data

Data Descriptor

Dataset for Device-Free Wireless Sensing of Crowd Size in Public Transportation Environments

Robin Janssens [1,2,*], Rafael Berkvens [1] and Ben Bellekens [2]

[1] IDLab, Faculty of Applied Engineering, IMEC, University of Antwerp, 2000 Antwerp, Belgium; rafael.berkvens@uantwerpen.be
[2] CrowdScan BV, 2640 Mortsel, Belgium; ben.bellekens@crowdscan.be
[*] Correspondence: robin.janssens@uantwerpen.be

Abstract

Congested platforms in public transportation systems can jeopardize the safety and comfort of passengers. Real-time crowd size estimation using Device-Free Wireless Sensing (DFWS) can offer a privacy-preserving solution for monitoring and preventing overcrowding. However, no public dataset exists on DFWS in public transportation environments. In this work, we introduce a new dataset comprising two different public transportation environments, which contains data on the presence of rail vehicles at the platform, as well as manual people counts at regular intervals. By providing this dataset, we aim to offer a foundation for other DFWS researchers to explore novel algorithms and methods in public transportation environments.

Dataset: https://doi.org/10.5281/zenodo.18175571

Dataset License: CC-BY

Keywords: people counting; crowd counting; wireless sensing; RSSI; WSN; smart city; public transportation

1. Introduction

The collection and analysis of historical and real-time data has become the cornerstone of the smart city [1]. A common information feature that concerns many applications is the number of people in a specified location, since large crowds can cause congestion or even safety risks. In addition, from a service perspective, we can optimize the allocation of staff and resources based on forecasted and real-time occupation.

In recent years, the Mobility as a Service (MaaS) paradigm has received increased attention as a key strategy for achieving sustainable urban mobility. In [2], Jittrapirom et al. present an overview of existing MaaS implementations and identify three core characteristics: demand modeling, supply-side analysis, and business model.

In such a MaaS framework, real-time crowd monitoring can be used in both the demand modeling and supply-side analysis. The demand model can use real-time crowd data for dynamically adjusting the model based on real-world demand or unexpected conditions. The supply side can leverage the real-time crowd data for detecting congestion and adjust supply dynamically, improving efficiency.

In [3], the author identifies the supply-side challenges to implementing Sustainable MaaS (S-MaaS). The implementation is divided into four categories, one of them being the

Academic Editors: Diego Gabriel Rossit and Pedro Moreno-Bernal

Received: 27 November 2025
Revised: 8 January 2026
Accepted: 10 January 2026
Published: 14 January 2026

immaterial components. This includes real-time data-streams that are essential for effective decision-making and detecting unexpected conditions. Information on real-time system status, including platform occupation, can be used to monitor supply–demand interactions to optimize system efficiency [4].

In this work, we present a dataset for crowd size estimation on public transportation platforms in two light rail environments.

The dataset shared in this work is collected using a Device-Free Wireless Sensing (DFWS) technique. This technique is based on a Wireless Sensor Network (WSN) of battery-powered sensor nodes placed inside and around the environment. Each sensor node in the network periodically transmits a message that is received by the other sensor nodes in the network. The Received Signal Strength Indicator (RSSI) values corresponding to the received messages are stored until the next transmission to be included as the data payload. An internet-connected gateway will receive all the messages in the network as well and will forward the data included in the payload to a back-end server for processing [5]. A high-level representation of the three operations executed by the measurement cycle is depicted in Figure 1.

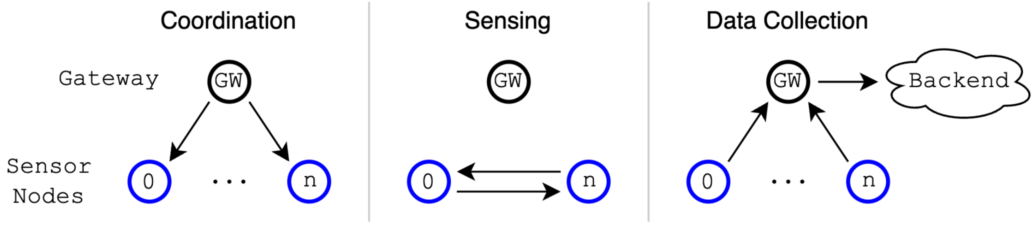

Figure 1. This diagram shows the three communication components of the Device-Free Wireless Sensing technique.

The measured RSSI values of the various links in the network contain valuable information about the state of the environment. When people stand between the sensor nodes, they obstruct the Line of Sight (LoS) between these sensor nodes, affecting the RSSI values. As the human body has a large water content, it will absorb part of the radio waves that pass through it, reducing the signal strength of the remaining signal, which will be received on the other side of the environment by the receiving sensor node.

As we are dealing with public transportation environments, people are not the only entities obstructing the radio links; rail vehicles and road vehicles inside or near the environment can also have a significant impact on the measured signal strength. As these objects are made out of metal, they not only heavily reduce the RSSI value of links passing through the object, but also cause significant constructive and destructive multipath interference for links to and from sensor nodes near the object.

By gathering data from multiple links and combining it with ground truth data, we can leverage machine learning techniques to filter out unwanted effects and estimate the number of people in the environment.

This DFWS technique has its origin in the field of Device-Free Localization (DFL) using Radio Tomographic Imaging (RTI), where the attenuation of the links in the WSN are used to determine the location of people or objects [6].

Coluccia et al. [7] introduced a novel framework for RTI that operates effectively with extremely sparse and location-uncertain transmitters. They propose both an optimal and a suboptimal algorithm, achieving near-optimal performance with significantly reduced computational cost.

In [8], the use of mmWave radio is proposed for DFL. This method uses multipath reflections as virtual anchor nodes to reduce hardware requirements and improve accuracy. The technique is based on compressed sensing, clustering, and ray-tracing-assisted processing.

As the number of people in the environment increases, it can become increasingly difficult to accurately determine their individual locations, but estimating higher-level metrics such as the approximate number of people present in the environment is still possible, as demonstrated by Denis et al. in multiple large-scale festival environments [9]. This formed the basis for DFWS crowd counting applications.

Different techniques and methods exist to count or estimate the number of people present in an environment. These different techniques have their own strengths and weaknesses when it comes to the accuracy and modalities offered.

The most established technique for people counting and density estimation uses image data collected by a camera system. Image data can be used in two different ways to count people. The first way is a direct approach using feature extraction on an image to estimate the number of people in the image. In [10], a Convolutional Neural Network (CNN)-based approach is used to count people standing on metro platforms.

The second way camera data can be used is to count people walking in and out of a specific area. This technique can be referred to as line counting or flow counting, as it counts the flow of people crossing a virtually defined line on the image. In [11], security cameras are used in combination with a CNN and a tracking algorithm to demonstrate the ability to count people moving between the platform and a rail vehicle.

An alternative to camera-based counting can be the use of a radar-based sensor. Radar-based sensors can be used similarly to camera-based counters to count people within the field of view (FoV) of the radar [12] or to count people walking in and out of the environment when placed above the entrance [13].

Both camera- and radar-based counting techniques suffer from limitations that affect their usability in real-world scenarios. For example, both techniques have a limited FoV and range, which limits the area that the counting system can cover. Another big factor is occlusions. Examples of such occlusions are people holding umbrellas or even the dense crowd itself. Such occlusions can negatively impact the system's performance. DFWS offers a crowd size estimation solution suitable for wide-area scenarios with limited reliance on visual LoS.

The use of radio waves in DFWS makes it nearly impossible to identify individuals. The information captured by the attenuation of the radio waves is low enough in spatial and temporal resolution that identifiable information cannot be recovered from the data. Even broader characteristics such as age, sex, or length cannot be recovered. This makes DFWS a privacy-by-design technology. This contrasts with cameras [14] and, to a limited extent, radar systems [15], which are capable of collecting personal characteristics and potentially identifying individuals.

The number of open datasets on DFWS systems for people counting is very limited. One of the few datasets in existence was published by Kaya et al. in 2020 [16]. This dataset contains data on three environments at music festivals. This dataset offers a solid baseline for DFWS applications. However, the applicability of this dataset is limited to environments with only people. Public transport environments have much more dynamic behavior, with rail vehicles being introduced into the environment and faster changes in crowd densities.

In this work, we introduce a DFWS dataset consisting of two environments. The first environment is an underground metro station. We refer to this environment as the indoor environment. The second environment is an outdoor above-ground platform. We refer to this environment as the outdoor environment. For both environments, ground truth data were collected on three different days for periods of approximately one hour. This data

https://doi.org/10.3390/data11010021

includes the presence of rail vehicles and the number of people on the platform counted at regular intervals.

The data of the indoor environment have been used in a prior publication [5]. In this paper, a WSN was used to estimate the number of people standing on the subway platform with and without a rail vehicle present. To achieve this, the WSN was used to perform a classification on whether a rail vehicle is present in order to switch between separately trained linear regression models.

To the best of our knowledge, this is the first DFWS dataset dedicated to crowd size estimation in public transportation environments. In this data descriptor paper, we provide two datasets collected in two distinct public transportation environments. By making these datasets publicly available, we aim to provide researchers in the field of DFWS and crowd counting the opportunity to develop and test new data processing methodologies on real-world data.

This paper is structured as follows: In Section 2, we discuss other relevant works within the field of DFWS. In Section 3, we discuss the content of the dataset and how to interpret the different files. Section 4 explains how the data was collected. In Section 5, we demonstrate some example use cases for the dataset. Finally, in Section 6, we conclude this paper.

2. Related Work

In this section, we review other works related to DFWS crowd monitoring applications.

In [17], Yuan et al. present a crowd density estimation method using RSSI in WSNs. The approach clusters density levels via k-means and refines results through spatial–temporal calibration to filter out noise and deviations. The authors validated the proposed solution in a 7.2 m by 7.2 m area with 16 nodes by classifying between three density levels in four subareas; the method achieved coincidence rates of 89–96%.

The paper [18] introduces a Wi-Fi-based crowd counting system using beacon message samples and compares RSSI- and Channel State Information (CSI)-based descriptors in two indoor environments. Using a single transmitter–receiver pair and up to five people, results show that CSI outperforms RSSI in larger, multipath-rich rooms, while RSSI remains effective in smaller spaces.

In [19], the authors propose a Wi-Fi CSI-based system for simultaneous crowd counting and localization using low-cost ESP32 nodes. By extracting dynamic and static CSI features, the system applies regression for counting and classification for localization. The experiment includes four transmitter–receiver pairs in two indoor environments, a small room and a medium-sized room, with up to 5 and 10 people. The experiment achieved median absolute errors of 0.35 and 0.41 for crowd size estimation and localization accuracies of 91.4% and 98.1%, respectively.

In [20], De Sanctis et al. present a device-free Wi-Fi sensing approach for queue counting using RSSI measurements. The system adopts a modular design, splitting the queue into 1 m by 1 m modules, each estimating 0–4 persons via a naïve Bayes classifier. Two feature extraction methods are evaluated: histogram-based Probability Density Function (PDF) and statistical measures of RSSI. The modular approach simplifies training and scales to long queues. Experiments report up to 98% accuracy for a single module and 93% for six modules (24 people).

The work [21] proposes a DFWS method for estimating the number of people through walls using only Wi-Fi RSSI from a single transmitter–receiver pair placed outside the area. The approach exploits inter-event times of LoS blockage, modeled as a superposition of renewal-type processes, and applies a maximum likelihood estimator. Experiments across five areas with different wall materials and up to 20 people achieve a smaller than or equal to two-person error 100% of the time.

3. Data Description

The dataset has been split into two folders, each containing data for its respective environment. Figures 2 and 3 show the files contained in the dataset for the indoor and outdoor environments, respectively.

```
data_platform_indoor
 |__wsn_indoor_environment.json
 |__rssi_data
 |   |__rssi_platform_indoor_2023-05-17.csv
 |   |__rssi_platform_indoor_2023-05-18.csv
 |   |__rssi_platform_indoor_2023-05-19.csv
 |   |__rssi_platform_indoor_2023-07-17.csv
 |__training_data
     |__training_platform_indoor_2023-05-17.csv
     |__training_platform_indoor_2023-05-19.csv
     |__training_platform_indoor_2023-07-17.csv
```

Figure 2. Folder structure of the indoor dataset.

```
data_platform_outdoor
 |__wsn_outdoor_environment.geojson
 |__rssi_data
 |   |__rssi_platform_outdoor_2024-06-11.csv
 |   |__rssi_platform_outdoor_2024-06-12.csv
 |   |__rssi_platform_outdoor_2024-07-02.csv
 |__training_data
     |__training_platform_outdoor_2024-06-11.csv
     |__training_platform_outdoor_2024-06-12.csv
     |__training_platform_outdoor_2024-07-02.csv
```

Figure 3. Folder structure of the outdoor dataset.

3.1. RSSI Measurement Data

Table 1 shows the first three lines of `rssi_platform_indoor_2023-05-17.csv`. The RSSI data files contain five fields. Figure 4 shows a diagram of the first two messages in a cycle, which correspond to the data shown in Table 1 and are preceded by a "start message" used to synchronize the nodes.

Table 1. Sample of RSSI data file.

Timestamp	Node_ID	Cycle_ID	Rssi_GW	Rssi_VALUES
2023-05-17T00:00:01.726565+0200	0	4	69	"[0, 70, 70, 65, 77, 82, 0, ...]"
2023-05-17T00:00:01.917667+0200	3	4	58	"[64, 62, 42, 0, 62, 72, 0, ...]"
2023-05-17T00:00:01.816649+0200	1	4	66	"[70, 0, 78, 61, 71, 76, 0, ...]"

The `timestamp` is assigned when the message is parsed on the server back-end. The gateway immediately forwards each message it receives from a node in the network. The `timestamp` is stored in localized ISO 8601 format [22].

`node_id` contains the zero-indexed node identification number. This node ID is assigned to the sensor node during provisioning of the sensor network before or during installation.

https://doi.org/10.3390/data11010021

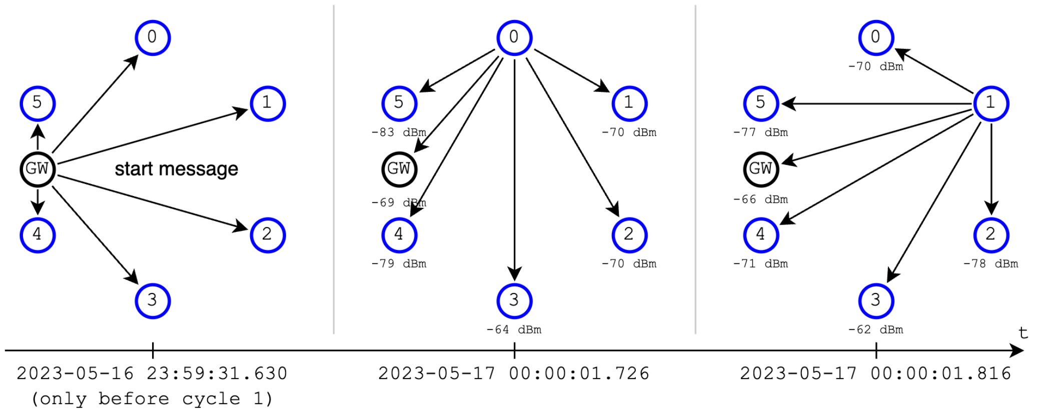

Figure 4. This diagram shows the "start message" used for coordination of the network and the first two messages of the dataset, sent by node 0 and 1.

`cycle_id` is the one-indexed counter for each measurement cycle. We perform ten measurement cycles between synchronizing the network. This value is used during processing to group messages from the same cycle into a two-dimensional matrix.

`rssi_gw` contains the receiving signal strength of the message by the gateway in absolute dBm (i.e., 42 ⇒ −42 dBm).

`rssi_values` contains the buffered received signal strength values, which are the contents of the message payload. The buffer has a fixed length of 60 values, allowing for 60 nodes per network. The position of the value in the list corresponds to the node ID of the transmitting node. E.g., the first field corresponds to the signal strength corresponding to messages originating from node 0. Similarly to the `rssi_gw` field, the values are RSSI measurements in absolute dBm. A zero value means that no value was received. The field corresponding to itself will always be zero, as the node does not receive its own transmissions. In Python, the `rssi_value` list can be loaded from the stored string using the `eval()` function.

3.2. Ground Truth Data

Table 2 shows a sample from `training_platform_indoor_2023-05-17.csv`. The ground truth data files contain two fields.

Table 2. Sample of ground truth data file. Side notes are added in grey writing next to the table.

Timestamp	Value	
2023-05-17T17:39:57.942000+0200	24	← People count without tram present
2023-05-17T17:40:10.060000+0200	−1	← Tram arrival
2023-05-17T17:40:22.248000+0200	27	← People count with tram present
2023-05-17T17:41:05.795000+0200	−2	← Tram departure

The first column of the file is the `timestamp` represented in localized ISO 8601 format [22], identical to the raw data files. This timestamp is generated when manual data is entered via a smartphone app.

The `value` field can hold two types of data: people count values and information on the arrivals and departures of rail vehicles. Values greater than or equal to zero represent people count values. "−1" represents the arrival of a rail vehicle to the platform. "−2" represents the departure of a rail vehicle. The arrival and departure values can be used to train rail vehicle detection models, as well as to separate the ground truth people count values into two training sets based on the presence of a rail vehicle [5].

Table 3 shows an overview of the time of day and duration of the available ground truth data. This table shows that the various data collection sessions have similar durations (approximately one hour). This can be ideal for treating the collection periods as single units of data in a cross-validation setting, as some level of autocorrelation may exist between consecutive samples within the same window due to the nature of the data. Therefore, performing training and evaluation on different days is recommended to reduce potential bias.

Table 3. Overview of ground truth collection windows.

Indoor Environment			
Date	**First Timestamp**	**Last Timestamp**	**Total Duration**
17 May 2023	17:38:47.595+0200	18:45:41.675+0200	1 h 7 min
19 May 2023	17:35:11.201+0200	18:28:46.231+0200	54 min
17 July 2023	15:08:29.560+0200	16:08:04.941+0200	1 h 0 min
Outdoor Environment			
Date	**First Timestamp**	**Last Timestamp**	**Total Duration**
11 June 2024	14:30:15.060+0200	15:16:14.055+0200	46 min
12 June 2024	18:29:07.557+0200	19:25:56.859+0200	57 min
2 July 2024	10:38:47.501+0200	11:45:56.432+0200	1 h 7 min

3.3. Node Locations

Both environments contain a JSON file that describes the node locations and gateway location in the environment.

The outdoor environment contains a GeoJSON file containing geographic coordinates and the mounting height of the device in meters, relative to the platform ground level.

The indoor environment contains a JSON file whose structure is closely related to the GeoJSON format but contains XYZ coordinates in meters, relative to the northwest corner of the platform. These coordinates are approximated locations based on a set of laser rangefinder measurements.

4. Materials and Methods

In this section, we discuss how this dataset was gathered. We discuss the hardware used, the methodology for ground truth collection, and the experimental setup in both environments.

4.1. Wireless Sensor Network

The WSN is constructed of two main components: a gateway and a set of wireless sensor nodes. All communication between the nodes and between the nodes and the gateway uses the DASH7 Alliance Protocol [23,24] in the 868 MHz Short-Range Devices (SRD) band. The gateway we used is a WizziLab (Montrouge, France) WizziGate Pro. The sensor nodes are WizziLab Wolt-XL nodes containing custom firmware. Both networks communicated on a single Hi-Rate channel with a bandwidth of 150 kHz and a baud rate of 166.667 kbit/s. The nodes transmit using an Effective Radiated Power (ERP) of 14 dBm.

The gateway has two main tasks in the network: The first is sending a coordination "start message". The start message has the purpose of performing the ad hoc synchronization in the DASH7 network. The nodes will listen in a low-power background mode for the start message. As all nodes receive this message and know their own node ID, they can calculate in which time slots they have to listen for messages of other nodes and in which time slot they have to transmit. The cycle is visually shown in Figure 5.

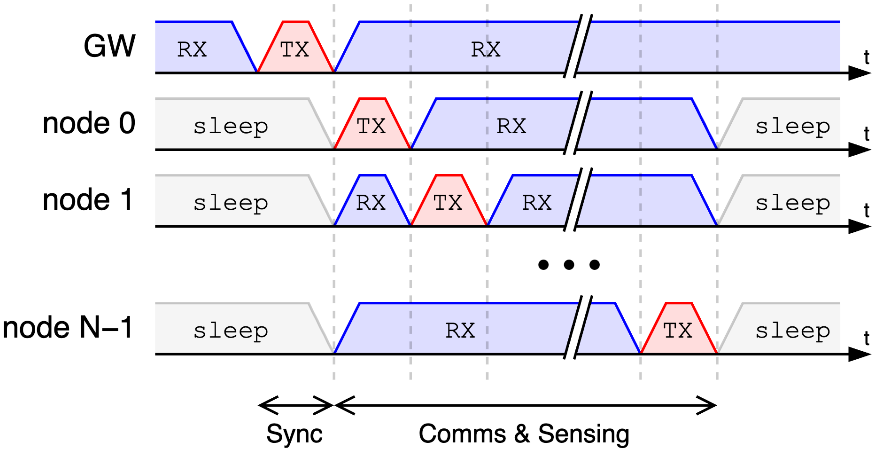

Figure 5. This diagram shows the transmit and receive time slots that make up a single first cycle.

The start message also clears the nodes' RSSI buffer storing the received signal strength values. This results in only half of an RSSI matrix for the first cycle, as only one direction of the links has been measured when the message is transmitted from the node to the gateway. The other half of the matrix is included in the subsequent cycle.

Resetting of the RSSI data buffer is performed for practical reasons. Some applications do not run the system continuously. Since the nodes lack an internal real-time clock and cannot determine the time elapsed since the previous cycle, we clear the buffer to prevent outdated data from remaining.

For power efficiency, the network will only perform a resynchronization every ten cycles. This allows the nodes to save power in between cycles, as they do not have to listen for the gateway in background scan state [24].

The second task the gateway performs is to receive all the messages transmitted by the sensor nodes and forward them to the back-end server via an LTE cellular backhaul. The payload of the message contains the last state of the RSSI receive buffer. This means that values for a lower node ID are from the current cycle while the values for a higher node ID are from the previous cycle, as this node has not received a message from the corresponding node yet during the current cycle.

4.2. Manual Ground Truth Collection

Ground truth data for both the number of people present and for rail vehicles arriving and departing the station were collected using a smartphone application.

To record the number of people present in the environment, the person collecting training data on site periodically performed manual counting of people within the defined area of the environment. The application allows the input of an integer value representing the number of people present. By pressing the save button, the inserted value, together with the current timestamp, will be stored.

For recording rail vehicle arrivals and departures, the application interface has two buttons, "in" and "out". When pressing one of these buttons, the value −1 for "in" or −2 for "out" is recorded together with the current timestamp. We use negative values −1 and −2 for arrivals and departures of rail vehicles, as numerical values greater than or equal to 0 represent the people count values. A negative people count would be meaningless and is not possible. We use negative integer values for other events, such as rail vehicle movements, to maintain the signed integer type of the field, allowing for ease of parsing the data.

Since the ground truth data are collected independently from the measurement samples, the timestamps in both sets will not align with each other. This is a problem we have to address when assigning labels to the measurement data. Due to the nature of the data, we perform this alignment differently for the people counts than for the vehicle ground truth data.

Since each collected people count represents a single point in time, we suggest combining it with the nearest measurement cycle, assigning the people count value as the label to this measurement cycle. For safety, a maximum time tolerance can be assigned to prevent combining values over large time gaps (e.g., maximum 5 min), after which the manual count value might be considered invalid. This process is visually depicted in Figure 6.

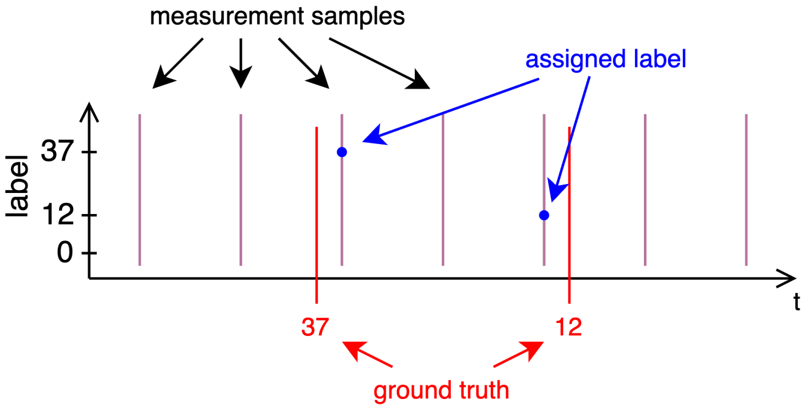

Figure 6. This diagram shows the labeling of samples based on the people ground truth data.

In contrast, vehicle ground truth data can be considered as a time span given by a start and end time. We want to label measurement samples with whether or not a rail vehicle was present for use later to train and evaluate a binary classification model. Thus, we can label more than only the nearest sample. We label all measurement samples between the first and last ground truth value. After a vehicle enters the station, we label all subsequent measurements as “present” or 1 until the vehicle leaves the station. Similarly, when a vehicle leaves the station, we label all subsequent measurements as “not present” or 0 until another vehicle enters the station. This process is visually depicted in Figure 7.

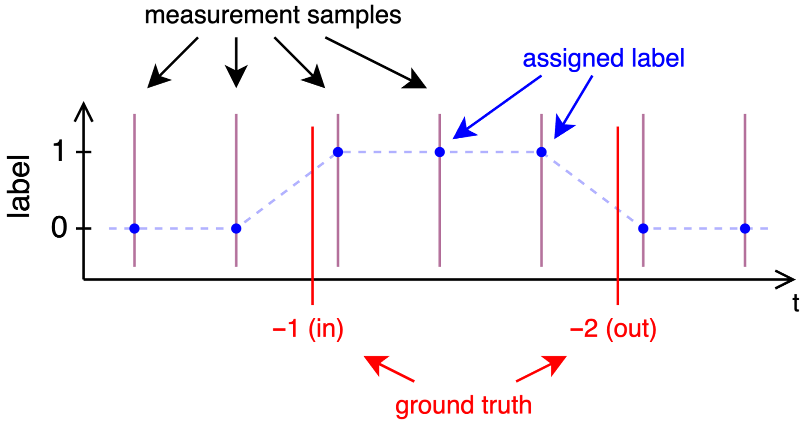

Figure 7. This diagram shows the labeling of samples based on the vehicle ground truth data.

https://doi.org/10.3390/data11010021

4.3. Experimental Setup

This work presents data from two distinct public transportation environments, an indoor and an outdoor platform environment. The first environment, the indoor environment, is part of the “Groenplaats” subway station located in the city of Antwerp, Belgium. The environment is an underground subway platform. This environment contains a single platform and rail track. The second environment, the outdoor environment, is the platform of the “Hospital Sant Joan Despí | TV3” tram stop located in Barcelona, Spain. This environment features a central platform enclosed by two rail tracks offering a bidirectional tram service.

These two environments feature distinctive properties: “Groenplaats”, being underground, acts as an indoor environment, experiencing more multipath effects on the Radio Frequency (RF) links than outdoor environments. Outdoor environments such as “Hospital Sant Joan Despí” are susceptible to weather conditions and other external effects. One such external effect is dynamic reflections occurring outside the observed environment, for example, from cars passing on the adjacent road. In the remainder of this paper, we will refer to these environments as the ″indoor″ and ″outdoor″ environments.

Figure 8 shows a top-view and side-view representation of the indoor environment. Figure 9 shows a picture of the indoor environment. Similarly, Figure 10 shows a top-view and side-view representation of the outdoor environment. Figure 11 shows a picture of the outdoor environment. The top view shows the node locations and their node ID. The side view shows the mounting height. The different (colored) shapes group the nodes for later reference. The hip height sensors are mounted at a height between 1 m and 1.2 m above the platform level. The ceiling nodes are mounted at a height of 2.7 m in the indoor environment and 3 m in the outdoor environment. More detailed node locations are included in the dataset, as discussed in Section 3.3.

The node groups can be combined to optimize for different use cases, such as people counting or rail vehicle detection. There is no physical difference between the node groups or how the data is handled. The gateway location is depicted as the crossed square.

Looking at Figures 8 and 10, it becomes clear that the outdoor environment contains significantly fewer sensor nodes than the indoor environment due to the limitations of the outdoor environment in terms of potential mounting options. For the indoor environment, we opted for maximizing the number of nodes beyond what is common for similar setups to allow analyzing the impact of fewer nodes and node locations in post-analysis.

A note to be made for the indoor environment is that nodes 0 and 2 are missing for the last day in the dataset (17 July 2023) due to physical removal of these nodes by technical staff after detaching from the wall.

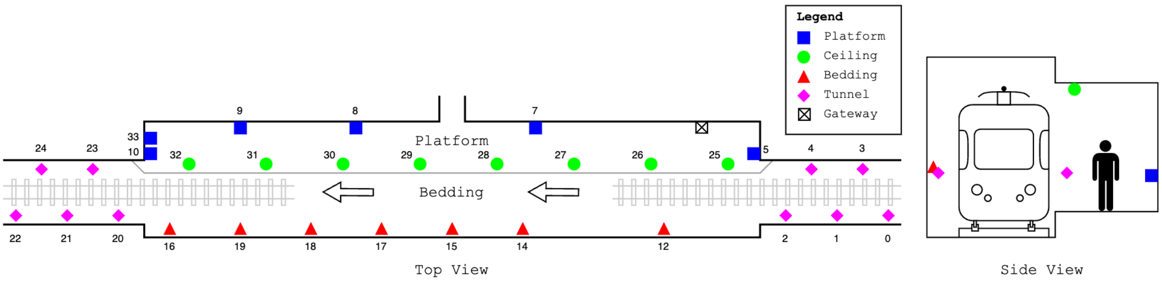

Figure 8. This figure shows a representation of the indoor experiment environment, including the approximate node locations with their corresponding node ID, as well as the gateway’s location.

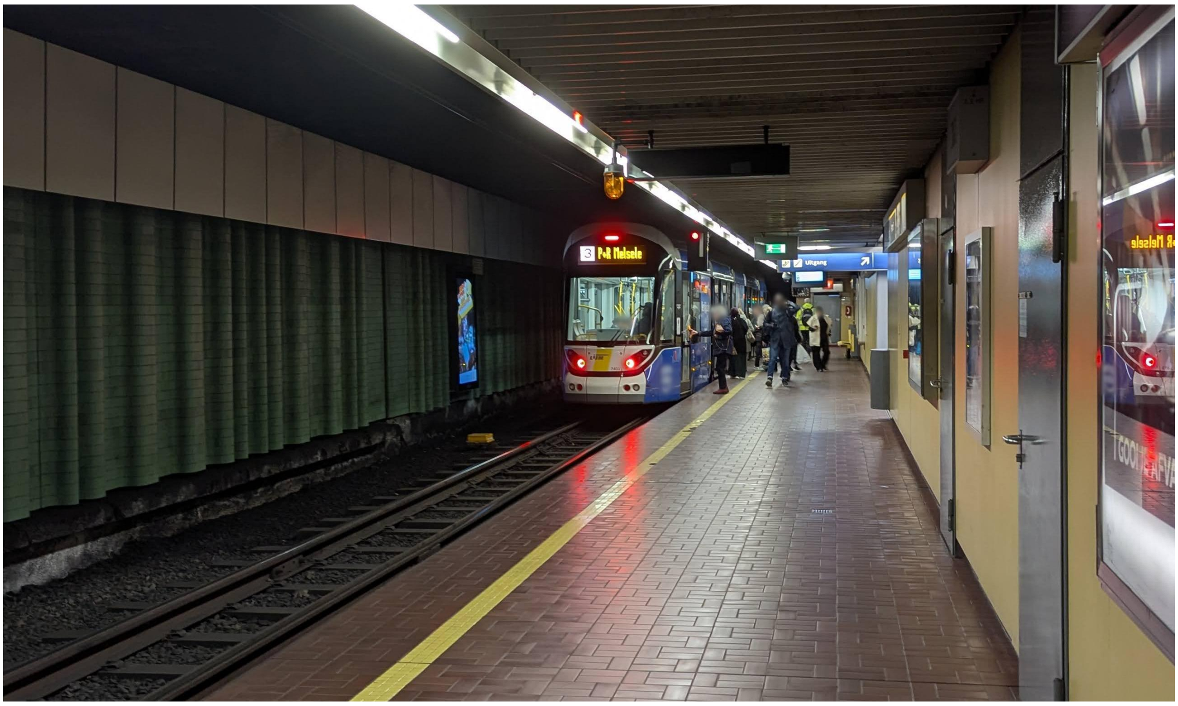

Figure 9. This picture shows the indoor experiment environment.

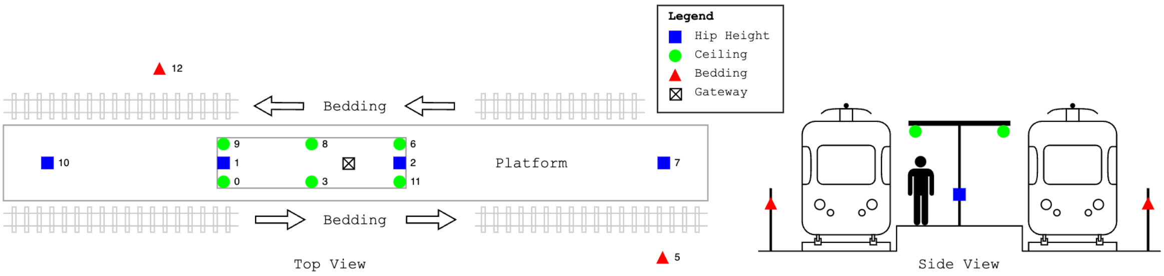

Figure 10. This figure shows a representation of the outdoor experiment environment, including the approximate node locations with their corresponding node ID, as well as the gateway's location.

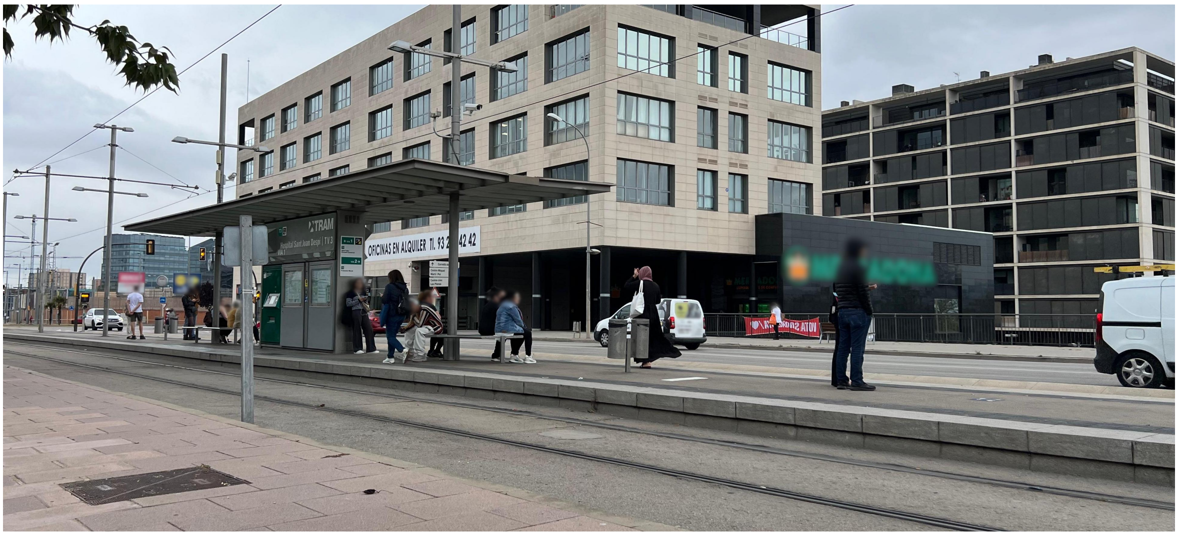

Figure 11. This picture shows the outdoor experiment environment.

4.4. Data Processing

For each measurement cycle, we can compile an RSSI matrix of size $N \times N$, with N being the number of nodes in the network. For simplicity, we use a fixed value $N = 60$

https://doi.org/10.3390/data11010021

during processing to allow for future expansion without changing the index mapping; additionally, this also aligns with the size of the "rssi_values" field, preventing the need to map when certain node IDs are not used. Figure 12 shows an example of an RSSI matrix for the first cycle of the indoor environment on 17 May 2023. A row in the table corresponds to a row in the dataset. "RX Node" corresponds to the "node_id" field; the values correspond to the "rssi_values" in order.

TX Node

RX Node	0	1	2	3	···	N−1
0	0	70	70	65	···	0
1	70	0	78	61	···	0
2	70	78	0	42	···	0
3	64	62	42	0	···	0
···	···	···	···	···	···	···
N−1	0	0	0	0	···	0

Figure 12. RSSI matrix example.

Notice that the values on the diagonal are always 0, as the node cannot receive a message from itself. The RSSI values corresponding to the transmitting (TX) nodes from which the receiving (RX) node did not receive a message will all be 0 in the "rssi_values" list in the dataset; e.g., the N−1 node did not exist here. Similarly, the dataset does not include lines for receiving nodes that did not exist, since they did not send anything to be received by the gateway. For consistency we also fill these rows with 0. In processing, it might be easier to replace all zero values with a Not a Number (NaN) value to prevent including the zero values in calculations as they are not real measurement values.

To work with the measurement data, we need to align the data to a reference point. For most applications, we want to align the empty environment to 0 dB, which will give us the attenuation relative to the empty environment. We want to be able to recalibrate this reference periodically to compensate for changes in the environment that otherwise could induce an unwanted offset in the calibrated data. For this reason, we perform a calibration each night when such environments are usually empty. To compensate for external impacts and noise in the measurement, we average the collected RSSI values for each (directional) link in the RSSI matrix over a set interval. For this dataset we suggest using an interval between 3:00 and 3:15 (a.m.). In more dynamic environments a more dynamic approach can be used, e.g., looking for an interval with the lowest values and standard deviation.

To perform the calibration, we subtract the calibration matrix from each RSSI matrix. In other words, each measurement cycle gets calibrated using the calibration matrix for that day. When no calibration matrix for the current day can be created, we use the calibration matrix of the next day. This can happen due to incomplete data at the time of calibration. The calibration step can be described as a formula, shown in Equation (1), with $\boldsymbol{A}$ being the attenuation matrix, $\boldsymbol{R}$ being the RSSI matrix, and $\boldsymbol{C}$ being the calibration matrix. This happens for each measurement cycle c.

$$\boldsymbol{A}_c = \boldsymbol{R}_c - \boldsymbol{C} \quad \forall c \tag{1}$$

To calibrate the first day in both datasets, we use data from the following day. For calibrating the indoor environment on 17 May 2023, we advise using data from the night of 18 May 2023. On the 17th there was no data in the morning for nodes 7, 8, 9, and 10. The batteries were replaced during that day before the ground truth data was collected. Because of this, these links cannot be calibrated using data from the night before. For the outdoor environment, there was no data available in the morning of 11 June 2024 due to the installation of the sensors that day. We therefore recommend using the data from the following night (12 June 2024) to perform the calibration.

The format of the attenuation matrix still contains a lot of redundant information for feeding into a model. The use of 1-dimensional data is preferred for use in most linear models. We transform the 2-D attenuation matrices into 1-D attenuation vectors. Instead of simply flattening the entire matrix, we reduce the amount of data by implementing two measures. The first measure is to discard the zero values on the diagonal. The second measure is to average the two directions for the same link. This reduces the attenuation vector from a length of N^2 to a length of $\frac{N^2-N}{2}$, more than halving the size of the input vector. This reduces computational complexity, as well as reducing the measurement noise remaining in the input data by averaging two measurements of the same link, assuming the measurement noise is Gaussian-distributed. This leaves us with an attenuation vector $\boldsymbol{a}$, which contains a single value for each link in the network.

We can use these attenuation vectors, containing data on individual links, to train and validate models. However, even smaller and simpler models can be of value. For the most simple models, as will be the focus in the remainder of this section, we use a single attenuation value for the entire network as input. We obtain this single value by calculating the mean attenuation value for the attenuation vector. However, including each link in the network in the mean attenuation value is not always optimal. For this reason, we use a binary mask to select the links and nodes that align with the intended application when calculating the mean attenuation for the cycle.

5. Applications

5.1. Vehicle Detection

We use the attenuation of the radio links that cross the rail bedding to detect the presence of rail vehicles. The ground truth data includes the entry and exit events of rail vehicles in the station. We use these to label the measurement samples with a binary state: whether or not a rail vehicle is present. We used these labeled samples to train and evaluate a vehicle detection model. For the shown examples, we used a first-order logistic regression model.

$$\hat{y} = \sigma(\beta_1 \bar{\mathbf{a}} + \beta_0) \quad \textit{with} \quad \sigma(x) = \frac{1}{1+e^{-x}} \tag{2}$$

Equation (2) shows the logistic regression model employed in this use case. $\bar{\mathbf{a}}$ represents the mean attenuation of the selected links. β_0 and β_1 are the trained parameters. The σ function represents the sigmoid function mapping the output between zero and one. The output of the function $\hat{y}$ represents the predicted vehicle presence. We use the Limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm (L-BFGS) to estimate the parameters of the logistic regression model.

Figure 13 shows an evaluation graph for vehicle detection in the indoor environment. The model was trained using data from 17 May 2023. The evaluation results shown in the graph correspond to data from 17 July 2023. The black line shows the prediction output of the classification model. The green spans depict the period between the "in" and "out" timestamps from the ground truth data.

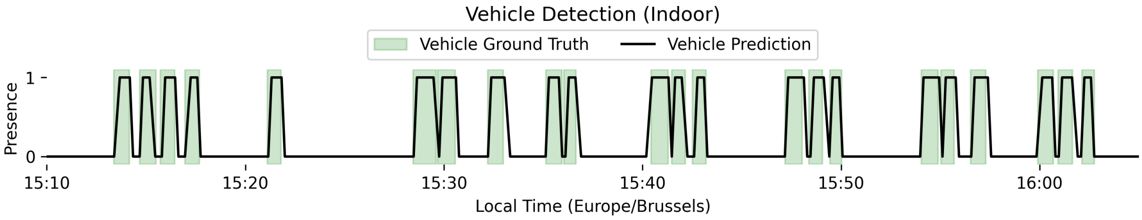

Figure 13. This graph shows an evaluation result of a logistic regression model for vehicle detection (in black) overlaid with the collected ground truth data (green spans) for the indoor environment.

Similarly, Figure 14 shows an evaluation graph for vehicle detection in the outdoor environment. The model was trained using data from 11 June 2024. The evaluation results shown in the graph correspond to data from 12 June 2024. The black line shows the prediction output of the classification model. The green spans depict the period between the “in” and “out” timestamps from the ground truth data.

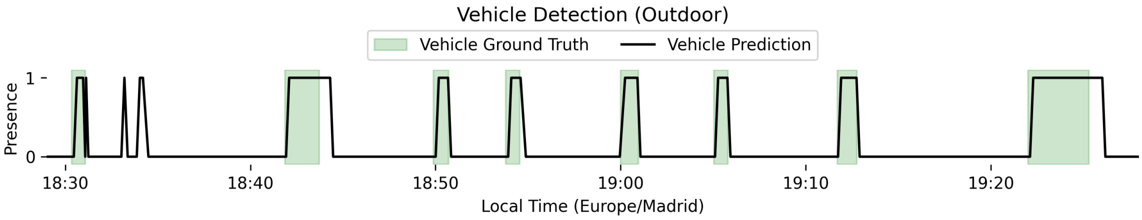

Figure 14. This graph shows an evaluation result of a logistic regression model for vehicle detection (in black) overlaid with the collected ground truth data (green spans) for the outdoor environment.

To validate the model, we used three-fold cross-validation between data of the different days, using two days for training and one day for evaluation. Because the classes suffer from an imbalance, we used undersampling of the training set to balance both classes during training. The evaluation still uses unbalanced data, but results are presented using the average F1-score and standard deviation between the folds. The results are shown in Table 4.

Table 4. Cross-validation results of vehicle detection application.

Case	F1-Score	StdDev
Indoor	0.897	0.0107
Outdoor	0.808	0.0632

5.2. People Count Estimation

The main intended application for this setup was to estimate the number of people standing on the platforms. Previous research [5] has shown that a linear relationship can be derived between measured attenuation between links crossing the platform and manual counts.

$$\hat{p} = \beta_1 \bar{\mathbf{a}} + \beta_0 \tag{3}$$

Equation (3) shows the linear regression model employed in this use case. $\bar{\mathbf{a}}$ represents the mean attenuation of the selected links. β_0 and β_1 are the trained parameters. The output of the function $\hat{p}$ represents the predicted number of people in the environment. We use Ordinary Least Squares (OLS) to estimate the parameters of the linear regression model.

Figure 15 shows the relationship between the measured mean attenuation and the number of people standing on the indoor platform when there was no rail vehicle present.

https://doi.org/10.3390/data11010021

For this graph, we only used the nodes mounted on the platform, ceiling, and bedding wall and discarded the nodes inside the tunnel. We also discarded the links between the ceiling and bedding nodes, and links between bedding wall nodes, as these links do not cross or pass by the platform close enough.

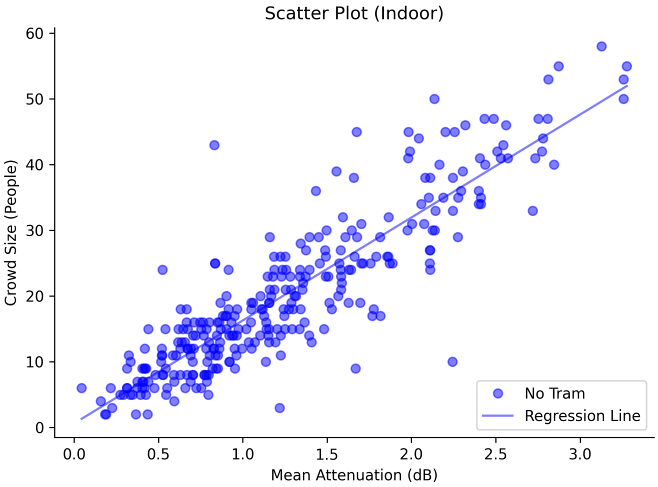

Figure 15. This scatter plot shows the relationship between the mean attenuation (between nodes on the platform, ceiling, and bedding wall) and the number of people when no vehicle was present at the platform for the indoor environment.

Similarly, Figure 16 shows the relationship between the measured attenuations between the sensor nodes and the number of people standing on the indoor platform when a rail vehicle was present. For this graph, we only used the nodes mounted on the platform and ceiling. None of these links cross the rail bedding, and they are only slightly affected by the presence of a rail vehicle.

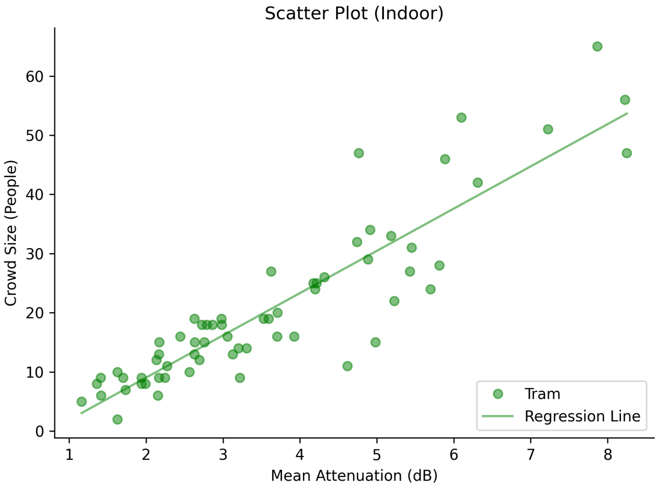

Figure 16. This scatter plot shows the relationship between the mean attenuation (between nodes on the platform and ceiling) and the number of people when a vehicle was present at the platform for the indoor environment.

Figures 17 and 18 show the distribution of the residuals in the indoor environment when there is no vehicle present (Figure 17) and when there is a vehicle present (Figure 18). The residuals have a standard deviation of 5.569 and 6.020, respectively.

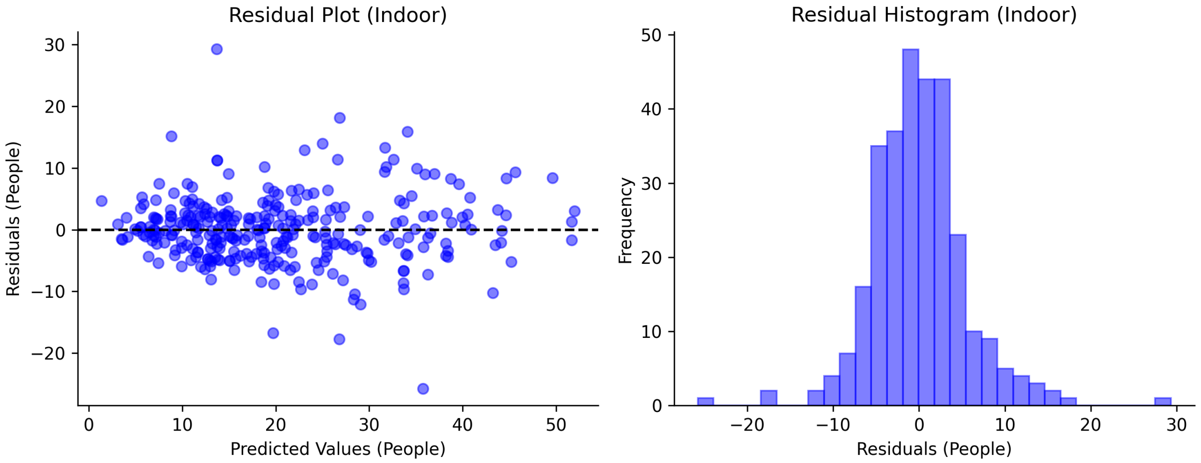

Figure 17. This residual plot and histogram show the distribution of the residuals for the indoor environment when no rail vehicle is present.

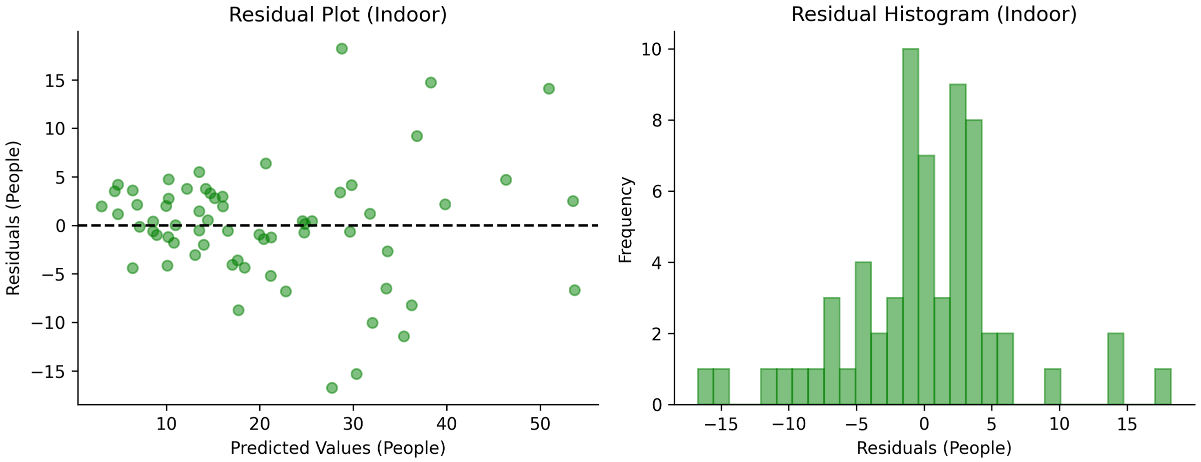

Figure 18. This residual plot and histogram show the distribution of the residuals for the indoor environment when a rail vehicle is present.

Figure 19 shows the relationship between the measured mean attenuation and the number of people standing on the outdoor platform when there was no rail vehicle present. For this graph, we used all the nodes but discarded the links between the ceiling and bedding nodes, as these links pass over instead of through the crowd.

Similarly, Figure 20 shows the relationship between the measured attenuations between the sensor nodes and the number of people standing on the outdoor platform when a rail vehicle was present. For this graph, we only used the nodes mounted on the platform and ceiling. None of these links cross the rail bedding, and they are only slightly affected by the presence of a rail vehicle. However, the number of samples in this dataset is limited, as only two of the three data collection periods contain people counts during the presence of rail vehicles and the number and frequency of trams at this station is fairly limited. Another feature observable in this graph is the occurrence of negative attenuation values. This can be explained by the difference between calibration in the empty environment, when no rail vehicle was present, and measurements on the graph for low values, where the constructive interference from reflection on the rail vehicle outweighs the attenuation caused by the people on the platform.

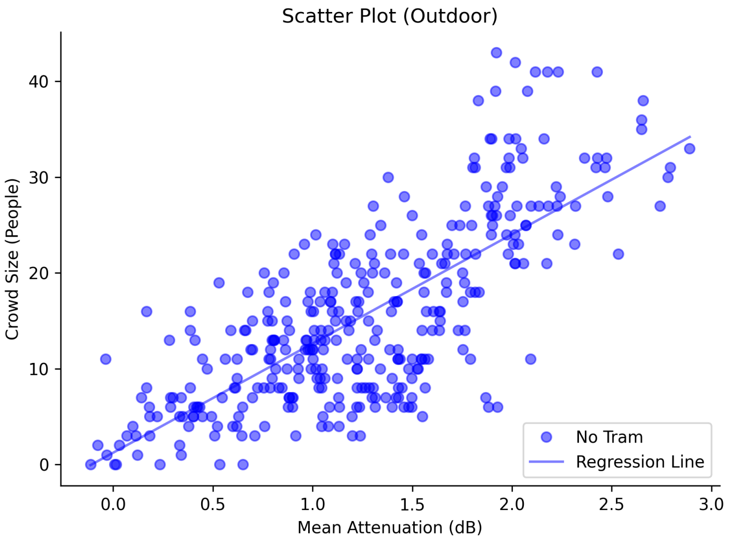

Figure 19. This scatter plot shows the relationship between the mean attenuation (between nodes on the platform, ceiling, and bedding) and the number of people when no vehicle was present on either side of the platform for the outdoor environment.

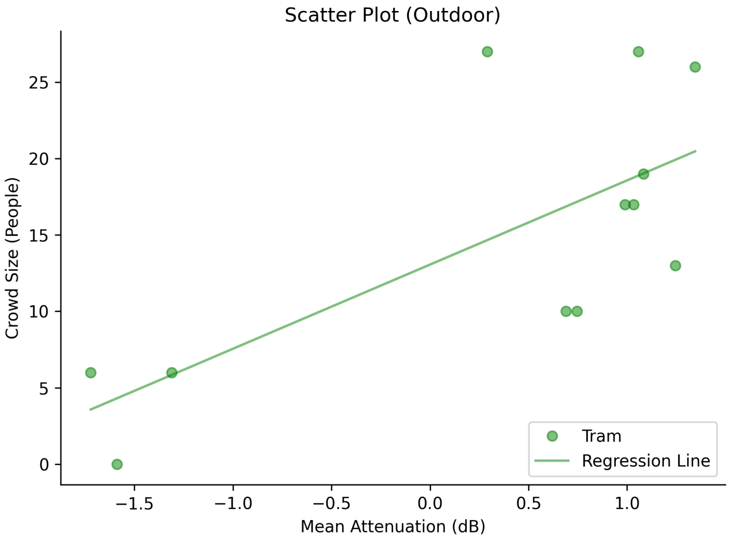

Figure 20. This scatter plot shows the relationship between the mean attenuation (between nodes on the platform and ceiling) and the number of people when a vehicle was present at the platform for the outdoor environment.

Figures 21 and 22 show the distribution of the residuals in the outdoor environment when there is no vehicle present (Figure 21) and when there is a vehicle present (Figure 22). The residuals have a standard deviation of 6.322 and 6.216, respectively.

Table 5 shows the Pearson correlation coefficients for the scatter plots of both environments. It can be observed that for all four cases a decent correlation exists. However, for the outdoor environment with a tram case, the significance is much lower due to the smaller sample size. The Pearson correlation for this case has a *p*-value of 0.00874, which is still considered statistically significant.

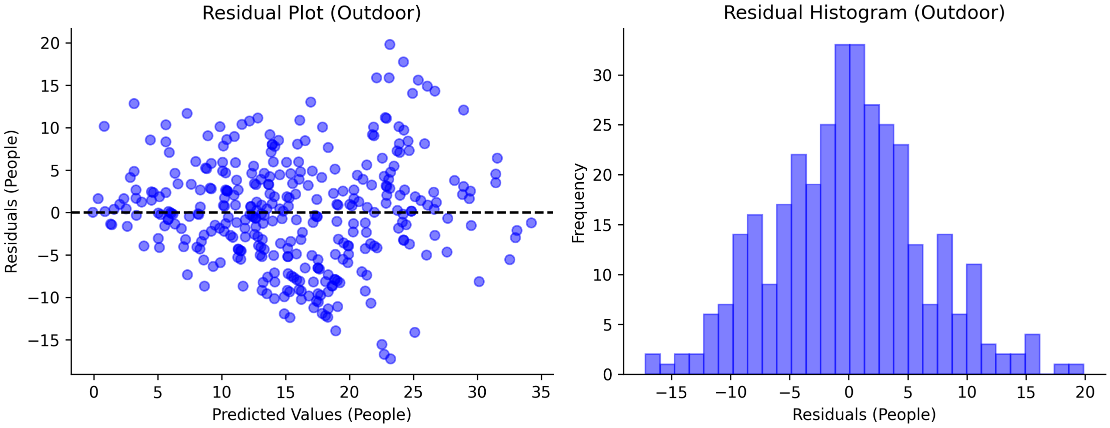

Figure 21. This residual plot and histogram show the distribution of the residuals for the outdoor environment when no rail vehicle is present.

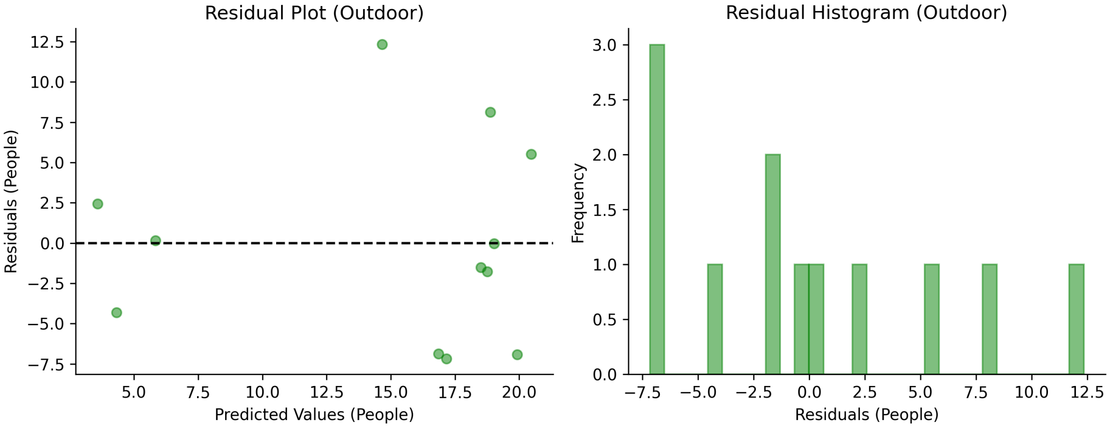

Figure 22. This residual plot and histogram show the distribution of the residuals for the outdoor environment when a rail vehicle is present.

Table 5. Correlation coefficients.

	Indoor		Outdoor	
	No Tram	**Tram**	**No Tram**	**Tram**
Pearson Correlation	0.890	0.901	0.746	0.717

To validate the model, we used k-fold cross-validation between data of the different days. In most cases, we use three-fold cross-validation, using two days for training and one day for evaluation. In one case, the outdoor environment with a vehicle present, we use two-fold cross-validation, since the third day has no people counts when the rail vehicle was present. The results are shown in Table 6, showing the average Mean Absolute Error (MAE) and the standard deviation between the folds.

Table 6. Cross-validation results of people counting application.

Case	k-Folds	MAE	StdDev
Indoor (no vehicle)	3	3.651	0.550
Indoor (vehicle)	3	4.554	1.820
Outdoor (no vehicle)	3	5.396	1.647
Outdoor (vehicle)	2	6.299	0.946

6. Conclusions

In this work, we presented a novel dataset on DFWS in public transportation environments. The dataset contains periodically collected RSSI values of radio links within a WSN deployed at two different light rail platforms. We demonstrated the potential of using RSSI values for applications such as rail vehicle detection through classification and estimating the number of people standing on a platform using regression analysis.

The dataset introduced in this work provides a baseline for further research into DFWS applications in public transportation environments. It broadens the scope of the technology beyond the traditional applications, such as large-scale events.

We believe that the dataset, in combination with this data descriptor, enables researchers to develop novel methods for processing DFWS data.

Future work may explore more sophisticated models and data processing techniques to improve accuracy and applicability in more challenging environments.

In this work, similarly to previous studies, we aggregated the measurements of individual links into a single value for each cycle. This step reduces the dimensionality of the input data, but this may also eliminate valuable information about the spatial distribution of the attenuations and reflections. Future work could attempt to utilize data from the different links independently to improve the system.

Expanding the dataset to additional environments might provide overarching insights and pave the way for developing generalized estimation models, thereby reducing or eliminating the need to train models for each environment separately.

Future work may examine how node locations impact the generalizability of the trained models across environments and which low- and high-level features are transferable between environments and thus could enable environment-agnostic models.

Author Contributions: Conceptualization, R.J. and B.B.; methodology, software, validation, formal analysis, and visualization, R.J.; data curation, R.J. and B.B.; writing—original draft preparation, R.J.; writing—review and editing, R.J., B.B. and R.B.; supervision, B.B. and R.B. All authors have read and agreed to the published version of the manuscript.

Funding: This project was partially funded by Agentschap Innoveren en Ondernemen (VLAIO) under Grant No. HBC.2021.0865.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used in this work is available at https://doi.org/10.5281/zenodo.18175571 under a Creative Commons Attribution (CC-BY) license.

Conflicts of Interest: Authors R.J. and B.B. were employed by the company CrowdScan. The remaining author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Zotano, M.A.G.; Bersini, H. A Data-driven Approach to Assess the Potential of Smart Cities: The Case of Open Data for Brussels Capital Region. *Energy Procedia* **2017**, *111*, 750–758. [CrossRef]
2. Jittrapirom, P.; Caiati, V.; Feneri, A.M.; Ebrahimigharehbaghi, S.; González, M.J.A.; Narayan, J. Mobility as a Service: A Critical Review of Definitions, Assessments of Schemes, and Key Challenges. *Urban Plan.* **2017**, *2*, 13–25. [CrossRef]
3. Rindone, C. Sustainable Mobility as a Service: Supply Analysis and Test Cases. *Information* **2022**, *13*, 351. [CrossRef]
4. Musolino, G.; Rindone, C.; Vitetta, A. Models for Supporting Mobility as a Service (MaaS) Design. *Smart Cities* **2022**, *5*, 206–222. [CrossRef]
5. Janssens, R.; Mannens, E.; Berkvens, R.; Denis, S. Device-Free Crowd Size Estimation Using Wireless Sensing on Subway Platforms. *Appl. Sci.* **2024**, *14*, 9386. [CrossRef]
6. Wilson, J.; Patwari, N. Radio Tomographic Imaging with Wireless Networks. *IEEE Trans. Mob. Comput.* **2010**, *9*, 621–632. [CrossRef]
7. Coluccia, A.; Mele, E.; Fascista, A. Radio Tomographic Imaging with Extremely Sparse and Location-Uncertain Transmitters. In Proceedings of the 33rd European Signal Processing Conference (EUSIPCO 2025), Isola delle Femmine, Palermo, Italy, 8–12 September 2025.
8. Ikegami, T.; Kim, M.; Miyake, Y.; Tsukada, H. Multipath-RTI: Millimeter-Wave Radio Based Device-Free Localization. *IEEE Access* **2024**, *12*, 42042–42054. [CrossRef]
9. Denis, S.; Bellekens, B.; Kaya, A.; Berkvens, R.; Weyn, M. Large-Scale Crowd Analysis through the Use of Passive Radio Sensing Networks. *Sensors* **2020**, *20*, 2624. [CrossRef] [PubMed]
10. Zhang, J.; Liu, J.; Wang, Z. Convolutional Neural Network for Crowd Counting on Metro Platforms. *Symmetry* **2021**, *13*, 703. [CrossRef]
11. Velastin, S.A.; Fernández, R.; Espinosa, J.E.; Bay, A. Detecting, Tracking and Counting People Getting On/Off a Metropolitan Train Using a Standard Video Camera. *Sensors* **2020**, *20*, 6251. [CrossRef] [PubMed]
12. Choi, J.H.; Kim, J.E.; Kim, K.T. People Counting Using IR-UWB Radar Sensor in a Wide Area. *IEEE Internet Things J.* **2021**, *8*, 5806–5821. [CrossRef]
13. Choi, J.W.; Quan, X.; Cho, S.H. Bi-Directional Passing People Counting System Based on IR-UWB Radar Sensors. *IEEE Internet Things J.* **2018**, *5*, 512–522. [CrossRef]
14. Rothkrantz, L. Person identification by smart cameras. In Proceedings of the 2017 Smart City Symposium Prague (SCSP), Prague, Czech Republic, 25–26 May 2017; pp. 1–6. [CrossRef]
15. Vandersmissen, B.; Knudde, N.; Jalalvand, A.; Couckuyt, I.; Bourdoux, A.; De Neve, W.; Dhaene, T. Indoor Person Identification Using a Low-Power FMCW Radar. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 3941–3952. [CrossRef]
16. Kaya, A.; Denis, S.; Bellekens, B.; Weyn, M.; Berkvens, R. Large-Scale Dataset for Radio Frequency-Based Device-Free Crowd Estimation. *Data* **2020**, *5*, 52. [CrossRef]
17. Yuan, Y.; Qiu, C.; Xi, W.; Zhao, J. Crowd Density Estimation Using Wireless Sensor Networks. In Proceedings of the 2011 Seventh International Conference on Mobile Ad-Hoc and Sensor Networks, Beijing, China, 16–18 December 2011; pp. 138–145. [CrossRef]
18. Cianca, E.; De Sanctis, M.; Di Domenico, S. Radios as Sensors. *IEEE Internet Things J.* **2017**, *4*, 363–373. [CrossRef]
19. Choi, H.; Fujimoto, M.; Matsui, T.; Misaki, S.; Yasumoto, K. Wi-CaL: WiFi Sensing and Machine Learning Based Device-Free Crowd Counting and Localization. *IEEE Access* **2022**, *10*, 24395–24410. [CrossRef]
20. De Sanctis, M.; Domenico, S.D.; Fioravanti, D.; Abellan, E.B.; Rossi, T.; Cianca, E. RF-Based Device-Free Counting of People Waiting in Line: A Modular Approach. *IEEE Trans. Veh. Technol.* **2022**, *71*, 10471–10484. [CrossRef]
21. Depatla, S.; Mostofi, Y. Crowd Counting Through Walls Using WiFi. In Proceedings of the 2018 IEEE International Conference on Pervasive Computing and Communications (PerCom), Athens, Greece, 19–23 March 2018; pp. 1–10. [CrossRef]
22. *ISO 8601-1:2019*; Date and Time—Representations for Information Interchange—Part 1: Basic Rules. International Organization for Standardization: Geneva, Switzerland, 2019. Available online: https://www.iso.org/standard/70907.html (accessed on 29 October 2025).
23. DASH7 Alliance. D7A Specification Version 1.2. 2018. Available online: https://www.dash7-alliance.org/product/dash7-alliance-protocol-specification-v1-2/ (accessed on 29 October 2025).
24. Weyn, M.; Ergeerts, G.; Berkvens, R.; Wojciechowski, B.; Tabakov, Y. DASH7 alliance protocol 1.0: Low-power, mid-range sensor and actuator communication. In Proceedings of the 2015 IEEE Conference on Standards for Communications and Networking (CSCN), Tokyo, Japan, 28–30 October 2015; pp. 54–59. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/data11010021