unfavourable environmental conditions such as low-lighting (e.g. night-time or tunnel environment) or adverse weather conditions (e.g. heavy rain or fog). Also, it would make sense to integrate sensor fusion and tracking of objects which both can additionally benefit to the reduction of false negative detections.

The results for the train class are shown in Figure 8. Again, as for the person class before we show the errors related to different error categories. The AP for IOU >=0.75 is greatly improved by the training. False positive detections (“BG”) and localization errors are reduced significantly. False negative errors remain on a similar level, and class (“Oth”) / super category (“Sim”) confusions are negligeable.

The general detection performance compared to the person case is on a higher level, also false negatives are much less frequent. Again, all evaluations have been conducted only on the detector level and could be improved in the system by adding sensor fusion, tracking or processing of other sensor modalities.

Exemplary detection results are shown in Figure 9. The left images depict ground truth annotations, right images show detection results. Results from the pre-trained model (first row) contain a false positive detection in the pretrained model, which has been corrected in the refined model (second row) using additional training.

**Figure 9: Exemplary detection results OSDaR23**
**(left: ground truth, right: detections. 1st row pre-trained model, 2nd row refined model)**

### 2.4.3 Refining the model on RailGoerl24 dataset

In addition to the experiments conducted on OSDaR23, we also evaluated the recently published RailGoerl24 dataset on all its classes, i.e. persons and bicycles. The results are shown in Table 2 / Figure 10.

After refining the model, there are improvements in both classes and thus over the whole dataset. Regardless of the metric chosen, the object size or the class, the refined model outperforms the pre-trained one in all evaluations. The improvements and overall gains are significant and show that the refinement on RailGoerl data led to much better detection performance.

## Object Detection Metrics: Refined vs. Pre-trained Model (RailGoerl24)

The table below compares key object detection metrics for a model on the RailGoerl24 dataset. It highlights the performance of a pre-trained model versus a refined model, showcasing the impact of dedicated, domain-specific training. The final column quantifies the improvement.

| Metrics | Pre-trained | Refined | Difference (=Refined - Pre-trained) |
|---|---|---|---|
| bbox_mAP | 0.258 | 0.644 | 0.386 |
| bbox_mAP_50 | 0.392 | 0.895 | 0.503 |
| bbox_mAP_75 | 0.278 | 0.779 | 0.501 |
| bbox_mAP_l | 0.474 | 0.775 | 0.301 |
| bbox_mAP_m | 0.282 | 0.683 | 0.401 |
| bbox_mAP_s | 0.163 | 0.360 | 0.197 |

**Table 2: Object detection metrics and improvement for refined vs. pre-trained model (RailGoerl24, all classes)**

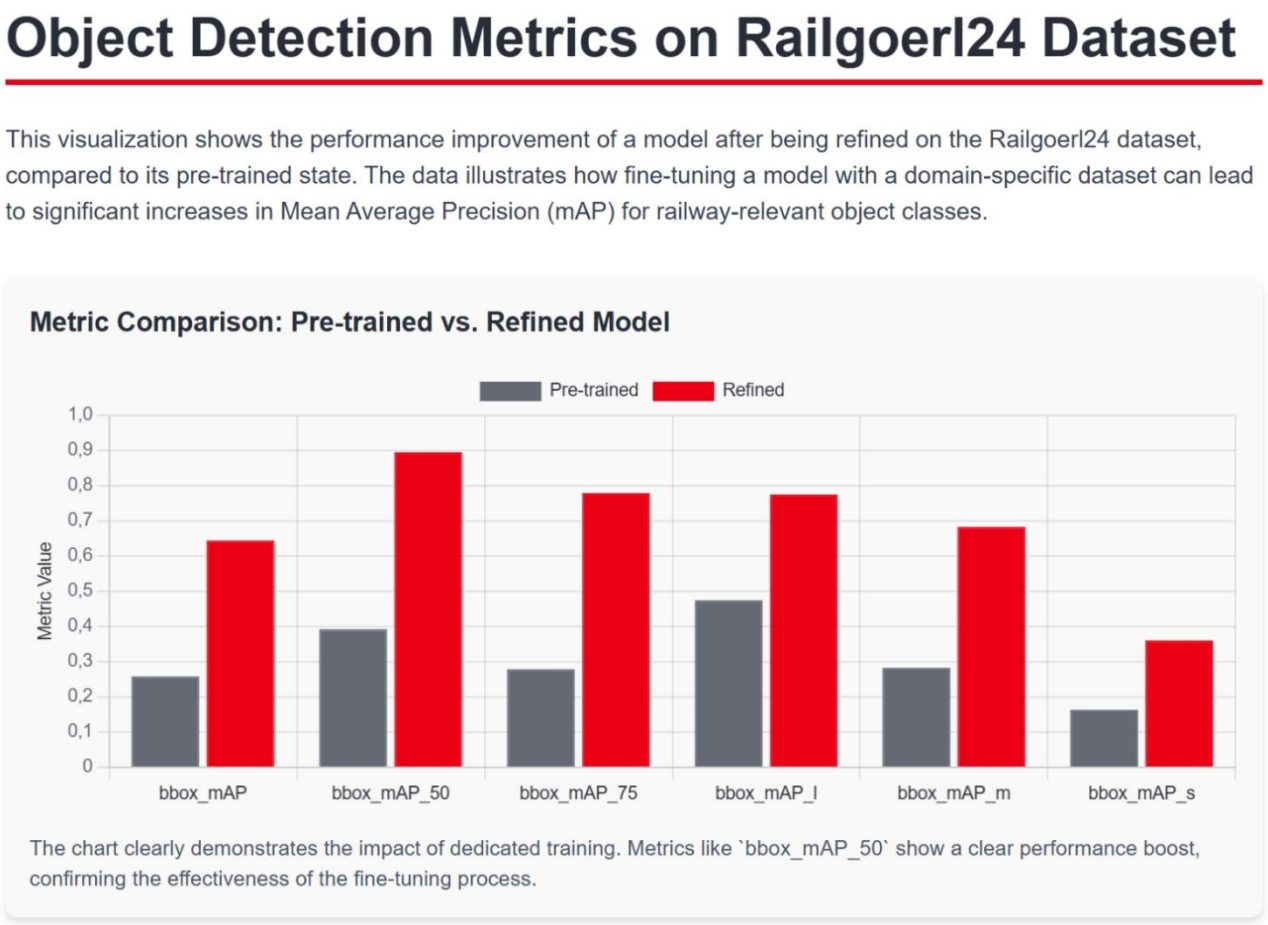


**Figure 10: Object detection metrics and improvement for pre-trained vs. refined model (RailGoerl24, all classes)**

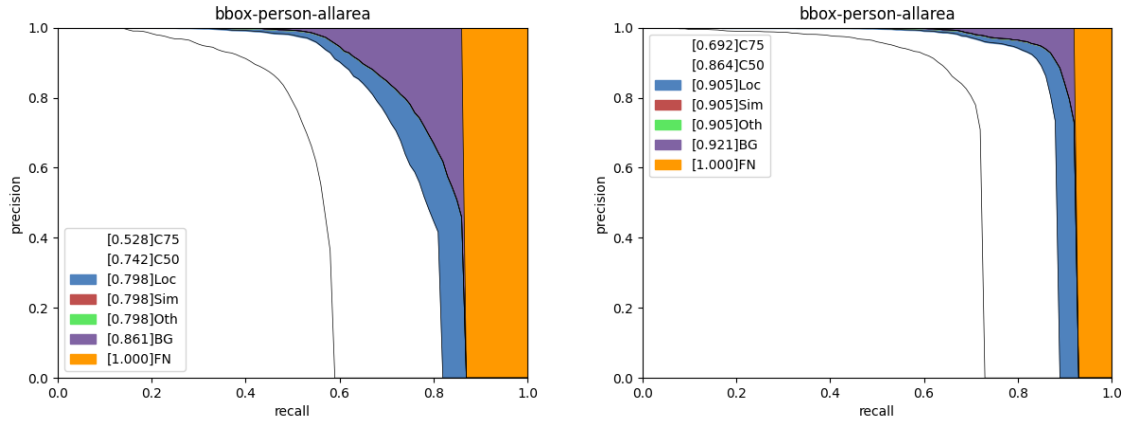


**Figure 11: PR-Curves RailGoerl24 - Pretrained Model (left) vs. Trained Model (right), person class**

Further results are shown in Figure 11 as a stacked PR-curve diagram. The refined model (right) shows a lower number of false negatives, less localization errors and a reduction of false positive detections. Class or supercategory confusion is not relevant in the testcase shown, neither for the pre-trained system nor for the refined detector.

PR-curves for the bicycle class are shown in Figure 12. The earlier mentioned reduction in false negatives is clearly visible as the orange area for the refined model is much smaller. Also, false positive detections are reduced in the refined detector. Localization errors and class/supercategory confusions are on a similar scale for both models.

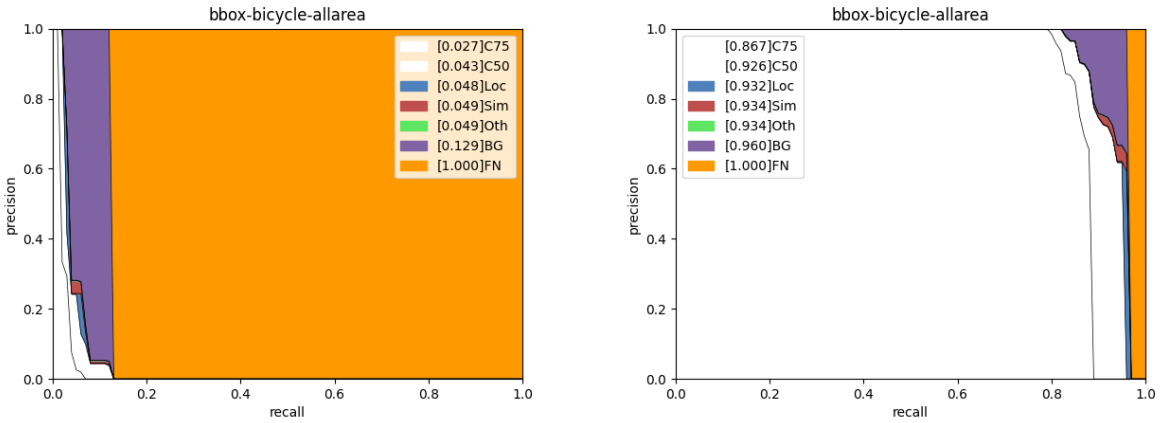


**Figure 12: PR-Curves RailGoerl24 - Pretrained Model (left) vs. Trained Model (right), bicycle class**

The most prominent improvement of the detection performance in this dataset is visible in the "bicycle" class. While there is of course a high degree of dissimilarity for the bicycle pictures in this benchmark compared to the COCO dataset, it is also possible that the pre-trained detector model learned more about the context of the object than about the actual representation of the object itself. In this case, bicycles on rails would be in a highly different context than within the COCO training data. If the model has learned to expect bicycles e.g. on a road or next to persons, this could be a hint to so-called "Clever Hans" behaviour (Sebastian Lapuschkin, 2019) and would be something to consider and investigate for a real product development.

Exemplary detections are visible in Figure 13 for the bicycle class. The pre-trained model misses the bicycle, while the refined model detects it. Also, the pre-trained model produces an additional false-positive person detection.

Figure 14 shows exemplary detection results for persons. They are very far from the camera but while the pre-trained detector misses them, the refined model manages to detect both persons correctly.

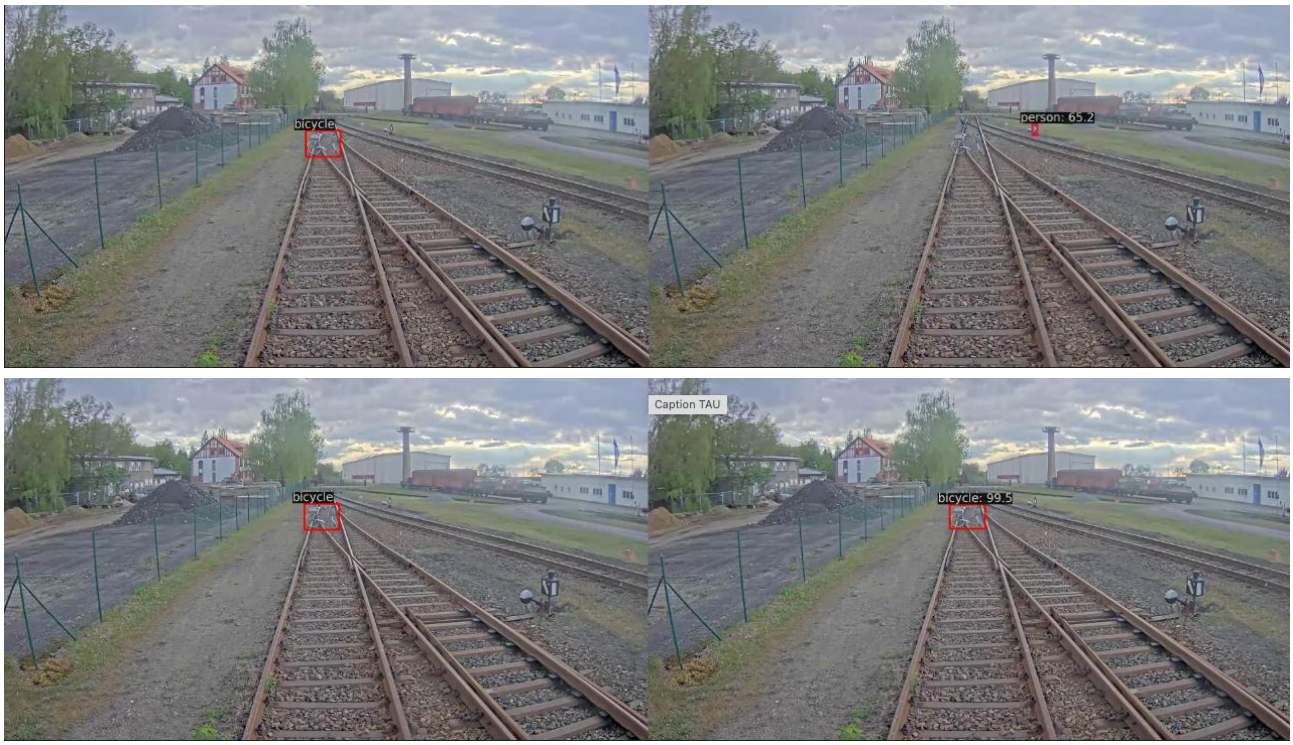


**Figure 13: Example detections RailGoerl24 (bicycle class),**
**left: ground truth, right: detections. Pre-trained model (1st row) vs. refined model (2nd row)**


**Figure 14: Example detections RailGoerl24 (person class),**
**left: ground truth, right: detections. Pre-trained model (1st row) vs. refined model (2nd row)**

### 2.4.4 Conclusion of the evaluation

The experiments shown in the deliverable underline the usefulness of the DB Data Factory for developing data-driven use cases and systems, such as e.g. perception systems for automated rail operation.

While there are still not enough datasets available for such use cases in the rail domain, the experiments have shown that it is possible to train and evaluate a machine learning model for obstacle detection and to understand both the requirements fulfilled in the data as well as the performance details needed to iteratively improve the system. It has been shown how data can be used within the Data Factory for training and testing of a machine learning system, and compared to a pre-trained baseline model, the refined detectors yield better results and can be analysed e.g. in terms of false positive or false negative detections, location errors and other properties needed for development.

It would be beneficial to support further work in the area of generating and curating standardized datasets for fully automated driving on the rail in order to address the lack of data in the domain.

## 2.5 EXTENDED EVALUATION USING D7.4 DATASET

### 2.5.1 Context and Data Selection

An extended evaluation was conducted to validate the baseline and fine-tuned models using newly acquired and annotated data originating from Deliverable D7.4. This step ensures that the models are assessed against the most recent and relevant data sources produced within the R2DATO Data Factory framework.

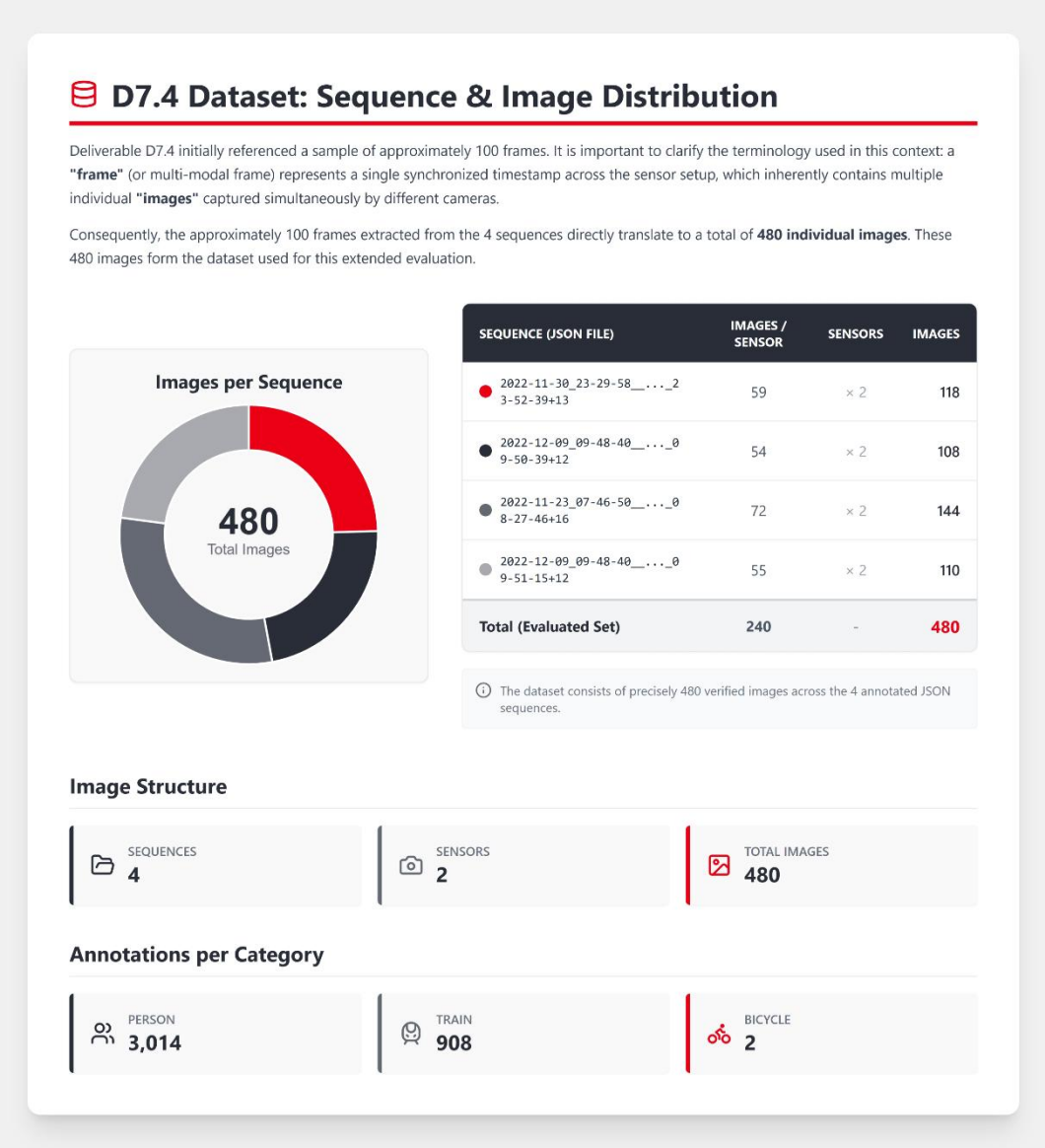


Figure 15 D7.4 Dataset: Sequence & Image Distribution (DB)

Note on Data Inclusion: The evaluation specifically utilizes data generated and annotated by DB. Data provided by SNCF was excluded from this specific metric calculation. This decision was necessary because the SNCF data was annotated using a different methodology and schema, which would have required extensive mapping and harmonization to ensure a fair and consistent evaluation baseline. This highlights the critical importance of standardized annotation schemas (as addressed in D7.6) for cross-border AI training.

### 2.5.2 Quantitative Results

In addition to the already conducted experiments, the models were evaluated against the newly compiled D7.4 DB test set to compare their performance across the previously reported test domains (OSDaR and RailGoerl). For clarity regarding this distinction: the evaluation compares a generic pre-trained baseline against two distinct fine-tuned models—one model that was specifically fine-tuned on the OSDaR dataset, and another model that was specifically fine-tuned on the RailGoerl dataset.

| Model | Category | mAP | mAP_50 | mAP_75 | mAP_s | mAP_m | mAP_l |
|---|---|---|---|---|---|---|---|
| **Fine-Tuned** | Person | 0.281 | 0.437 | 0.320 | 0.000 | 0.057 | 0.442 |
| | Train | 0.027 | 0.077 | 0.015 | 0.000 | 0.049 | 0.000 |
| | *Overall (bbox_mAP)* | *0.154* | *0.257* | *0.167* | *0.000* | *0.053* | *0.221* |
| **Pre-Trained** | Person | 0.345 | 0.617 | 0.344 | 0.000 | 0.146 | 0.483 |
| | Train | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| | *Overall (bbox_mAP)* | *0.173* | *0.308* | *0.172* | *0.000* | *0.146* | *0.214* |

Table 3 Performance Comparison on D7.4 Data (OSDaR Classes)

Interpretation: In the specific comparison between the pre-trained baseline and the model fine-tuned on the OSDaR dataset, it is evident that while the pre-trained model exhibits a higher overall mAP for the 'person' class, it completely fails to detect the 'train' class (mAP 0.0). The OSDaR-fine-tuned model successfully introduces a basic capability to detect trains (mAP 0.027), albeit with a slight trade-off in person detection performance on this specific dataset slice. The reason for the relatively low mAP-value in train detection is that the trains in the D7.4 dataset are visually very different from the ones present in the OSDaR training data.

| Model | Category | mAP | mAP_50 | mAP_75 | mAP_s | mAP_m | mAP_l |
|---|---|---|---|---|---|---|---|
| **Fine-Tuned** | Person | 0.356 | 0.596 | 0.373 | 0.000 | 0.174 | 0.497 |
| | Bicycle | 0.000 | 0.000 | 0.000 | N/A | N/A | 0.000 |
| | *Overall (bbox_mAP)* | *0.178* | *0.298* | *0.187* | *0.000* | *0.174* | *0.248* |
| **Pre-Trained** | Person | 0.345 | 0.617 | 0.344 | 0.000 | 0.146 | 0.483 |
| | Bicycle | 0.000 | 0.000 | 0.000 | N/A | N/A | 0.000 |
| | *Overall (bbox_mAP)* | *0.173* | *0.308* | *0.172* | *0.000* | *0.146* | *0.214* |

Table 4 Performance Comparison on D7.4 Data (RailGoerl Classes)

Interpretation: For the RailGoerl-specific classes, the fine-tuned model slightly outperforms the pre-trained model overall (mAP 0.178 vs 0.173). Neither model successfully detects bicycles. However, the information value here is low as in the test data, only 2 instances of bicycles were present in the evaluated dataset, making a reliable detection and statistical validation impossible for this class. In the class ‘person’ the fine-tuned model outperforms (0.356 vs 0.345) the pre-trained model (mAP).

#### 2.5.2.1 Detailed Precision-Recall Analysis (Baseline vs. Fine-Tuned)

To further investigate the performance characteristics of the models, an in-depth analysis of the Precision-Recall (PR) curves was conducted for the relevant object classes across both the OSDaR and RailGoerl datasets. These charts break down the models' accuracy and highlight the distribution of error types, particularly focusing on False Negatives (FN), Background errors (BG), and Localization errors (Loc).

##### *2.5.2.1.1 Baseline Performance: Pre-Trained Models*

Establishing the baseline performance of the pre-trained, generic object detection model helps to highlight the necessity and impact of the subsequent domain adaptation.

**Analysis of the 'Person' Class**

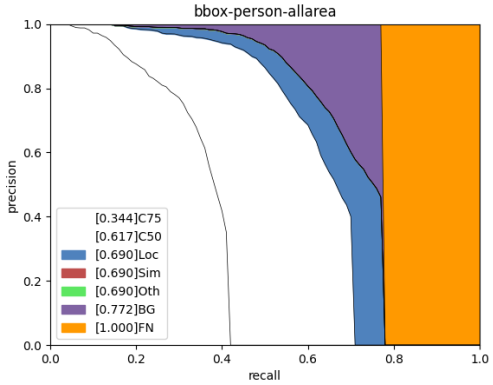


Figure 16 Precision-Recall curve for the 'person' class (pre-trained baseline)

The pre-trained model demonstrates a strong generalized baseline for the 'person' category, achieving a high mAP_50 of 0.617 and an mAP_75 of 0.344. The curve C75 maintains high precision (close to 1.0) up to a recall of approximately 0.40 before starting to decline. While False Negatives (FN = [1.000], orange area) ultimately limit the maximum achievable recall, there is also a noticeable fraction of Background (BG) errors, represented by the purple area. This indicates that the highly sensitive generic model occasionally registers false positive person detections in the background.

**Analysis of the 'Train' Class**

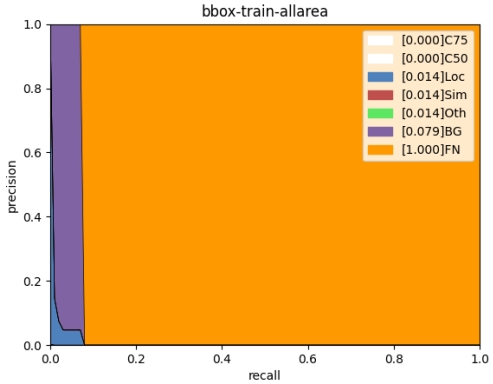


Figure 17 Precision-Recall curve for the 'train' class (pre-trained baseline)

The PR curve for the 'train' category starkly illustrates the limitations of generic models in this specific railway context. The model completely fails to detect the challenging train instances in the test set,

resulting in an mAP of 0.000. The chart is entirely dominated by the orange False Negative area. This confirms that the baseline model extracts zero actionable features for the trains in this dataset, which are primarily located far away and in dark environments (see Figure 17).

#### *2.5.2.1.2 Domain Adaptation: Fine-Tuned Models*

The analysis of the fine-tuned models illustrates how training on domain-specific data shifts the error distribution and introduces new detection capabilities.

**Analysis of the 'Person' Class (OSDaR Dataset)**

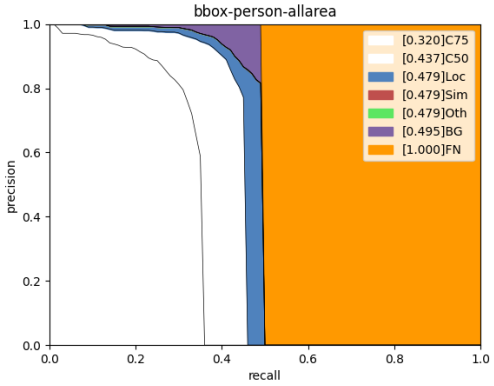


Figure 18 Precision-Recall curve for the 'person' class (OSDaR fine-tuned)

For the model fine-tuned on the OSDaR dataset the class "person" the performance profile of the 'person' class changes noticeably. It achieves an mAP_50 of 0.437 and an mAP_75 of 0.320. Compared to the pre-trained baseline, precision drops earlier as recall increases. The massive orange area indicates that False Negatives (FN) are the overwhelming error type, meaning the model misses a significant portion of 'person' instances in this highly challenging specific data slice. Background false positives (BG) are notably reduced compared to the pre-trained model. In practice, the trade-off between these two error-types could additionally be tuned in favor of fewer FN (and accepting higher BG) errors by adjusting the detection threshold appropriately – doing so allows the user to adjust the performance to their needs.

**Analysis of the 'Train' Class (OSDaR Dataset)**

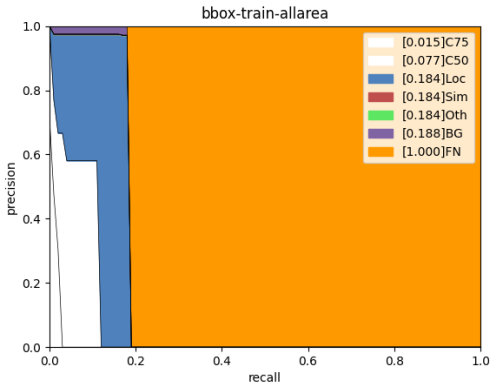


Figure 19 Precision-Recall curve for the 'train' class (OSDaR fine-tuned)

In direct contrast to the complete failure of the pre-trained baseline, the model fine-tuned on the OSDaR dataset successfully introduces the capability to detect trains under these severe conditions. Specifically, these severe conditions are characterized by very low illumination (night images), large distances to the objects, and visual characteristics that are significantly different compared to the training dataset (see train images in Fig. 21). While the overall mAP_50 is still low at 0.077 and the error distribution remains dominated by False Negatives, the presence of actual precision/recall measurements (the blue and purple areas on the left side of the chart) provides empirical proof that the domain-specific fine-tuning successfully enabled the model to learn and extract relevant rail-specific features.

**Analysis of the 'Person' Class (RailGoerl Dataset)**

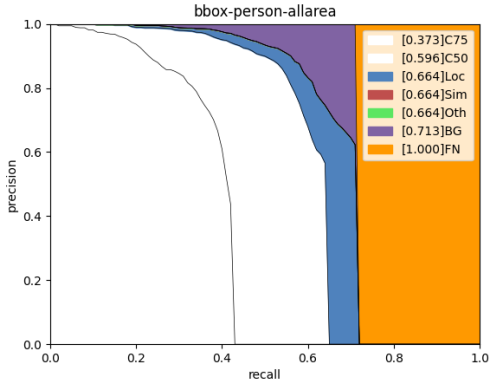


Figure 20 Precision-Recall curve for the 'person' class (RailGoerl fine-tuned)

For the model fine-tuned on the RailGoerl dataset, the PR curve of the category "person" shows a significantly stronger performance profile compared to the OSDaR subset. The model exhibits very high precision (near 1.0) up to a recall of approximately 0.5. At a confidence threshold of 0.75, it clearly outperforms the strong pre-trained baseline (mAP_75 of 0.373 vs. 0.344). Furthermore, the overall mAP also surpasses the baseline (0.178 vs. 0.173), demonstrating the clear advantage and success of the fine-tuning process. While False Negatives (orange area in the PR curve) restrict the maximum recall to roughly 0.7, there is a clearly visible portion of Background errors (BG, purple area). This indicates that in the RailGoerl domain, the model is confident but occasionally hallucinates person detections in empty background spaces.*Note on Bicycles:* Due to the extreme lack of 'bicycle' instances (only two in the evaluated dataset), generating meaningful statistical PR curves for this category was not possible for neither the pre-trained nor the fine-tuned model.

### 2.5.3 Qualitative Results and Visual Assessment

To complement the quantitative metrics, visual inspections were conducted comparing the ground truth annotations with the detector outputs.

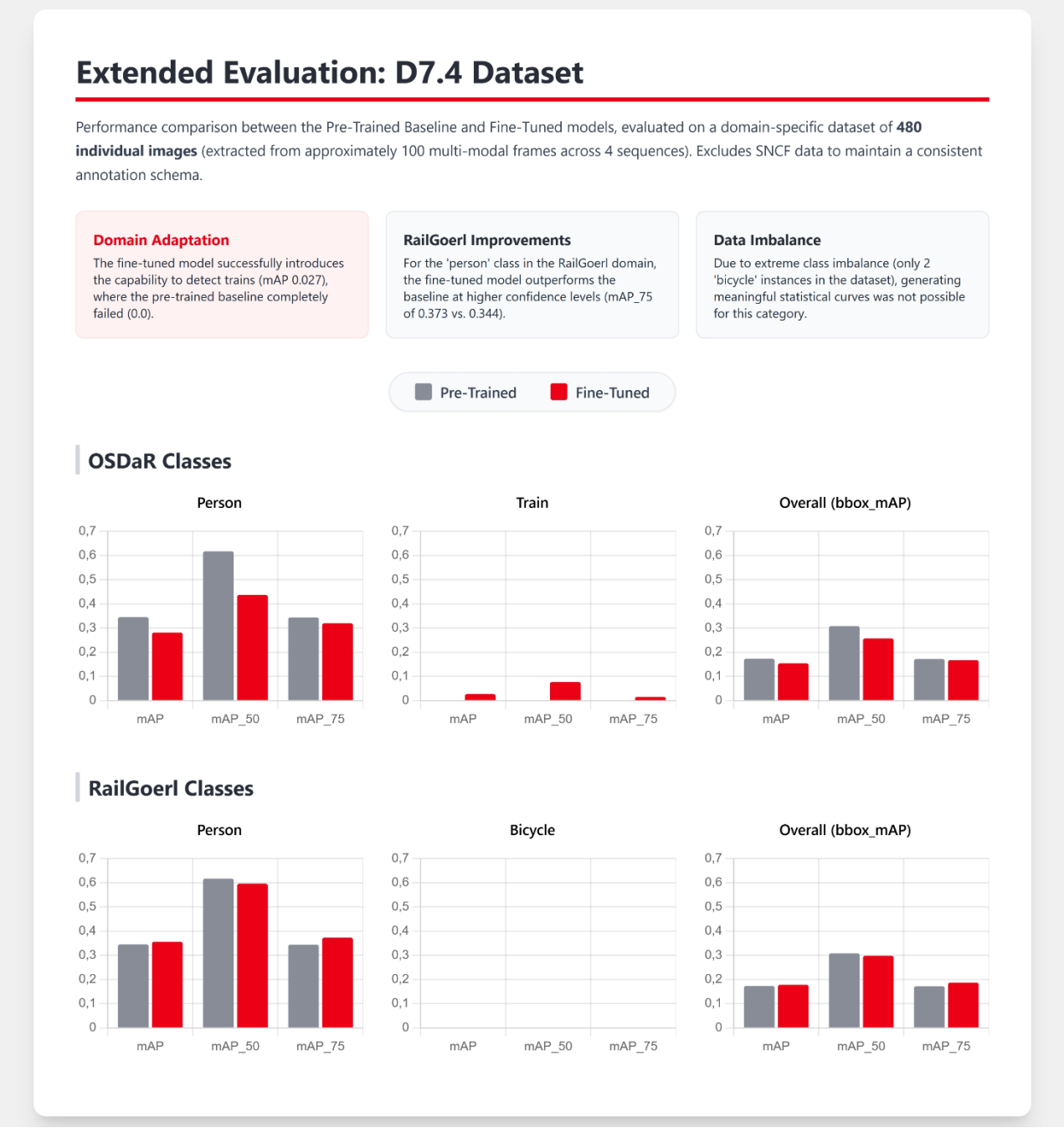


Figure 21 Comparison of mAP scores between Pre-Trained and Fine-Tuned models across different datasets

#### 2.5.3.1 Pre-Trained Model Performance

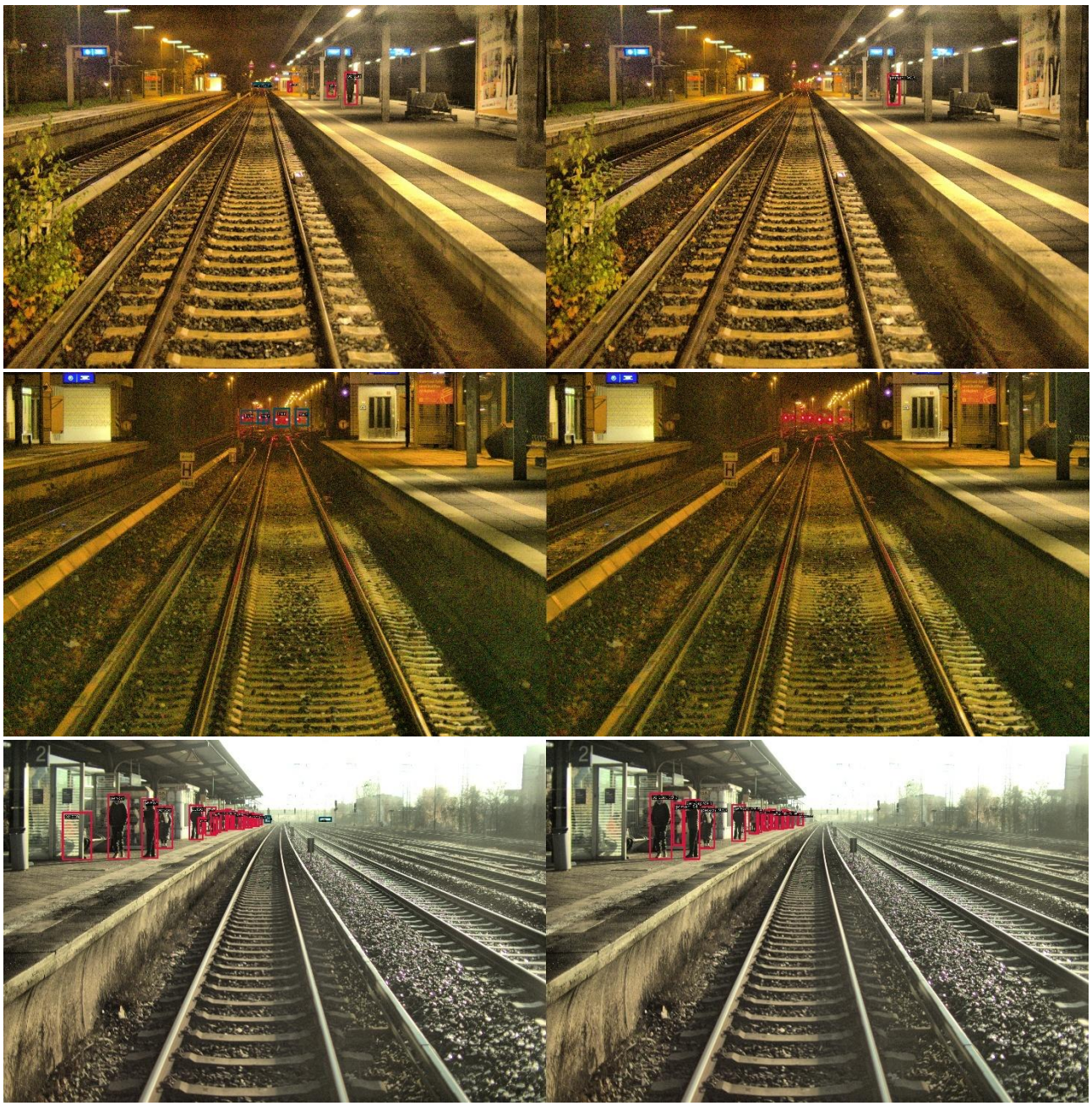

Figure 22 Qualitative evaluation of the pre-trained model. (Left: Ground Truth annotations; Right: Model predictions). The pre-trained model shows proficiency in detecting common objects like persons but misses domain-specific assets.

#### 2.5.3.2 Fine-Tuned Model Performance (OSDaR & RailGoerl)

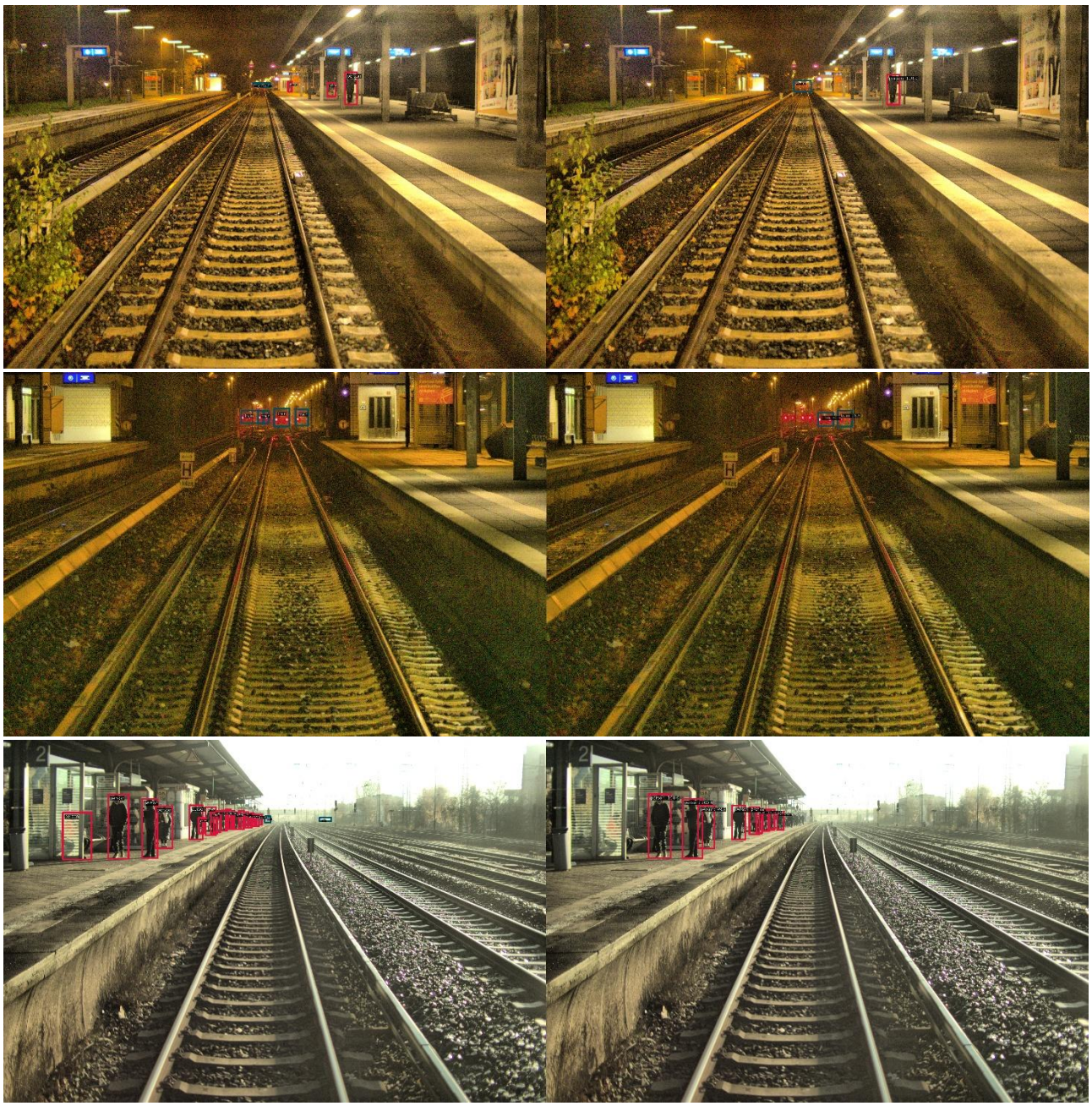

Figure 23 OSDaR fine-tuned Images: Left - Ground Truth, Right - Detector Result[2]

[2] The visual assessment demonstrates the fine-tuned model's capability to detect domain-specific objects (trains), successfully identifying 2 out of 4 instances in this scenario.

Figure 24 RailGoerl fine-tuned Images: Left - Ground Truth, Right - Detector Result[3]

Qualitative evaluation of the fine-tuned models on OSDaR and RailGoerl scenarios: The visual results confirm the quantitative findings, demonstrating improved recognition of railway-specific objects following the fine-tuning process within the Data Factory toolchain.

[3] Please note that the RailGoerl dataset inherently does not contain the object class 'train'. Consequently, the qualitative visual evaluation for this specific domain is limited to the available categories (primarily 'person'), which explains the absence of train scenarios in these examples

# 3 CONCLUSIONS

The completion of Deliverable 7.5 represents a significant progress for the WP7 Data Factory. The central objective, to implement a robust and functional ML training workflow within the Data Factory toolchain, has been successfully fulfilled. The rigorous evaluation of our methodology and the resulting performance gains provide a strong, empirical proof-of-concept for the entire data-to-model pipeline. This work proves the Data Factory is not only a theoretical framework but a demonstrable, operational system, significantly advancing the Technology Readiness Level (TRL) for data-driven perception and automation solutions. This transition from theory to practice is key as it proves that the conceptual approaches of D7.1 and the data preparation of D7.4 are indeed capable of generating measurable and valuable results.

This deliverable contributes new knowledge to the railway sector by providing a clear, quantifiable demonstration of the value of domain-specific data and a standardized toolchain. The model refinement process resulted in a 181% increase in bbox_mAP_75 over all classes when evaluated on the OSDaR23 dataset. On RailGoerl, the refined model achieved an impressive 180% increase in mAP_75 performance over all classes. Especially the commonly used, pre-trained bicycle detector was shown to have severe weaknesses in rail settings that could be overcome by refining it on rail-specific data.

These are not merely incremental improvements; they are irrefutable evidence that generic, off-the-shelf models are insufficient for the stringent safety demands of autonomous rail and that the Data Factory's strategic approach is the correct path forward. The high increase in the bbox_mAP_75 metric is particularly significant as it quantifies the precision of object localization—a critical factor for ensuring safety and preventing potentially catastrophic failures caused by object localization errors. The results reinforce the central thesis that a collaborative, standard-based data strategy is the only feasible solution to create the necessary data foundation for trustworthy AI systems.

Furthermore, the integration of D7.4 data into the evaluation loop successfully demonstrates the core concept of the Data Factory across three crucial dimensions:

1. Domain Adaptation: Fine-tuning is essential for domain-specific classes (e.g., 'train'), which standard pre-trained models fail to recognize.

2. Data Imbalance Mitigation: The inability to detect bicycles highlights the ongoing need for targeted data collection campaigns (or synthetic data generation via D7.3) for edge cases and underrepresented classes.

3. Standardization Imperative: The exclusion of SNCF data due to differing annotation formats underscores the necessity of the unified formatting standards that are being established for the final Open Dataset (D7.6).

While the primary objectives were met, this work also brought to light several key challenges and limitations. We confirmed the inherent data scarcity for rare and hazardous Non-Regular Situations (NRS), which remains a significant hurdle for achieving GoA4 readiness. Although training on publicly available datasets was successful, it underscored the need for a much larger, more diverse corpus of data to ensure robust performance across all environmental conditions. For instance, there is a clear lack of data for more extreme scenarios such as heavy rain, dense fog, or during direct sunlight at sunrise or sunset. Furthermore, our results showed that performance for small objects

(bbox_mAP_s) remains a major open problem, highlighting a fundamental technical challenge, as such objects offer very few pixels for neural networks to extract features from. An important lesson learned is that while a technical solution can be validated, the true bottleneck for real-world deployment lies in the time-intensive processes of data collection, annotation, and quality control, which pose significant scalability challenges.

Building on these achievements and insights, several open points and proposals for future activities within Europe's Rail are recommended. The most critical next step is to bridge the gap to certification and homologation. This requires moving beyond a "proof-of-concept" and establishing a standardized framework for the rigorous validation of AI models in safety-critical applications. Future work should focus on integrating a much larger volume of realistic data from rail contexts to train models for critical edge cases and to ensure they are robust to a wide range of dynamic scenarios. It is also important to extend beyond static object detection to include the ability to track objects over time and anticipate their motion intentions. All these developments should be in line with a clear requirements-based architecture to integrate all stakeholder needs appropriately.

By providing a clear technical roadmap and demonstrating the tangible benefits of a collaborative data ecosystem, this deliverable leaves the reader with an impression of the project’s impact, validating the strategic vision of a Pan-European Data Factory as the essential foundation for a safer, more automated European rail system.

# REFERENCES

Eichenbaum, J. e. (2024). *Towards Conceptually Elevating Modern Concepts of Operational Design Domains and Implications for Operating in Unstructured Environments* (Vols. Systems, Software and Services Process Improvement). Cham: Springer Nature Switzerland.

Girshick, R. (2015). Fast R-CNN. *arXiv*, https://arxiv.org/abs/1504.08083.

J. Erz, B. S. (2022). Towards an Ontology That Reconciles the Operational Design Domain, Scenario-based Testing, and Automated Vehicle Architectures. *2022 IEEE International Systems Conference (SysCon)* (pp. 1-8). Montreal: IEEE.

Rustam Tagiew, I. W. (2025). RailGoerl24: Görlitz Rail Test Center CV Dataset 2024. *https://arxiv.org/*.

Sebastian Lapuschkin, S. W.-R. (2019). Unmasking Clever Hans predictors and assessing what machines really learn. *Nature Communications*, https://doi.org/10.1038/s41467-019-08987-4.

Shaoqing Ren, K. H. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *arXiv*, https://arxiv.org/abs/1506.01497.

Tilly, R., Neumaier, P., Schwalbe, K., Klasek, P., Tagiew, R., Denzler, P., . . . Köppel, M. (2023). *Open Sensor Data for Rail 2023 [Data set]*. Retrieved from https://doi.org/10.57806/9mv146r0

Tsung-Yi Lin, M. M. (2014). Microsoft COCO: Common Objects in Context. *arXiv*, https://arxiv.org/abs/1405.0312.