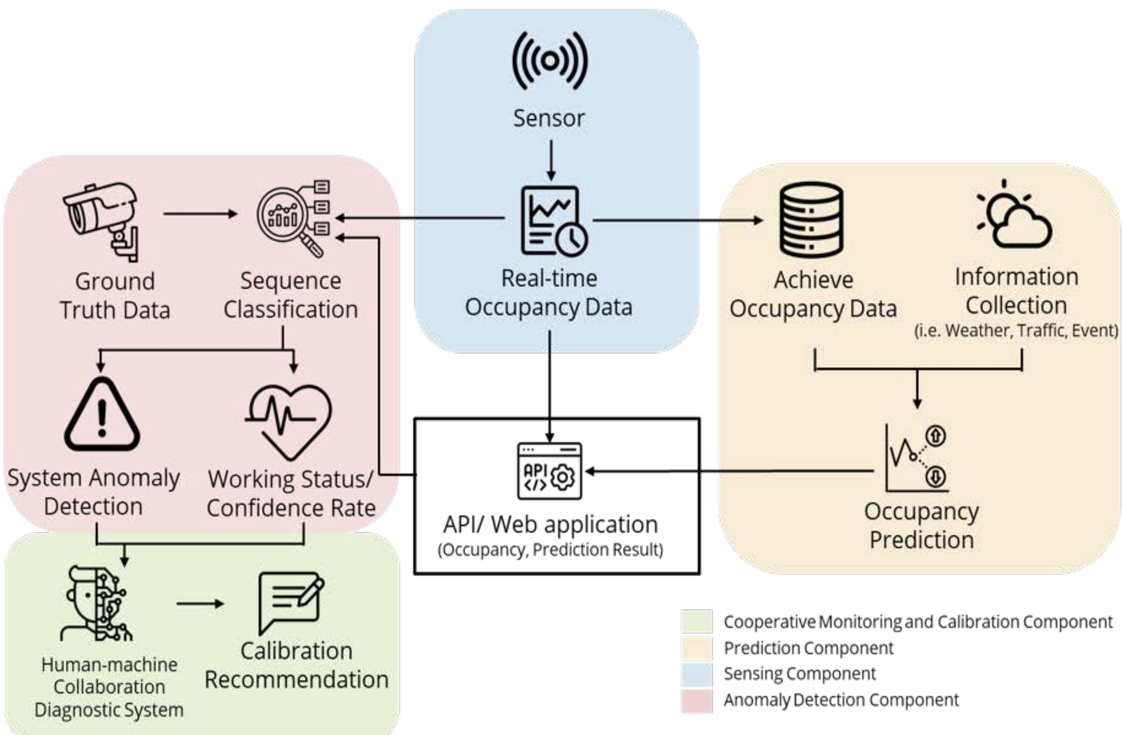
Smart and Cooperative Truck Parking Monitoring and Calibration System Empowered by Machine Learning

WA-RD 958.1

Yinhai Wang
Kehua Chen
Nutvara Jantarathaneewat

March 2026



Integration of the anomaly detection and calibration framework within the existing Truck Parking Information Management System (TPIMS) architecture

Washington State Department of Transportation
Office of Research & Library Services

WSDOT Research Report

Research Report
Agreement T1461 Task 97
WSDOT Smart and Cooperative Truck Parking Monitoring and Calibration System
Empowered by Machine Learning
WA-RD 958.1

Smart and Cooperative Truck Parking Monitoring and Calibration System Empowered by Machine Learning

by

Yinhai Wang
Department of Civil and Environmental Engineering

Kehua Chen
Department of Civil and Environmental Engineering

Nutvara Jantarathaneewat
Department of Civil and Environmental Engineering

University of Washington, Box 352700
Seattle, Washington 98195

Washington State Transportation Center (TRAC)
University of Washington, Box 359446
University Tower, 4333 Brooklyn Ave NE
Seattle, Washington 98195-9446

Washington State Department of Transportation Technical Monitor
Matthew Neeley

Prepared for
The State of Washington
Department of Transportation
Julie Meredith, Secretary

March 2026

Technical Report Documentation Page

<table>
<tr><td>1. Report No.
WA-RD 958.1</td><td>2. Government Accession No.</td><td colspan="2">3. Recipient's Catalog No.</td></tr>
<tr><td colspan="2" rowspan="2">4. Title and Subtitle
Smart and Cooperative Truck Parking Monitoring and Calibration System Empowered by Machine Learning</td><td colspan="2">5. Report Date
March 2026</td></tr>
<tr><td colspan="2">6. Performing Organization Code</td></tr>
<tr><td colspan="2">7. Author(s)
Yinhai Wang, Nutvara Jantarathaneewat, and Kehua Chen</td><td colspan="2">8. Performing Organization Report No.</td></tr>
<tr><td colspan="2" rowspan="2">9. Performing Organization Name and Address
Washington State Transportation Center
University of Washington, Box 359446
University Tower, 4333 Brooklyn Ave NE
Seattle, WA 98195-9446</td><td colspan="2">10. Work Unit No. (TRAIS)</td></tr>
<tr><td colspan="2">11. Contract or Grant No.
Agreement T1461-97</td></tr>
<tr><td colspan="2" rowspan="2">12. Sponsoring Organization Name and Address
Washington State Department of Transportation
Transportation Building, MS 47372
Olympia, Washington 98504-7372 14
David Strich, Project Manager, 360-705-7943</td><td colspan="2">13. Type of Report and Period Covered
Final Research report</td></tr>
<tr><td colspan="2">14. Sponsoring Agency Code</td></tr>
<tr><td colspan="4">15. Supplementary Notes
Conducted in cooperation with Federal Highway Administration, U.S. Department of Transportation.</td></tr>
<tr><td colspan="4">16. Abstract
Accurate truck parking availability information is essential for improving freight efficiency and safety. Many Truck Parking Information Management Systems (TPIMS) rely on in/out counting sensors to estimate parking occupancy; however, these sensors may accumulate counting errors over time as a result of missed detections, irregular parking behavior, or environmental conditions. These errors can significantly reduce the reliability of reported parking availability. This study proposed an anomaly detection and calibration framework to improve the reliability of truck parking occupancy estimation. The framework integrates prediction-based anomaly detection, multi-sensor comparison, and sequence classification to monitor sensor performance. A time-series prediction model learns normal occupancy patterns, and deviations between predicted and observed values are used to detect potential anomalies. Sensor status is then classified into three levels—OK, Moderate, and Suspect—based on discrepancies between radar sensors and a reference camera sensor. A confidence score is further estimated to quantify the reliability of the classification. When strong evidence of sensor error is detected, the system generates calibration recommendations by leveraging reference sensor measurements to correct radar-based estimates. The framework was integrated into an operational TPIMS environment and visualized through a monitoring dashboard for system operators. Results demonstrated that the approach can effectively identify sensor anomalies and reduce occupancy estimation errors, improving the reliability of truck parking information systems.</td></tr>
<tr><td colspan="2">17. Key Words
Truck parking, parking information management system, calibration</td><td colspan="2">18. Distribution Statement
No restrictions.</td></tr>
<tr><td>19. Security Classification (of this report)
Unclassified.</td><td>20. Security Classification (of this page)
Unclassified.</td><td>21. No. of Pages
75</td><td>22. Price
NA</td></tr>
</table>

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and accuracy of the data presented herein. This document is disseminated through the Washington State Department of Transportation. The contents do not necessarily reflect the views or policies of Washington State Department of Transportation or the Federal Highway Administration. This report does not constitute a standard, specification, or regulation.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLES

1. INTRODUCTION AND BACKGROUND

In today's practice, the most common ways for Truck Parking Information Management Systems (TPIMS) to determine the occupancy of a parking lot are individual space-by-space occupancy detection and in/out counting (Vital et al., 2020). By investigating the current TPIMS based on in/out counting across the U.S., two obvious limitations are found and summarized as follows (Vital et al., 2020; FMCSA, 2019; Yang et al. 2021; Analysis Division FMCSA, 2011):

- Lack of sensing error calibration (Vital et al., 2020; Analysis Division FMCSA, 2011; and FMCSA, 2019). For the in/out counting system, a miscount, i.e., missed detection or false alarm, will be carried on until a manual check is made or an opposite miscount is made. Without timely system calibration, the errors will accumulate and generate more significant mistakes. For a small or medium-sized truck parking lot, limited parking spaces will lead to errors that will quickly build up and reach a significant percentage of the total capacity. Therefore, a reliable and automatic sensing calibration algorithm for truck parking can hugely increase in/out counting accuracy and reliability.
- The ignorance of ground-truth data collection and sensing quality monitoring (Vital et al., 2020; FMCSA, 2019). In today's TPIMS, sensing results are always used as inputs for downstream tasks and disseminated to road users directly (Yang, 2021). However, the truck parking sensing result cannot be 100 percent accurate and reliable. Sometimes the abnormal status of a sensor or extreme weather conditions can cause errors. To avoid error accumulation and monitor the system's working performance, an independent system to collect ground-truth data and monitor sensing quality is necessary.

Based on these current limitations of TPIMS, an automated framework to monitor and calibrate the in/out counting system for truck parking is required. However, three key challenges must be addressed. The first challenge is how to collect the ground-truth data and summarize the sensors' errors for a given truck parking lot. At present, many parking lots that are based on in/out counting are likely to have some manned facilities, with people present, who could be able to help with ground-truth correction (Vital et al., 2020; FMCSA, 2019; Analysis Division FMCSA, 2011). However, it would be much better for "ground-truth data" to be automatically extracted by a reliable sensing system based on advanced technologies. The second challenge is how to detect anomaly status and measure the error levels for a given sensing system. The third challenge is how to provide a recommended calibration solution to eliminate sensing errors by ground-truth references and advanced technical tools.

Current truck parking supply is not sufficient to meet the increasing demand for truck drivers in the U.S. The state of California has one of the largest inventories of truck parking spaces in the county. However, because of the large highway network, large number of seaports, and heavy truck traffic, this inventory falls short of the growing need. As of 2000, California had estimated the state's total number of truck parking spaces as 8,600, which is 38 percent of the estimated demand of 22,700 parking spaces (Analysis Division FMCSA, 2011; Rodier and Shaheen, 2007). Similarly, a 2015 report by the Virginia Department of Transportation calculated a statewide deficit of nearly 5,000 parking spaces, which means that the state only satisfies roughly 60 percent of the calculated demand (12,500 spaces) (Horn, 2015). According to the U.S. Department of Transportation (USDOT) and Federal Highway Administration (FHWA), 36 states are experiencing shortages in rest areas, either public or private, which negatively affect truck parking as well (Rodier and Shaheen, 2007; WSDOT, 2016; Federal Highway Administration, 2018; Bayraktar et al., 2015).

The shortage of truck parking has negative consequences and leads to traffic safety issues such as illegal parking, unsafe driving, and increased costs. This shortage increases the value of every single truck parking space and makes it paramount to offer truck drivers real-time and accurate predictions of truck parking availability information *ahead* of their arrival. This will allow them to make a safer and more reliable schedule of their stops. Washington State Department of Transportation (WSDOT) is aware of the growing challenge of truck parking shortage and seeks cost-effective solutions to enhance the sensing and usage efficiency of existing truck parking facilities (WSDOT, 2016; Luley et al., 2018).

Functions	Type	Iowa	Ohio	Michigan	Kentucky	Wisconsin	Indiana	Kansas	Minnesota
Procurement	Public	DBOM	DBOM	DBB	DBB	DBB	DBB	DBB	DBB
	Private		N/A	DBOM		N/A	N/A	N/A	N/A
Data Collection Method	Public	Functional Requirements	Functional Requirements	In/Out	In/Out	In/Out	In/Out	Space-by-Space	Space-by-Space
	Private		N/A			N/A	N/A	N/A	N/A
Data Collection Technology[6]	Public	Functional Requirements	Functional Requirements[1,2]	Video	Magnetometer	Magnetometer	Magnetometer	Video[2]	Magnetometer
	Private		N/A	Video		N/A	N/A	N/A	N/A
Operations & Maintenance	Public	Third Party	Third Party	Internal[3]	Third Party[4]	Third Party	Internal	Third Party	Internal
	Private		N/A	Third Party		N/A	N/A	N/A	N/A
	Sign Operations	N/A	Internal	Internal	Internal	Internal	Internal	Internal	Internal
Data Analytics & Sharing	Processing	Third Party	Third Party	In-House ATMS[7]	In-House ATMS	Third Party	In-House ATMS	In-House[8]	In-House ATMS
	Software	Not Developed	Not Developed	Current	Not Developed	Current	Not Developed	Not Developed	Needs Additional Development
	Sharing Format	XML Data Feed	XML Data Feed	XML Data Feed	XML Data Feed	XML Data Feed	XML Data Feed	XML Data Feed	XML Data Feed
Information Dissemination	Signs	No Signs	DTPS	DTPS	DTPS	DTPS	DTPS	DTPS	Full-Matrix Color DMS
	Website	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]
	Mobile Website/ Mobile App	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]	State and Third Party[5]

Notes:
[1]Wants to own CCTV at Rest Areas
[2]Prefers to stay out of pavement
[3]Would consider Third Party for public sites
[4]Embedded Staff
[5]Data is available and a Third Party Provider will do this for us with no cost to us
[6]All states are assumed to have video surveillance at all monitored sites
[7]Third Party for private sites
[8]IRIS software run on KDOT server

Figure 1-1.: Current deployed truck parking information management systems (TPIMS) core functions matrix review summary (Luley et al., 2018)

The most common way to access the occupancy of a parking lot is individual space occupancy detection (Vital et al., 2020; Luley et al., 2018). In general, the space-by-space detection method consists of checking the occupancy of every single parking space and then counting the number of occupied spots (Vital et al., 2020; Rodier and Shaheen, 2007; WSDOT, 2016; Luley et al., 2018). The advantage of this method is that a sensing error in a single parking space does not have a compounding impact on the total lot occupancy and does not accumulate. This method makes it easier to consider detachable parts, such as trucks dropping or picking up trailers, and to deal with passenger vehicles parking in truck parking spaces. The main drawback of this method is that the number of sensors needed is proportional to the number of parking spaces, which increases the installation and maintenance costs for large facilities (Vital et al., 2020).

Another truck parking sensing system is the in/out counting system. Given the parking lot's initial occupancy, this method relies on counting the number of vehicles that enter and exit the lot, then calculating how many vehicles are currently parked. The advantage of this method is its scalability (Vital et al., 2020; Federal Highway Administration, 2018; Luley et al., 2018; Lin et al., 2017). Regardless of the parking area's size, the infrastructure needed is limited by the number of entrances/exits. A downside to this method is that the absence of parking space markings and the misuse of parking spaces do not affect the occupancy counting even though the total capacity of the parking lot is affected, compromising availability estimates (Vital et al., 2020; Federal Highway Administration, 2018; Luley et al., 2018; Lin et al., 2017; Haque et al., 2017; Sun et al., 2018). Focusing all resources on the access points also creates some disadvantages.

One such disadvantage is that any error in this method is cumulative and can only be corrected with ground-truth information (Vital et al., 2020; Federal Highway Administration, 2018; Luley et al., 2018; Haque et al., 2017). A miscount will be carried on until a ground-truth correction is made or a miscount in the opposite direction is made. Even if the sensors at the entry/exit points do not fail, errors can still appear in certain situations. For example, if a truck enters carrying a trailer and drops it in a parking space before leaving, then the system would not consider the trailer in its calculation and would undercount the occupancy. These detachable parts are invisible to this type of system. So, this method needs frequent ground-truth corrections (Vital et al., 2020; Federal Highway Administration, 2018; Luley et al., 2018). Another problem, which is usually solvable through extra sensors and/or software updates, is the behavior of drivers at the entrance/exit points. Behaviors such as tailgating and using the wrong lanes to enter/exit can affect detection, so the system's design must take these situations into account to avoid errors.

2. RESEARCH OBJECTIVES

The primary goal of this research was to create a truck parking monitoring and calibration system empowered by machine learning for the in/out truck parking counting system. Four components are integrated into the proposed scheme as shown in Figure 2-1. These include (1) the truck parking lot sensing component, (2) the information collection component, (3) the sensing anomaly detection component, and (4) the cooperative monitoring and calibration component.

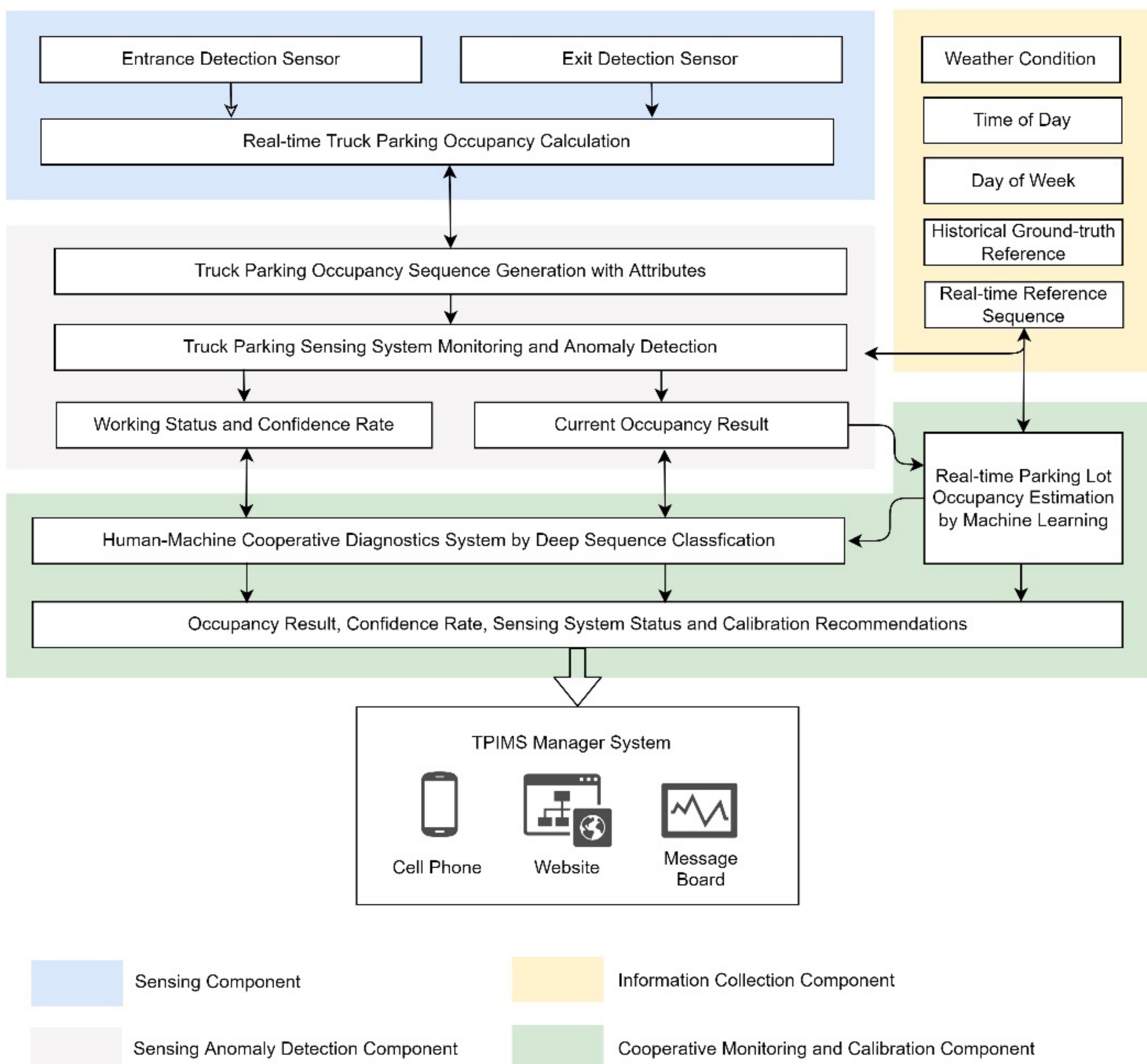


Figure 2-1 The proposed smart and cooperative truck parking sensing diagnostics and calibration system powered by deep learning

Counting data can be obtained from installed sensors, (i.e., radar, cameras, etc.) at the entrance and exit of a truck parking lot. A separate video surveillance system will also be installed to collect ground-truth data. Then, real-time parking occupancy can be calculated, and the anomaly status can be identified based on a comparison with ground-truth records. The next step is to send the collected information, such as real-time or historical ground-truth occupancy sequence and the anomaly detection results into the cooperative calibration component. Finally, the proposed system will generate the calibrated occupancy result, confidence rate, sensing system status, and calibration recommendations. To achieve this goal, the research team identified four objectives:

1) Propose a real-time truck parking sensing system status anomaly detection framework.
2) Propose a real-time parking lot occupancy estimation system empowered by deep learning.
3) Propose a human-machine cooperative monitoring and calibration algorithm empowered by sequence classification.
4) Build a live website to visualize the truck parking sensing monitoring and calibration results.

A significant benefit of this research is providing a pilot truck parking sensing monitoring and calibration system for the TPIMS based on in/out counting. With the proposed system, the calibrated truck parking occupancy result, real-time sensing system operation status, confidence rate and calibration recommendations can be generated in a human-machine cooperative approach. Current efforts heavily rely on regular, manual calibration, which is labor intensive, inefficient, and difficult to scale up. This innovative research will monitor and calibrate the installed sensing system by offering a pipeline with advanced technologies, including deep learning and cooperative AI methods to support the TPIMS. The proposed system follows suggestions and findings from a recently completed WSDOT-sponsored project (Wang et al., 2025). Furthermore, this project will support future projects related to truck parking innovative technology.

3. LITERATURE REVIEW

3.1 TRUCK PARKING DETECTION TECHNOLOGIES

Truck-parking detection systems rely on two primary counting methods—individual space (slot-by-slot) detection and in/out counting—each suited for different facility sizes and operational needs. Slot-by-slot systems directly monitor each parking space by using technologies such as magnetic sensors, radar, or video imaging. These provide good accuracy and avoid cumulative errors, but they can become costly for large truck lots because of the need for many sensors. In contrast, in/out counting systems install sensors at entrances and exits to track vehicles entering and leaving. These solutions are highly scalable and less dependent on pavement markings, although they are prone to error accumulation over time and require periodic ground-truth corrections (Wang et al., 2022).

A wide range of sensing technologies has been used in pilot deployments. Magnetic sensors offer low-power, wireless, and durable designs but may require multiple sensors per space (Bayraktar et al., 2015; Gu et al., 2012; Guan et al., 2019; Sun et al., 2018; Yang et al., 2019). Laser and radar sensors provide precise detection and classification, especially in challenging lighting or weather conditions, although they can be sensitive to physical obstructions and environmental effects (Cook et al., 2014; EasyWay Project 2015; López-Jacobs et al., 2013; Siemens 2013). Video image processing, including AI-enhanced and 3D reconstruction methods, enables coverage of large areas with fewer devices and can resolve occlusion issues, but it requires higher bandwidth and processing capacity (Bong et al., 2008; Boureau et al., 2010; Gertler and Murray 2011; Ke et al., 2020; Modi et al., 2011; Morris et al., 2017; Morris et al., 2018; Woodrooffe et al., 2016). GPS-based methods provide insights into regional parking demand but lack the penetration and granularity needed for real-time occupancy estimation (Ford Torrey and Murray, 2017; Haque et al., 2016; Haque et al., 2017; Williams 2019; Xu et al., 2013).

Communication systems vary from wired to cellular and LoRa-based networks, depending on vendor design (Lin et al., 2017). Information dissemination channels include roadside variable message signs (VMS), websites, mobile apps, and in-cab devices. However, user surveys indicate limited awareness among drivers, highlighting persistent challenges in effective information delivery (Woodrooffe et al., 2016).

Pilot deployments in the U.S. and Europe have demonstrated that most modern technologies can achieve over 95 percent detection accuracy when properly calibrated (Bayraktar 2015; Bressler and Schuster, 2018; Dierke et al., 2016; EasyWay Project 2015; Fallon and Howard, 2011; I-95 Corridor Coalition 2018; Lopez-Jacobs et al., 2013; Morris et al., 2017; Nedap 2021; Washburn et al., 2016; Woodrooffe et al., 2016; Yang et al., 2021). However, each technology remains sensitive to site-specific factors such as lot geometry, driver behavior, environmental conditions, and installation quality. Studies have consistently shown that regular supervision, calibration, and design adjustments (such as clear markings and structured layouts) are crucial to performance. Overall, although multiple technologies have proved effective, large-scale, unattended deployment remains challenging, and the industry continues to face knowledge gaps regarding long-term reliability, scalability, and cross-site performance.

Several vendors provide sensing solutions commonly used in intelligent transportation systems (ITS):

3.1.1 Radar-Based in/out Counting Sensors

Radar-based in/out counting sensors are widely used for estimating truck parking occupancy because they provide reliable vehicle detection under a wide range of environmental conditions while requiring minimal maintenance. These systems are typically installed near the entrance and exit of a parking facility, and they estimate occupancy by counting vehicles entering and leaving the lot. Radar sensors detect vehicles using microwave signals and can measure parameters such as vehicle presence, speed, volume, and classification. Because they are non-intrusive roadside sensors, installation is relatively simple in comparison with pavement-embedded loop detectors, and they can operate effectively in rain, fog, snow, or low-light conditions.

Wavetronix provides radar-based traffic detection sensors widely used in transportation applications (IoT ONE Limited 2022). Its SmartSensor and Expanse systems use Digital Wave Radar technology to monitor traffic on highways, arterial roads, and intersections (G. H. S. a. G. G. S. Mitsuru Sait, 2016; Wavetronix LLC 2016). These sensors can detect multiple lanes of traffic and collect parameters such as vehicle volume, speed, occupancy, classification counts, headway, and direction counts. The XP20 side-fire radar sensor, part of the Expanse system, can monitor traffic across up to 22 lanes and provide both aggregated and per-vehicle data that include speed, length, vehicle class, and lane assignment (Figure 3-1). The system operates in the

24.0–24.25 GHz K-band radar frequency and achieves detection accuracy greater than 95 percent per direction. Wavetronix sensors are designed for long-term roadside deployment with minimal maintenance requirements, featuring corrosion-resistant housing and an expected mean time between failure of approximately ten years. The sensors communicate through Ethernet connections and support firmware upgrades and system monitoring through the Expanse network management platform. In truck parking applications, two sensors are typically installed at facility entrances and exits to count vehicle movements and estimate available parking spaces in real time.

Figure 3-1 XP20 Radar Sensor From "Wavetronix Support". by Wavetronix LLC, 2023 (https://www.wavetronix.com/support)

ImageSensing Systems provides another radar-based detection solution through its RTMS Echo sensor. This device is a side-fire radar system mounted on roadside poles that detects and measures traffic flow parameters such as vehicle volume, speed, occupancy, and classification (Image Sensing, 2024). The RTMS Echo operates in the 24 GHz radar band and can monitor multiple detection zones across roadway lanes (Figure 3-2). For parking applications, the sensor can be installed at the entrance of a parking facility to count trucks entering and exiting the lot. The system can also be paired with a camera to provide visual verification of roadway conditions. RTMS Echo sensors are designed for reliable performance in all weather conditions and require minimal maintenance. The sensor supports Ethernet communication and stores large amounts of vehicle detection data internally, allowing transportation agencies to access both real-time and historical traffic information. Configuration and monitoring can be performed through the Echo Web application, which enables automatic detection zone calibration and system setup. In addition, ImageSensing provides a Virtual Control Center platform that allows agencies to visualize traffic data and analyze patterns in traffic volume, speed, occupancy, and other operational metrics.

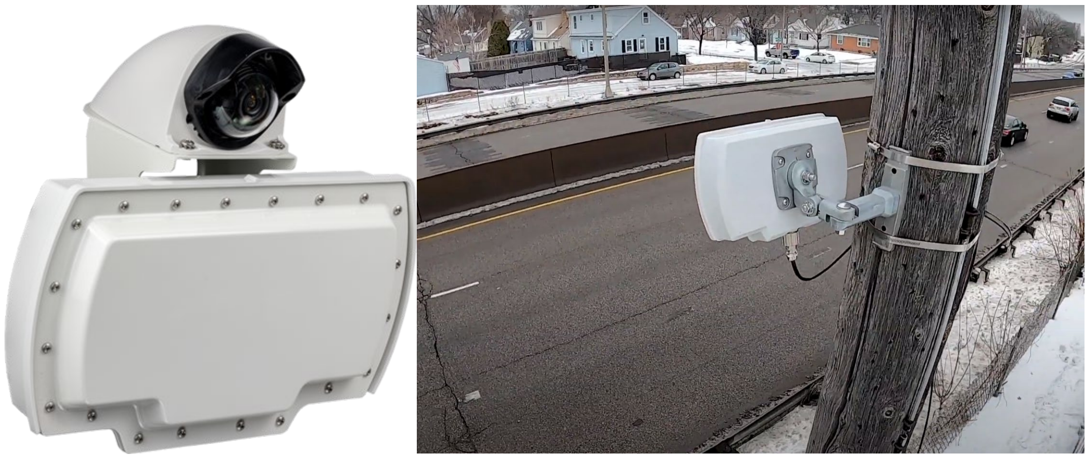


Figure 3-2 The RTMS Echo plus camera (left) and installation (right). From "RTMS Echo + Camera." by Image Sensing Systems, Inc. (2022) (https://www.imagesensing.com/solutions/radar-detection/radar-detection-products/rtms-echo-camera.html)

3.1.2 Video–Radar Fusion Sensors

Video–radar fusion sensors combine radar detection with camera-based image analytics to improve vehicle detection accuracy and provide richer traffic information. Radar sensors are highly reliable in adverse weather and low-visibility conditions, while cameras provide additional contextual information such as vehicle classification, lane positioning, and visual verification. By integrating both sensing modalities, fusion systems can achieve good detection accuracy and better object tracking than single-sensor solutions. These systems are commonly used in advanced traffic monitoring applications and are increasingly applied in truck parking monitoring systems where accurate counting and vehicle classification are required.

Smartmicro develops radar and video-based traffic sensors for smart transportation systems (Figure 3-3) (Smartmicro, 2024). Its traffic detection solutions use ultra-high-definition radar technology capable of measuring range, speed, and azimuth angle of vehicles in real time. One example is the TRUGRD STREAM system, which combines radar sensing with integrated camera capabilities. The radar operates within the 24.0–24.25 GHz frequency range and provides 3D or 4D measurements with high spatial resolution. These sensors can detect vehicles within ranges of approximately 1.5 m to 300 m and monitor traffic across up to 12 lanes simultaneously. The system is capable of tracking up to 256 traffic objects at the same time and provides detailed information that includes vehicle position, speed, lane assignment, and trajectory. Smartmicro sensors support Ethernet-based communication and can be connected to controller interface boards to coordinate multiple sensors and prevent interference. Configuration and data access are provided through the Traffic UI software and Smartmicro’s application programming interface (API), which allows integration with traffic management platforms and other ITS systems. In truck parking applications, the system can monitor entrance and exit points or designated detection zones to determine parking occupancy levels.

Figure 3-3 Traffic Detection Radar (left and center) and Video-Radar Fusion Sensor (right). From “UMRR-11 Type 44 Products sheet” by Smartmicro. (2022) (https://www.smartmicro.com/downloads)

Omnisight provides an integrated radar and video analytics solution specifically designed for vehicle tracking and parking monitoring applications (Figure 3-4) (Omnisight, 2024). The system uses millimeter-wave radar technology combined with edge-based video analytics to detect and track vehicles in three-dimensional space. The Mega Radar sensor operates in the 60–64 GHz frequency band using frequency-modulated continuous-wave radar technology and can achieve detection accuracies of approximately 98 percent. The sensor includes an integrated high-definition camera and on-board processing capabilities powered by a quad-core processor, allowing vehicle detection and classification to be performed directly at the sensor. This edge computing capability reduces communication bandwidth requirements and enables real-time monitoring of parking lot usage, staging areas, loading zones, and other truck parking facilities. The system communicates with servers using TCP/IP or HTTPS protocols and can transmit detection data through APIs for integration with traveler information systems. Installation is relatively simple, requiring only a power-over-Ethernet connection and network access. Omnisight also provides web-based tools for system monitoring, vehicle counting verification, and hardware management. These features allow transportation agencies to deploy scalable truck parking monitoring systems that provide accurate real-time parking availability information to drivers and system operators.

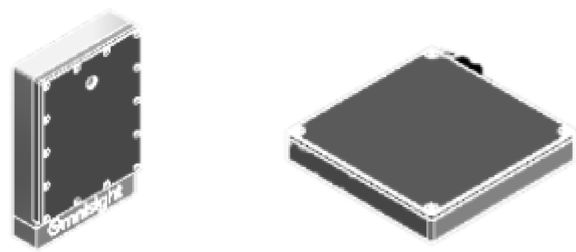


Figure 3-4 “Mega Radar” (left), and “Pico Radar” (right) (https://omnisightusa.com/)

3.2 ANOMALY DETECTION IN TRANSPORTATION SYSTEMS

Transportation systems increasingly rely on sensor networks that continuously generate large volumes of time-series data (Boniol et al., 2021; Boniol et al., 2022; Boniol et al., 2023; Boniol et al., 2024a; Boniol et al., 2024b; Bui et al., 2018; Fox, 1972; Abdul-Aziz et al., 2010). These data streams are often affected by measurement errors, environmental factors, communication failures, and unexpected operational events. As a result, anomaly detection has become an essential component of transportation data analytics, enabling systems to identify abnormal sensor readings, unusual traffic patterns, and potential system failures. In general, anomalies refer to observations that deviate significantly from expected patterns or previously observed behavior in the data (Boniol et al., 2024c). Such anomalies may arise from the complexity of data generation processes or from imperfections in measurement systems.

Time-series anomaly detection differs from traditional anomaly detection because observations are temporally ordered and often interdependent. Unlike static datasets, anomalies in time-series data cannot always be detected by examining a single data point (Fox 1972). Instead, anomalies often appear as unusual temporal patterns or subsequences that deviate from normal system behavior over time. Consequently, many anomaly detection approaches analyze temporal structures or compare subsequences within the time series rather than evaluating isolated observations (Boniol et al., 2024c).

Anomalies in time-series data can generally be categorized into three main types (Kampakis, 2022). Point anomalies refer to individual observations that significantly deviate from the expected distribution of the data. Contextual anomalies occur when a data point appears normal in isolation but become abnormal when evaluated within a specific temporal context, such as seasonal traffic variations or daily patterns. Collective anomalies involve sequences of observations that together form an abnormal pattern even though individual values may appear normal. In addition, many real-world systems involve multivariate time-series data, where anomalies may only become apparent when relationships among multiple variables are considered (Tsay et al., 2000).

Another important distinction in anomaly detection relates to the origin of the anomaly (Boniol et al., 2024c). Some anomalies arise from measurement noise or sensor errors, which should typically be removed during data cleaning. Other anomalies represent real events, such as incidents, equipment malfunction, or unusual traffic demand, and these should therefore be detected and analyzed rather than removed. Because the statistical distribution of real-world sensor data is often unknown or difficult to estimate, many modern approaches rely on machine learning algorithms to automatically learn normal behavior patterns from the data.

3.2.1 Detection Paradigms

Anomaly detection methods can generally be categorized based on the availability of labeled data (Du et al., 2025). Unsupervised approaches assume that no labeled anomaly examples are available and attempt to detect abnormal patterns solely from the structure of the data. Semi-supervised approaches assume that only normal data are available during training, and the model learns to identify deviations from the learned normal patterns. Supervised approaches require labeled examples of anomalies, and the model learns to classify abnormal patterns directly. In many real-world applications, anomalies occur infrequently and labeled datasets are limited, making unsupervised or semi-supervised approaches particularly attractive.

A typical anomaly detection pipeline consists of several stages (Boniol et al., 2024c). First, a data processing stage prepares the time-series data by dividing them into subsequences or sliding windows. Next, a detection model analyzes these subsequences using statistical, machine learning, or deep learning methods. The model generates an anomaly score representing the degree of abnormality for each observation or subsequence. Finally, a post-processing stage applies thresholds or additional filtering to identify significant anomalies.

3.2.2 Classical Anomaly Detection Methods

Traditional anomaly detection techniques for time-series data are typically based on statistical or distance-based approaches (Chang et al., 1988; Tukey 1977).

Distance-based methods identify anomalies by measuring the similarity between subsequences. Observations that are far from their nearest neighbors are considered potential anomalies. Common proximity-based approaches include the k-nearest neighbors (KNN) method and Local Outlier Factor (LOF), which measures how isolated a data point is relative to its surrounding neighborhood (Hawkins 1980; Breunig et al., 2000). Clustering-based methods such as k-means or DBSCAN partition the data into clusters and treat observations that do not belong to any cluster as anomalies (Sander et al., 1998). Another approach, known as discord-based detection, identifies subsequences that have the largest distance from their nearest neighbors.

Density-based methods estimate the statistical distribution of the data and identify observations that lie in low-density regions. Distribution-based methods include techniques such as Minimum Covariance Determinant (Rousseeuw 1984) and One-Class Support Vector Machines (Ma and Perkins, 2003), which attempt to learn the boundaries of normal data distributions. Histogram-based outlier detection is another method designed to handle highly unbalanced datasets (Goldstein and Dengel, 2013). Other density-based approaches represent time-series relationships using graphs, trees, or encoding techniques such as principal component analysis (PCA), Hidden Markov Models (HMM), and Bayesian networks.

3.2.3 Prediction-Based and Reconstruction-Based Methods

More recent approaches to time-series anomaly detection rely on predictive modeling. These methods assume that a model trained on normal data should be able to accurately predict future observations. If the prediction error becomes large, the observation may be considered anomalous.

Forecasting-based methods use past observations to predict future values and identify anomalies when the prediction error exceeds a threshold. Classical models include exponential smoothing and ARIMA (Rousseeuw 1984), while deep learning models such as Long Short-Term Memory (LSTM) networks (Malhotra et al., 2015; Chauhan and Vig, 2015), Gated Recurrent Units (GRU) (Wu et al., 2020; Zhao et al., 2020) and Echo State Networks (Chang et al., 2009) have been widely applied because of their ability to capture complex temporal dependencies.

Another class of approaches is reconstruction-based methods, which attempt to reconstruct the original time series from a compressed latent representation (Boniol et al., 2024c). These models are trained using only normal data, so anomalous patterns tend to produce large reconstruction errors. Popular methods include autoencoders and generative adversarial networks (GANs) (Li et al., 2019; Niu et al., 2020).

However, both prediction-based and reconstruction-based approaches face an important challenge. If anomalous patterns appear frequently during training, the model may inadvertently learn them as normal behavior. Conversely, unusual but valid patterns may be incorrectly identified as anomalies. To address this issue, some studies have proposed decomposing the time series into multiple components before modeling. One example is a framework based on spectral decomposition, in which the time series is treated as a signal and decomposed using Fourier transforms into trend, seasonal, and residual components (Lei et al, 2023). Deep learning models are then used to predict the trend and residual components separately, and the original signal is reconstructed by combining all components. Anomalies are detected by measuring the difference between the reconstructed signal and the original observation.

3.2.4 Evaluation of Anomaly Detection Methods

Evaluating anomaly detection models typically involves classification-based metrics. Threshold-based evaluation converts anomaly scores into binary decisions by using a predefined threshold and measures performance by using metrics such as precision, recall, false positive rate, and F-score (Boniol et al., 2024c). However, threshold-based metrics can be sensitive to extreme values and may produce unfair comparisons between models. Therefore, threshold-independent evaluation metrics, such as area-under-curve measures, are often used to provide more robust assessments of model performance.

3.3 LEARNING METHODS FOR TIME-SERIES DATA

3.3.1 Deep Learning for Time-Series Modeling

Deep learning has become an important tool for time-series analysis because of its ability to model complex temporal patterns. In supervised learning settings, two common tasks are time-series classification (TSC) and time-series regression (TSER) (Foumani et al, 2024). TSC aims to assign categorical labels to time-series sequences, typically using a softmax output layer and cross-entropy loss, whereas TSER predicts continuous numerical values using regression losses such as mean squared error or mean absolute percentage error.

Several neural network architectures have been applied to time-series tasks. Multi-layer perceptrons (MLPs) are simple feed-forward networks that operate on fixed-length feature vectors but do not explicitly model temporal relationships (Arias del Campo et al., 2021). Convolutional neural networks (CNNs) can capture local temporal patterns by applying convolutional filters across time-steps (Zheng et al., 2014). Recurrent neural networks (RNNs), including LSTM and GRU architectures, are specifically designed to model sequential dependencies through internal memory mechanisms. Hybrid models that combine CNN and RNN components can capture both spatial patterns and temporal dependencies. More recently, attention-based models such as transformers have been introduced to capture long-range dependencies in time-series data (Zerveas et al., 2021; Chowdhury et al., 2022). Additionally, graph neural networks (GNNs) can model spatial relationships among sensors or network components in transportation systems.

3.3.2 Sequence Classification for Time-Series Data

Time-series analysis can also be formulated as a sequence classification problem, in which a sequence of observations is mapped to a categorical label (Xing et al, 2010). A sequence can be defined as an ordered list of events, and the goal of sequence classification is to learn a function that maps a sequence to a corresponding class label.

Several approaches have been proposed for sequence classification. Feature-based methods transform sequences into feature vectors by using feature extraction or selection techniques (Leslie et al., 2002; Leslie and Kuang, 2004). Distance-based methods measure the similarity between sequences, often using distance metrics such as dynamic time warping (Saigo et al., 2004). Model-based approaches, such as Hidden Markov Models (HMMs), attempt to model the probabilistic structure of the sequence directly (Zhong, 2005). For time-series data, distance-based approaches are particularly common because numeric time-series data often make explicit feature enumeration difficult.

3.3.3 Self-Supervised Representation Learning

Because labeled anomaly data are often scarce, recent research has explored self-supervised learning (SSL) for time-series representation learning. SSL aims to learn useful feature representations from unlabeled data by designing pretext tasks derived directly from the data itself (Meng et al, 2023). Instead of relying on manually labeled examples, the model learns by solving auxiliary tasks that encourage it to capture underlying patterns in the data.

Several types of pretext tasks have been proposed. Adversarial learning trains models to distinguish between real and generated data (Yoon et al., 2019). Predictive learning requires the model to predict missing or future segments of the time series (Ma et al., 2019; Malhotra et al., 2017). Contrastive learning trains models to identify similarities and differences between samples by pulling related instances closer in the representation space and pushing unrelated instances apart (Huynh et al., 2022). Contrastive approaches can operate at different levels, including instance-level, prototype-level, and temporal-level representations (Chen et al., 2020; Dave et al., 2022; Franceschi et al., 2019).

4. SYSTEM ARCHITECTURE AND TECHNOLOGY OVERVIEW

4.1 IN/OUT COUNTING TECHNOLOGY

In/out counting sensors were installed at the entrance and exit points of a parking lot to monitor vehicle movements. The location of a sensor significantly affects measurement accuracy. Installation constraints may include proximity to electrical facilities to reduce installation costs or mounting sensors on existing poles to minimize infrastructure costs while ensuring that the equipment remains within the right-of-way. Figures 4-1 and 4-2 show placement of the sensors at the Maytown site.

Figure 4-1 Installation layout of in/out counting sensors at the entrance and exit of the Maytown site.

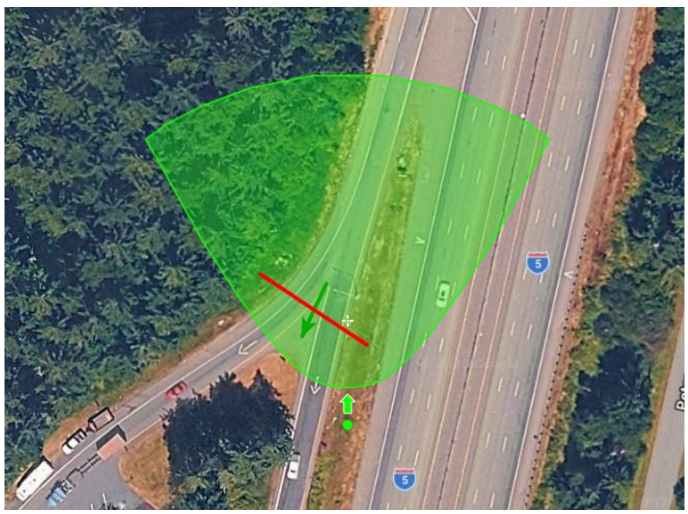


Figure 4-2 Example of the screenline configuration at the entrance of the Maytown site. The red line indicates the virtual screenline used by the sensor to detect vehicles crossing the entrance, while the shaded area represents the sensor's detection coverage.

The sensor operates on the screenline detection principle, where a virtual line is defined within the sensor's field of view. When a vehicle crosses this line, the sensor detects the event and measures key attributes such as vehicle length and speed (Figure 4-3). In addition, the sensor was equipped with an integrated camera to assist with vehicle classification. During the early deployment phase, the project team conducted field investigations to adjust the placement of the screenline, as its location would significantly affect detection accuracy. The optimal screenline position varies by site as a result of differences in installation angles, roadway geometry, and mounting locations. Therefore, each site requires calibration and adjustment to ensure reliable vehicle detection and classification.

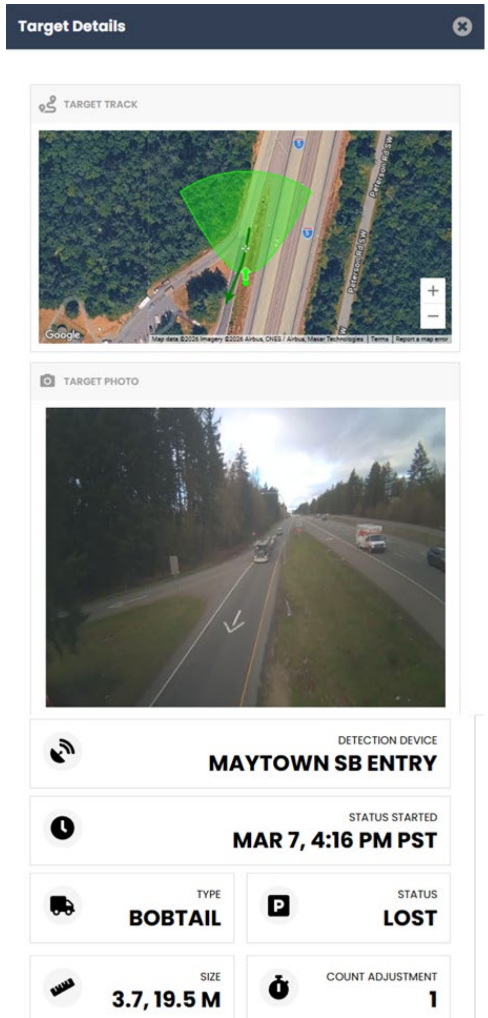


Figure 4-3 Example of detailed target detection information from the sensor, including vehicle track, camera image, classification, and detection attributes.

At some sites, truck parking areas and general parking areas share the same entrance and exit points (Figure 4-4). In such layouts, the system must rely on the sensor's vehicle classification capability to distinguish trucks from other vehicles. This ensures that only vehicles using the truck parking area—the primary focus of the monitoring system—are counted in the truck parking occupancy calculations.

Figure 4-4 Entrances/exits shared with passenger vehicles at Schrag WB

4.2 REFERENCE DATA

To evaluate and verify the accuracy of the in/out counting sensors, reference data were required as a ground-truth source. Traditionally, ground-truth verification is performed by manually reviewing closed-circuit television (CCTV) footage from the parking sites. This process allows researchers to visually confirm vehicle movements, parking occupancy, and counting accuracy. However, manual video review is labor-intensive and time-consuming, especially when data must be examined over long periods or across multiple sites.

To reduce the effort required for manual verification, some sites are equipped with a camera-based detection system that utilizes existing CCTV infrastructure. These systems apply computer vision techniques to automatically detect and count vehicles within the camera's field of view (Figure 4-5). The camera detection system can provide continuous reference data and help validate the performance of the radar-based in/out counting sensors.

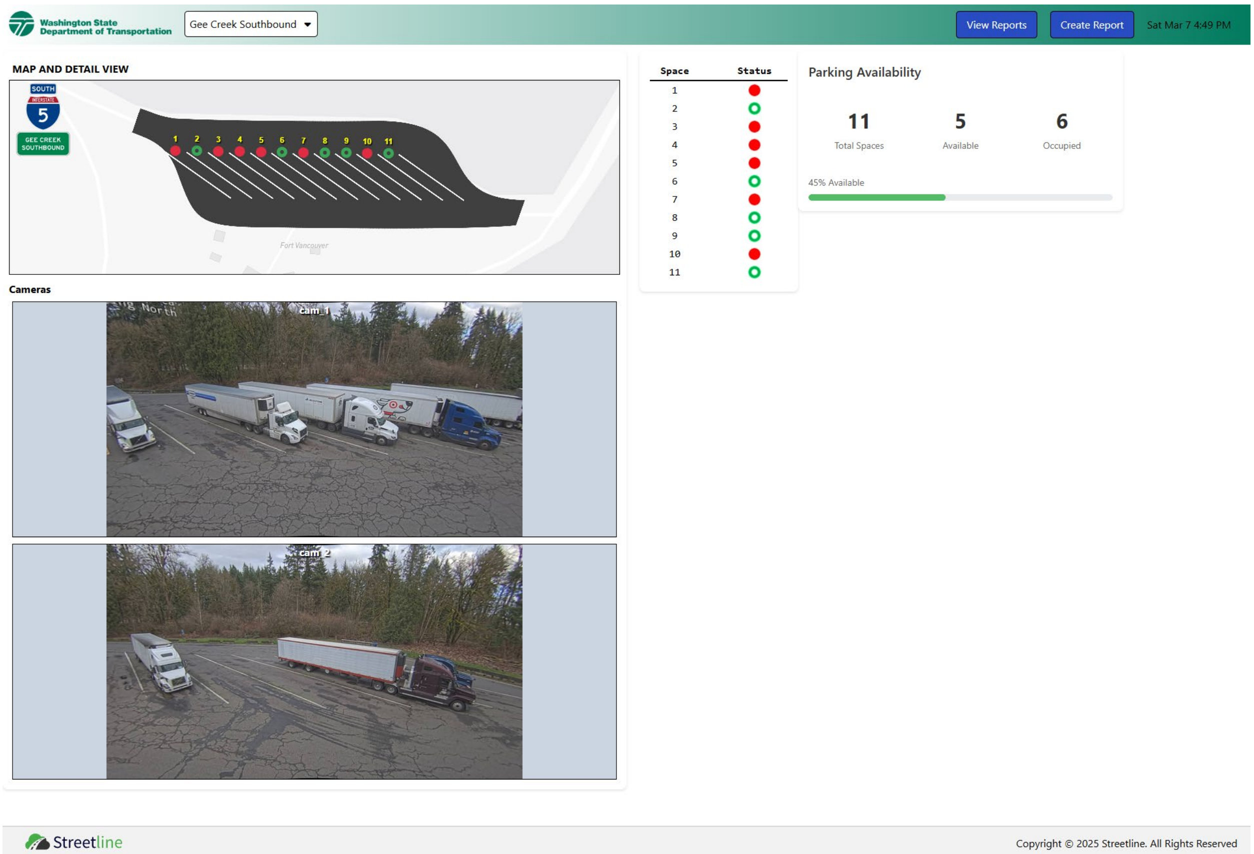


Figure 4-5 Camera-based detection dashboard showing parking space occupancy, availability, and live camera views.

Despite these advantages camera-based detection systems also have several limitations. First, environmental conditions can significantly affect camera performance. During severe weather events such as heavy rain, fog, or snow, visibility may be reduced, which can degrade detection accuracy. Lighting conditions, including nighttime operation, glare, or shadows, may also affect the quality of the captured images and the reliability of automated detection algorithms.

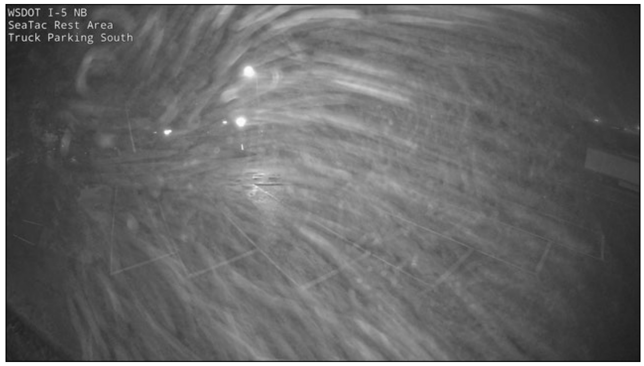


Figure 4-6 Example of camera visibility loss due to severe weather conditions at the SeaTac truck parking site (February 2, 2026, 00:30 AM).

Second, camera systems are limited by their field of view and potential occlusion. Vehicles may be partially or fully blocked by other vehicles, roadside structures, or vegetation, making it difficult for the system to detect them accurately. Trucks that park outside designated parking spaces (Figure 4-7), park too far from the camera's coverage area, or park in locations with limited visibility may not be captured by the camera system. These situations are common in crowded truck parking areas where drivers may not use designated spots when the facility is near capacity.

Figure 4-7 Example of trucks parking outside designated spaces at the SeaTac truck parking site (February 3, 2026, 00:00 AM).

Because both sensing technologies have strengths and weaknesses**, relying on a single sensor type may not provide reliable measurements under all conditions**. Radar-based sensors tend to be more robust in poor lighting and weather conditions, whereas camera systems provide valuable visual confirmation and spatial information about vehicle presence. Therefore, this study integrated information from multiple sensor sources to improve data reliability. By combining radar-based in/out counting data with camera-based observations, the system can better identify discrepancies and improve the overall accuracy of parking availability estimation and prediction.

4.3 OVERALL SYSTEM ARCHITECTURE

The overall system architecture of the Truck Parking Information Management System (TPIMS) integrates multiple sensing technologies, centralized data processing, prediction services, and user-facing applications to provide reliable truck parking availability information (Figure 4-8). The architecture was designed to support continuous data collection from field sensors, real-time processing, predictive analytics, and dissemination of parking information to various platforms.

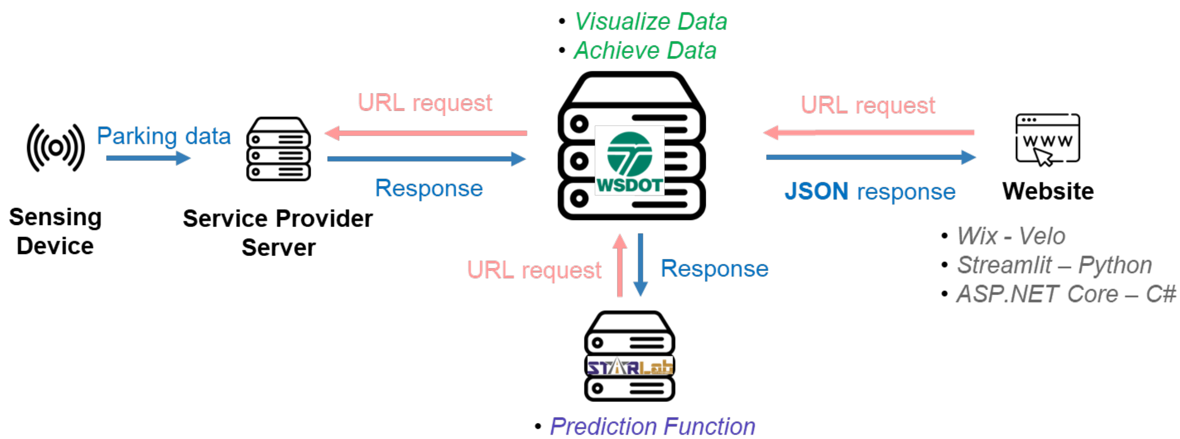


Figure 4-8 High-level system data exchange architecture between field sensors, WSDOT servers, UW prediction services, and public-facing applications.

At the sensing layer multiple detection systems are deployed at truck parking facilities. These include in/out counting sensors installed at the entrances and exits of parking lots and camera-based detection systems to monitor parking spaces directly. The in/out sensors are intended to detect vehicle movements across virtual screenlines and estimate parking occupancy by tracking arrivals and departures. Camera-based systems provide complementary information by directly observing parked vehicles and estimating occupancy within the camera's field of view. The combination of these sensing technologies improves robustness by compensating for the limitations of each individual sensor type.

Sensor data are transmitted to cloud services operated by the sensor vendors, where initial processing and formatting occur. The processed data are then retrieved by the WSDOT servers through scheduled data requests. Because different vendors provide data in different formats, the system converts the incoming data into a standardized data structure compliant with the Mid America Association of State Transportation Officials (MAASTO) truck parking data specification. This standardization enables consistent integration of data from multiple sensor vendors and simplifies downstream data processing.

Once the data are standardized the system performs additional processing steps, including occupancy estimation, anomaly checking, and prediction generation. The prediction module—developed as part of this study—uses historical and real-time parking data to estimate future parking availability over multiple time horizons. During the research and development phase, the prediction models were hosted on University of Washington (UW) servers to allow rapid experimentation, model updates, and performance monitoring.

The processed parking availability data and prediction outputs are exposed through a public TPIMS application programming interface (API), which has been developed and deployed to allow external systems and applications to retrieve the information in real time (Figure 4-9). These data are used by multiple user interfaces, including public websites and internal dashboards for system monitoring and analysis (Figure 4-10). Truck drivers can access real-time parking availability through mobile applications, while transportation agencies and researchers can use the data for operational monitoring and planning.

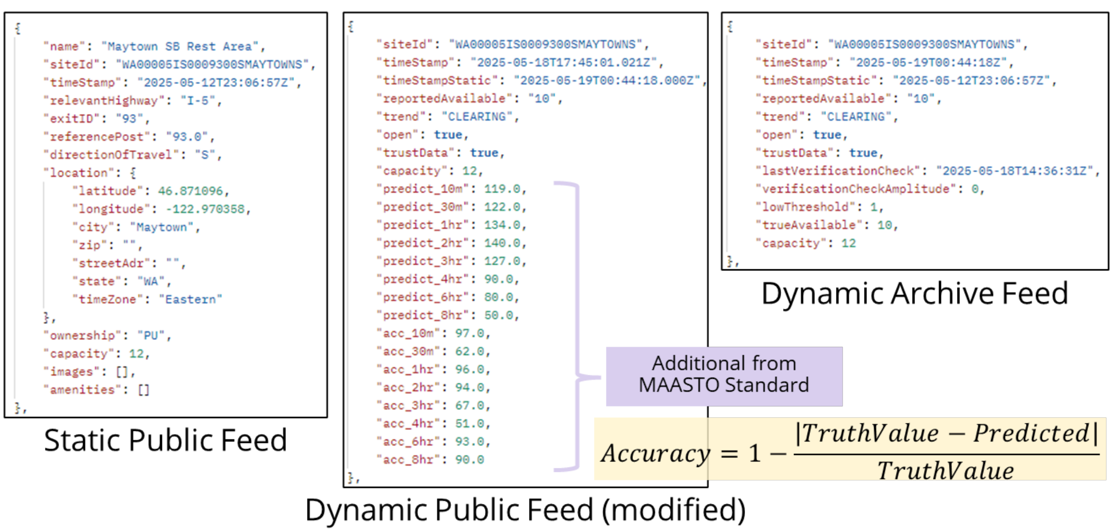


Figure 4-9 TPIMS API

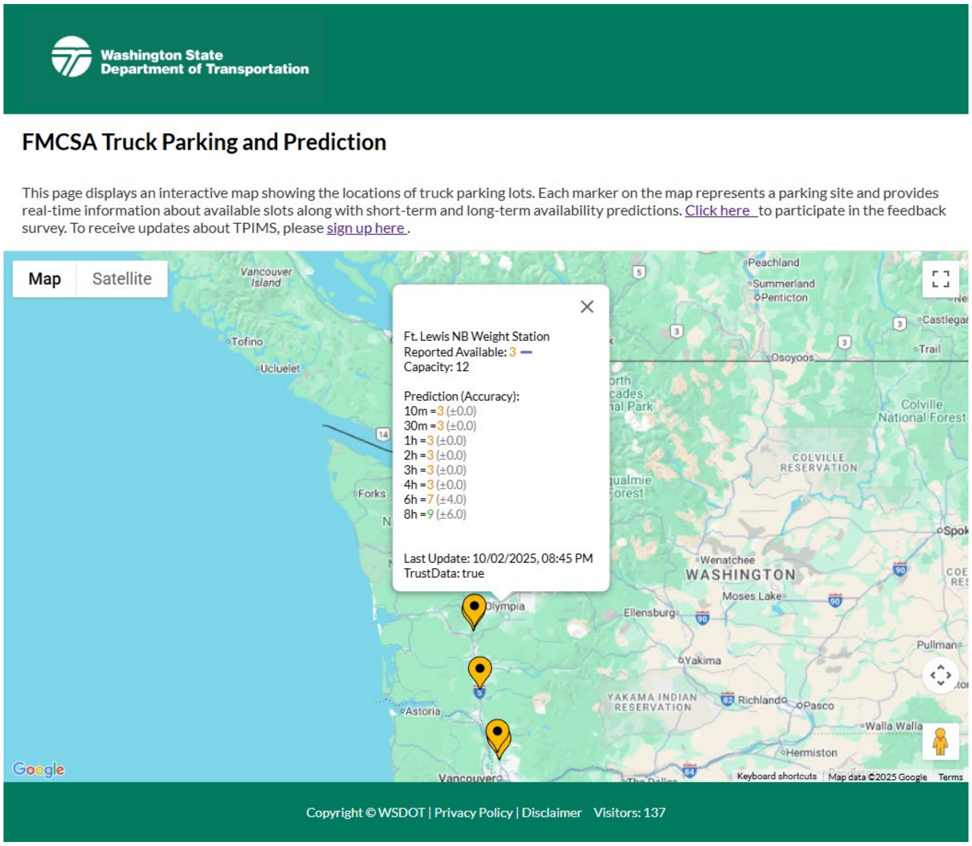


Figure 4-10 TPIMS web application

5. DATA COLLECTION AND EXPLORATORY DATA ANALYSIS

5.1 GROUND-TRUTH DATA COLLECTION

SeaTac was selected as the target site for developing and testing the calibration approach because its parking layout clearly separates truck parking entrances and exits from passenger vehicle access (Figure 5-1). This separation reduced the risk of misclassification by the radar-based in/out sensors. In addition, the SeaTac site was equipped with comprehensive CCTV coverage, which provided a reliable visual reference for verifying parking activity.

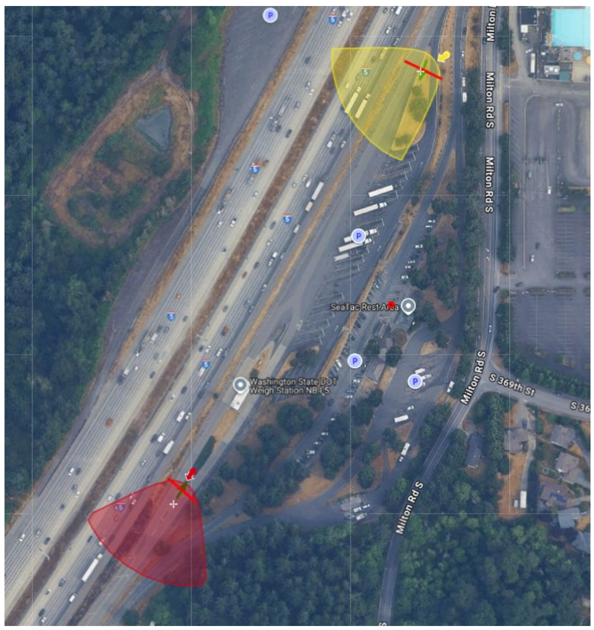


Figure 5-1 SeaTac truck parking layout showing separate entrance and exit points for trucks and passenger vehicles.

To obtain images for ground-truth data collection, the project team primarily relied on CCTV footage, which provided continuous visual confirmation of vehicle movements and parking occupancy. WSDOT supported this process by providing an SSH File Transfer Protocol (SFTP) site for data transfer. The system captured one-minute video segments from each camera and uploaded them to the SFTP location, where the files were stored temporarily and automatically deleted after 24 hours to manage storage capacity (Figure 5-2).

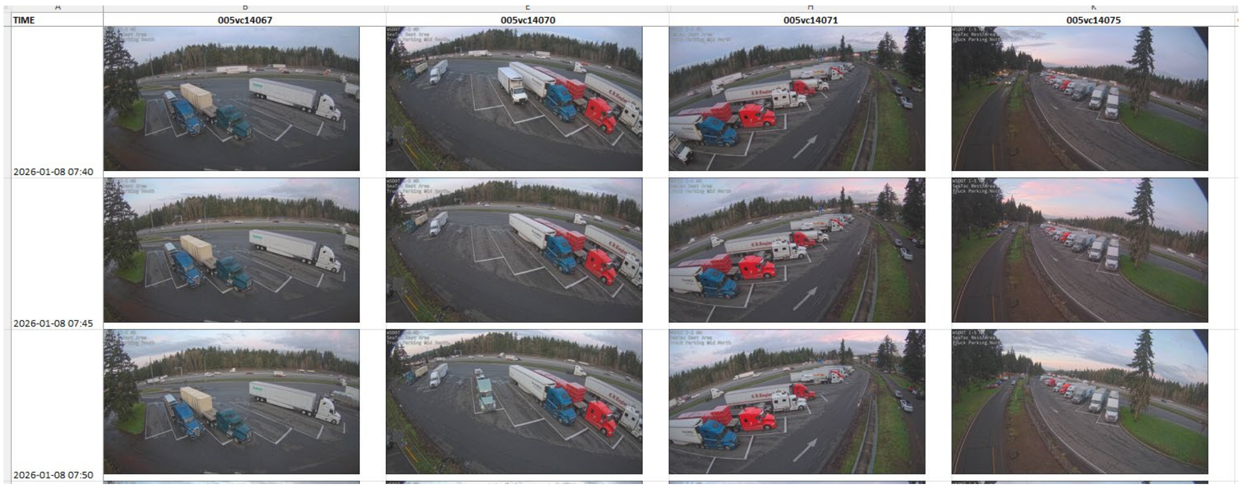


Figure 5-2 Example images retrieved from the SFTP site at the SeaTac location, showing synchronized views from four CCTV cameras covering all truck parking spaces.

For this study, nine data collection sessions were conducted at the SeaTac site. Each session covered observation intervals ranging from 5 to 15 minutes, with varying durations to capture different parking conditions and traffic patterns (Table 5-1).

Table 5-1 Summary of data collection sessions at the SeaTac site used for ground-truth verification and sensor calibration.

Date	Day	Time	Duration	Interval
1/7/2026	WED	20:50 – 23:05	2 HR	5 minutes
1/8/2026	THU	07:40 – 10:40	3 HR	5 minutes
1/25/2026 – 1//26/2026	SUN- MON	20:40 – 07:30 (+1)	11 HR	5 minutes
1/24/2026	SAT	00:15 – 20:50	20 HR	5 minutes
1/30/2026 – 1/31/2026	FRI	00:15 – 00:00 (+1)	24 HR	15 minutes
1/31/2026 – 2/1/2026	SAT	00:15 – 00:00 (+1)	24 HR	15 minutes
2/1/2026 – 2/2/2026	SUN	00:15 – 00:00 (+1)	24 HR	15 minutes
2/2/2026 – 2/3/2026	MON	00:15 – 00:00 (+1)	24 HR	15 minutes
2/3/2026 – 2/4/2026	TUE	00:15 – 00:00 (+1)	24 HR	15 minutes

Another source of ground-truth data was periodic verification conducted by the vendor's team. In general, sensors at each site were verified twice per day, typically once in the morning and once in the evening. During this process, the ground-truth checker reviewed the same CCTV footage and manually adjusted the reported occupancy values based on the observed parking conditions.

Unlike the previously described ground-truth data collection, which was conducted over shorter continuous observation periods, this verification process functioned more as a spot check. However, it provided coverage over a longer time span and helped identify systematic sensor errors or drift in reported parking availability.

5.2 EXPLORATORY DATA ANALYSIS

5.2.1 Initial Observations

An initial exploratory analysis was conducted by comparing the radar-based sensor outputs with the ground-truth values obtained from CCTV verification. This comparison helped identify patterns of agreement and discrepancy between the sensor-reported occupancy and the manually verified parking counts. A visual inspection of the time-series plots revealed two primary situations.

The first situation was slight drift, where the radar sensor measurements remained close to the ground-truth values over time (Figure 5-3). In these cases, the difference between the radar output and the verified occupancy was relatively small and stable. Although minor discrepancies may have occurred because of occasional miscounts or short-term detection errors, the overall trend of the radar signal generally followed the ground-truth pattern. This condition indicated that the sensor was functioning properly and that the counting logic was largely accurate. However, small adjustments or periodic recalibration may still be beneficial to improve accuracy and prevent errors from accumulating over time.

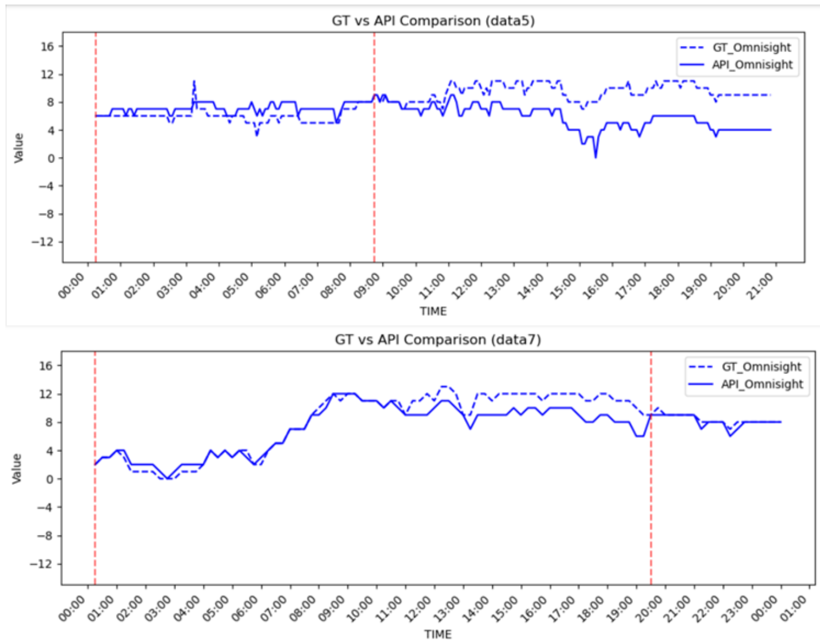


Figure 5-3 Comparison between ground-truth (GT) and radar sensor API occupancy values over time in slight drift case. Red dashed lines indicate the timestamps when manual verification was performed.

The second situation was significant drift, where the difference between the radar-based estimation and the ground-truth values gradually increased (Figure 5-4). In this case, counting errors accumulated as vehicles entered and exited the parking facility, resulting in a noticeable divergence between the two signals. Once such discrepancies had accumulated, the reported occupancy could deviate substantially from the actual parking condition. This type of drift indicated that sensor calibration or correction was required to restore the accuracy of the occupancy estimation.

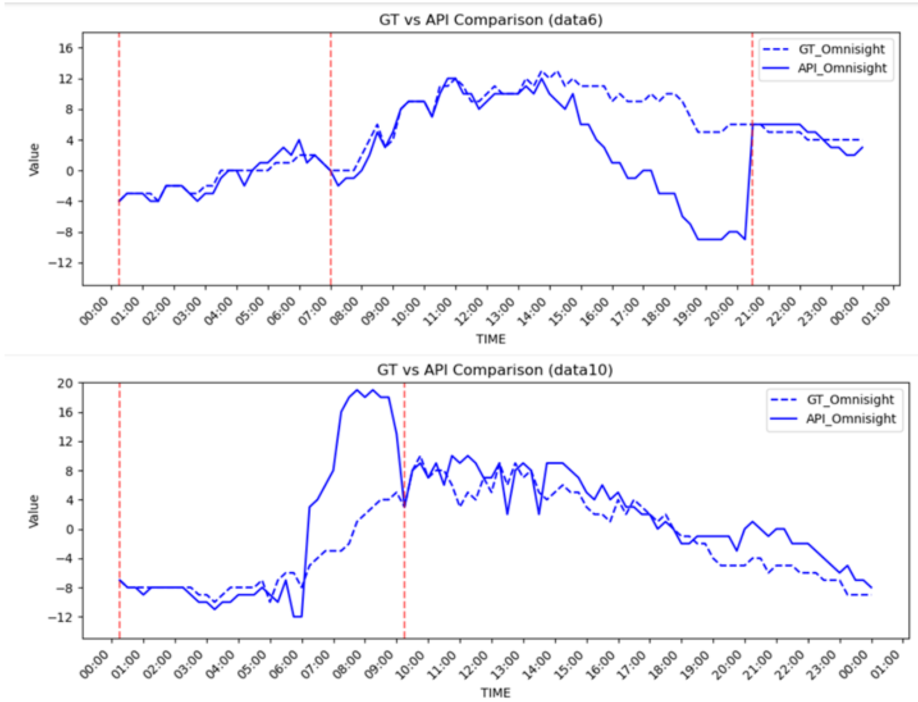


Figure 5-4 Example of significant drift between radar sensor API values and ground-truth counts over time in drift case. Red dashed lines indicate timestamps when manual verification was conducted.

With the availability of reference sensor data, we explored the opportunity to utilize camera-based observations to support the calibration process. Preliminary analysis suggested that when parking occupancy was high, the camera-based estimates tended to be closer to the ground-truth values (Figure 5-5). One possible explanation is that when a parking facility was near capacity most trucks were forced to park within designated spaces. As a result, fewer trucks were parked off-spot (i.e., outside designated parking spaces) or outside the camera's field of view, allowing the camera detection system to more accurately capture the actual parking occupancy.

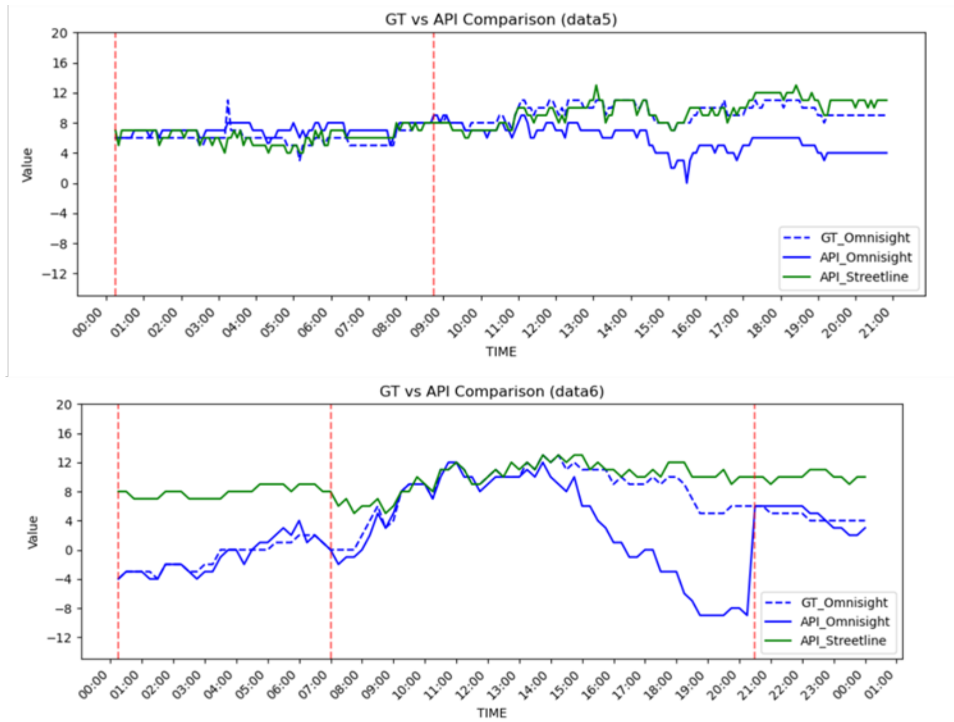


Figure 5-5 Comparison of ground-truth (GT), radar sensor API values, and camera-based detection (reference sensor) over time. Red dashed lines indicate manual verification timestamps used for calibration.

5.2.2 Temporal Patterns

This study adopted a detection-based approach for identifying anomalies in parking sensor data to learn the normal temporal patterns of parking occupancy from historical data and then compare new incoming data against these learned patterns. If the new observations deviated significantly from the expected trend, the system flagged the situation as a potential anomaly.

Truck parking usage often follows consistent daily and weekly temporal patterns driven by freight operations and driver schedules. For example, parking demand typically increases during late afternoon and evening hours when truck drivers stop to comply with hours-of-service regulations. Conversely, parking occupancy tends to decrease during morning hours as drivers resume their trips. Because of these recurring patterns, historical data can be used to establish baseline expectations for parking occupancy at different times of the day.

Figure 5-6 illustrates the temporal patterns of parking availability for several representative days of the week. The average curves show the typical occupancy pattern throughout the day, while the individual daily curves highlight the natural variability that occurs from day to day. Although the exact values vary, the overall trend remains relatively consistent, suggesting that temporal patterns can serve as a reliable reference for identifying abnormal sensor behavior.

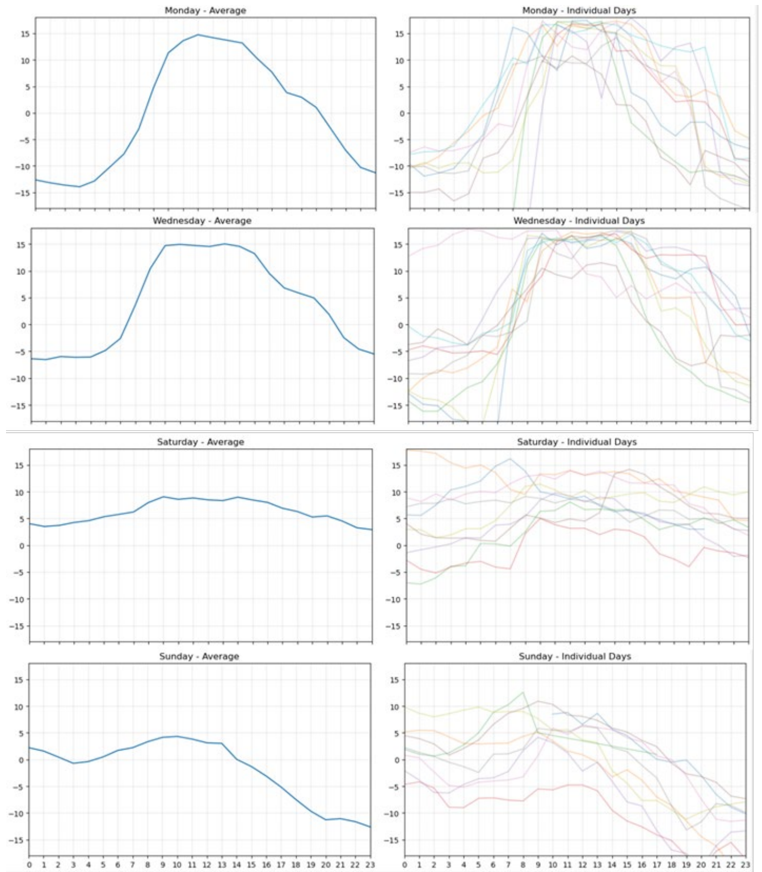


Figure 5-6 Temporal parking availability patterns by day of the week at SeaTac (Dec 1, 2025 - Feb 5, 2026)

5.3 SENSOR DISCREPANCIES

Most of the sensors currently deployed in the system rely on in/out counting technology, as this approach is relatively cost efficient and scalable for large-scale deployments across multiple truck parking sites. By detecting vehicles entering and exiting a parking facility, the system estimates the current occupancy based on the cumulative difference between entry and exit counts. However, this method also has several limitations. A key concern is that counting errors can accumulate over time. Because occupancy is estimated by using cumulative counts, even a small detection error, such as a missed entry or exit event, can gradually propagate through the system and lead to significant discrepancies between the estimated and actual number of parked vehicles.

One common source of error occurs when vehicle detections are missed because of occlusion or sensor blind spots. For example, when traffic becomes congested near the entrance or exit, multiple vehicles may move closely together, making it difficult for the sensor to clearly distinguish between individual vehicles crossing the detection zone (see Figure 5-7). As a result, some entry or exit events may not be properly detected, leading to inaccurate counts.

Figure 5-7 Congestion at the parking facility entrance, where closely spaced vehicles may reduce the accuracy of in/out sensor detections.

Another situation that may introduce discrepancies is irregular parking behavior. For instance, a truck may park in a way that partially occupies multiple spaces, such as when the rear portion of a trailer extends into an adjacent parking spot, or when a trailer is detached and left occupying a parking space (Figure 5-8). In such cases, the counting logic may interpret the space as available even though it is not fully usable for other trucks. This type of parking behavior can cause a mismatch between the estimated occupancy and the actual parking conditions observed on site.

Figure 5-8 Example of a trailer occupying multiple parking spaces, which can lead to discrepancies between sensor-based counts and actual parking availability.

6. ANOMALY DETECTION AND CALIBRATION FRAMEWORK

6.1 OVERALL FRAMEWORK

The framework aims to automatically detect anomalies in sensor-reported parking data and to provide calibration recommendations to improve the reliability of occupancy estimates (Figure 6-1). The framework evaluates incoming sensor data at each time-step and determines whether the reported values follow normal operational patterns or indicate potential sensor discrepancies.

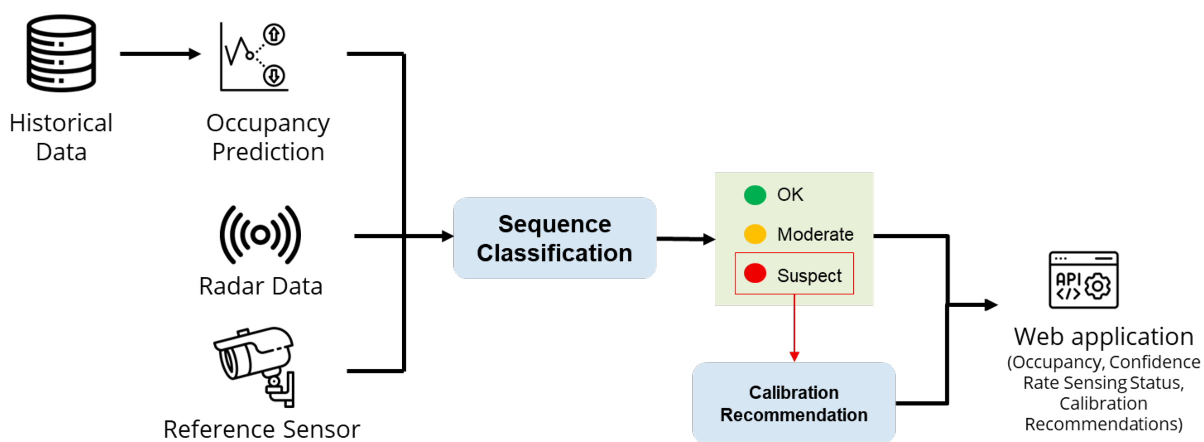


Figure 6-1 Overview of the anomaly detection and calibration framework.

First, the system analyzes the current observation and compares it with historical temporal patterns. A prediction-based approach is used to estimate the expected parking availability based on previously observed data. If the current sensor output deviates significantly from the predicted trend, then the observation has a potential anomaly.

Next, the framework incorporates reference sensor data and camera-based observations when available. By comparing radar-based in/out sensor estimates with reference data, the system can further identify inconsistencies and confirm whether the detected deviation is likely caused by sensor errors rather than actual changes in parking occupancy.

The outputs from these comparison processes are then organized into a sequence classification stage, where the system evaluates a sequence of observations to determine the overall sensor status. Instead of evaluating each data point independently, the sequence-based analysis considered temporal consistency and helps distinguish between temporary fluctuations and persistent sensor drift.

Finally, a post-processing step is applied when the system identifies situations where the sensor output is likely incorrect. In these cases, the framework generates recommended corrections or adjustments to the reported values to improve the overall accuracy of parking availability estimates. This step helps mitigate accumulated counting errors and ensures that the system continues to provide reliable information for downstream applications and users.

6.2 ANOMALY DETECTION AND SENSOR STATUS

To detect potential sensor discrepancies, this study developed a prediction model that learns the normal temporal behavior of the parking occupancy signal. The prediction model was trained by using historical data from a two-month period, allowing the system to learn typical parking patterns under different traffic, weather, and temporal conditions.

At each time-step, the model predicted the expected parking occupancy value. The difference between the predicted value and the observed radar sensor value (prediction residual) was then computed. Large residuals indicated that the current sensor reading deviated from the expected pattern and might represent a potential anomaly. In addition to the prediction residual, information from the reference sensor (camera-based detection) was incorporated to provide additional context for evaluating the reliability of the radar signal. By combining these signals, the system can better distinguish between true occupancy changes and sensor-related discrepancies.

Rather than evaluating each observation independently, the framework used a sequence classification approach to analyze a short sequence of recent observations. This allows the system to detect persistent patterns of deviation that indicate sensor drift or counting errors. Based on this analysis, the system assigned a sensor status label (e.g., OK, Moderate, or Suspect), which could then trigger further calibration or correction procedures.

6.2.1 Anomaly by Prediction-Based Method

Feature Selection

Several variables were considered as potential inputs for the prediction model. These included the following:

- **Radar signal** – the primary target variable representing the sensor-reported parking availability.
- **Reference sensor** – camera-based occupancy estimates.

- **Weather data** – retrieved from the OpenWeatherMap API. Because weather data include several rare categories, some uncommon weather conditions, such as Snow and Squall, were grouped into a single category labeled "Other" to reduce data sparsity.
- **Traffic data** – obtained from the WSDOT Traveler Information API.
- **Temporal variables** – including minute, hour, and day of week when the sensor data were recorded.
- **Elapsed time since last verification** – representing the time since the sensor was manually verified.

To determine the most informative feature combination, several feature sets were evaluated using a fixed model configuration. An *InceptionTime* model was used with a sequence length of 72 time-steps, which corresponded to approximately six hours of historical observations. The results of these experiments are presented in Table 6-1. Among the tested configurations, Feature Set 3 achieved the lowest test error, with a mean absolute error (MAE) of 0.6272, indicating the best predictive performance. Therefore, Feature Set #3 was selected as the input feature set for the subsequent modeling and anomaly detection steps

Table 6-1 Prediction performance of different feature combinations evaluated using the InceptionTime model (sequence length = 72 time-steps)

Feature Set	Radar	Reference	Weather	Traffic	Temporal	Elapse Time	Data	MAE	RSME	MAPE
#1	✓	✓	✓	✓	✓	✓	train	0.5719	0.8252	11.61
#1	✓	✓	✓	✓	✓	✓	val	0.704	0.9674	18.88
#1	✓	✓	✓	✓	✓	✓	**test**	**0.6731**	**0.9495**	**16.44**
#2	✓	No	✓	✓	✓	No	train	0.5939	0.8633	12.3
#2	✓	No	✓	✓	✓	No	val	0.6652	0.934	17.9
#2	✓	No	✓	✓	✓	No	**test**	**0.6342**	**0.9184**	**15.49**
#3	✓	✓	✓	✓	✓	No	train	0.5463	0.8255	11.46
#3	✓	✓	✓	✓	✓	No	val	0.6654	0.9451	18.27
#3	✓	✓	✓	✓	✓	No	**test**	**0.6272**	**0.9057**	**15.67**
#4	✓	No	✓	✓	✓	✓	train	0.5729	0.8576	12.06
#4	✓	No	✓	✓	✓	✓	val	0.658	0.938	17.47
#4	✓	No	✓	✓	✓	✓	**test**	**0.642**	**0.922**	**16.2**

Baseline Model Selection

After the feature set had been selected, the next step was to determine the most appropriate model architecture and sequence length. In this study, a time-step refers to a five-minute interval in the data, representing the temporal resolution at which parking availability and related features are recorded and modeled. Initially, the model was trained to predict one time-step ahead (five minutes). However, preliminary experiments showed that when only one time-step ahead was predicted, many models tended to simply repeat the previous observation rather than learning the underlying temporal pattern. The prediction horizon was extended to six time-steps ahead (30 minutes). This longer prediction window encouraged the model to rely not only on the most recent observation but also on the temporal patterns learned from the historical training data.

To address this issue, multiple sequence lengths were tested, including the following:

- Twelve time-steps (1 hour)
- 24 time-steps (2 hours)
- 48 time-steps (4 hours)
- 72 time-steps (6 hours)

Table 6-2 summarizes the performance of four candidate models: InceptionTime, LSTM, Temporal Signal Transformer (TST), and MiniRocket. Across the tested configurations, LSTM demonstrated the most stable performance across different sequence lengths and maintained relatively low prediction errors in comparison to other models. While some models achieved lower training errors, they exhibited higher validation and test errors, suggesting potential overfitting.

The sequence length of 48 time-steps (approximately four hours of historical observations) was selected as a reasonable balance between capturing sufficient temporal context and maintaining stable model performance. At this configuration, the LSTM model achieved a test MAE of 0.8921, with moderate root mean square error (RMSE) and mean absolute percentage error (MAPE) values, while maintaining consistent training and validation performance. Therefore, the LSTM model with a sequence length of 48 time-steps was selected as the baseline prediction model for the anomaly detection framework.

Table 6-2 Performance comparison of different prediction models and sequence lengths (12, 24, 48, and 72 time-steps)

Model	**Data**	**Seq=12 MAE**	**Seq=12 RSME**	**Seq=12 MAPE**	**Seq=24 MAE**	**Seq=24 RSME**	**Seq=24 MAPE**	**Seq=48 MAE**	**Seq=48 RSME**	**Seq=48 MAPE**	**Seq=72 MAE**	**Seq=72 RSME**	**Seq=72 MAPE**
Inception time	train	0.6385	0.8780	14.4100	0.5493	0.8022	12.6300	0.5826	0.7920	14.6900	0.6718	0.8763	17.8700
Inception time	val	0.7984	1.1798	16.6000	0.7216	1.0081	18.7800	0.8565	1.1819	24.3800	1.3370	1.8495	28.7400
Inception time	test	8.6304	10.6469	16.6000	0.6504	1.0115	15.2000	0.9504	1.3802	20.3500	1.2136	1.7592	30.2300
LSTM	train	0.4270	0.6220	9.6600	0.4560	0.6683	10.3400	0.4235	0.6054	10.7300	0.4566	0.6469	12.1200
LSTM	val	0.8139	1.2167	18.9100	0.7581	1.0677	19.7200	0.8278	1.1919	23.2300	1.0676	1.6840	25.6700
LSTM	test	0.8147	1.2007	16.9300	0.7252	1.0707	16.6800	**0.8921**	**1.3303**	**18.5600**	0.8439	1.2128	24.5000
TST	train	0.8317	1.0749	18.1600	0.8517	0.7905	13.9700	0.5489	0.7181	13.9400	0.5128	0.6733	13.6700
TST	val	0.9587	1.4042	19.8500	0.8850	1.1759	22.8000	1.3137	1.7456	35.8600	2.0776	2.6649	40.2100
TST	test	0.8340	1.1055	20.4300	0.7823	1.1325	17.9600	1.3905	1.9512	27.9000	1.8621	2.4102	50.3200
Mini rocket	train	0.6270	0.8370	14.4700	0.3900	0.5150	9.1900	0.3080	0.4110	8.0000	0.3240	0.4300	9.0000
Mini rocket	val	1.0980	1.6880	23.4300	1.4730	1.8710	36.2700	1.6240	2.1380	48.7300	2.5260	3.3750	45.9800
Mini rocket	test	0.9530	1.4190	23.6500	1.5410	2.4470	34.9500	1.9010	2.4760	37.0400	2.5210	4.7300	57.2800

6.2.2 Sensor Status by Sequence Classification

Rule-Based Label Generation

To train the sequence classification model, we first created labels for each observation based on the discrepancy between sensor signals. Specifically, we examined the residual between the predicted value and the radar signal, along with the difference between the radar sensor and the reference camera sensor. These discrepancies helped determine whether the radar sensor was operating normally or if potential counting errors were occurring.

To determine appropriate thresholds, the distribution of the prediction residual was analyzed by using historical baseline data (Figure 6-2). The distribution showed that most residual values were concentrated near zero, indicating that the prediction model generally followed the radar signal closely under normal conditions. Exploratory analysis of the residual distribution indicated that a difference of approximately three spaces represented a reasonable threshold for determining whether the prediction and radar signal were consistent. Based on this observation, a rule-based labeling scheme was defined to categorize sensor status into three levels:

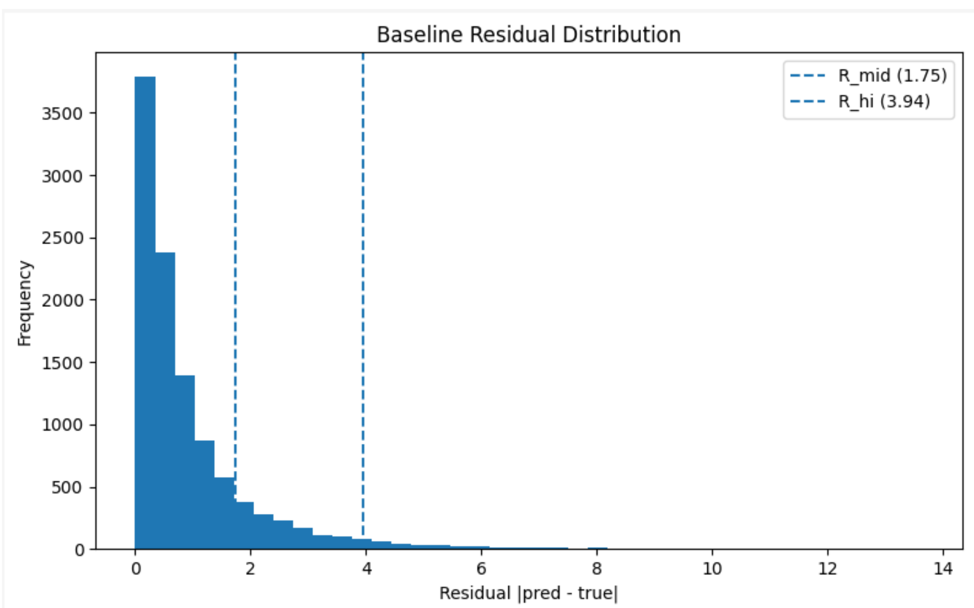


Figure 6-2 Distribution of baseline prediction residuals on the test dataset

- 🟢 **OK:** $|\text{Radar} - \text{Reference}| \leqslant 2$. The radar signal closely matches the reference sensor, and the prediction model supports the observed value. This indicates that the sensor is operating normally.

- 🟡 **Moderate:** $2 < |\text{Radar} - \text{Reference}| < 8$. There is a noticeable difference between the radar and reference sensors, suggesting potential minor counting discrepancies.
- 🔴 **Suspect:** $|\text{Radar} - \text{Reference}| \geqslant 8$. A large discrepancy exists between the radar and reference sensors. This condition likely indicates sensor malfunction or accumulated counting errors.

These rules allowed the system to generate training labels automatically, enabling the development of a supervised classification model for sensor status detection.

Model Selection

After the labeled dataset was generated, a sequence classification model was trained to automatically detect sensor status. The *InceptionTime* architecture was selected for this task because of its strong performance in time-series classification problems. The model used the same temporal configuration as the prediction model, with an input sequence length of 48 time-steps (approximately four hours) and a prediction horizon of 30 minutes ahead.

Using this configuration, the model achieved an overall classification accuracy of 74.5 percent on the test dataset. Figure 6-3 illustrates the predicted class probabilities for each true class, while Figure 6-4 shows the normalized confusion matrix of the classification results. The model demonstrated a strong performance in identifying the Suspect condition, with 85 percent correct classification, while most misclassifications occurred between the OK and Moderate categories because of their similar signal patterns.

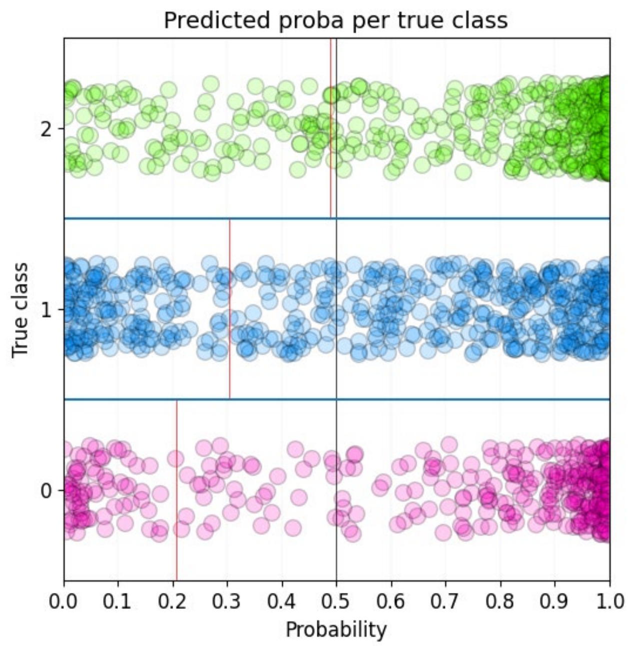


Figure 6-3 Distribution of predicted class probabilities for each true sensor status class

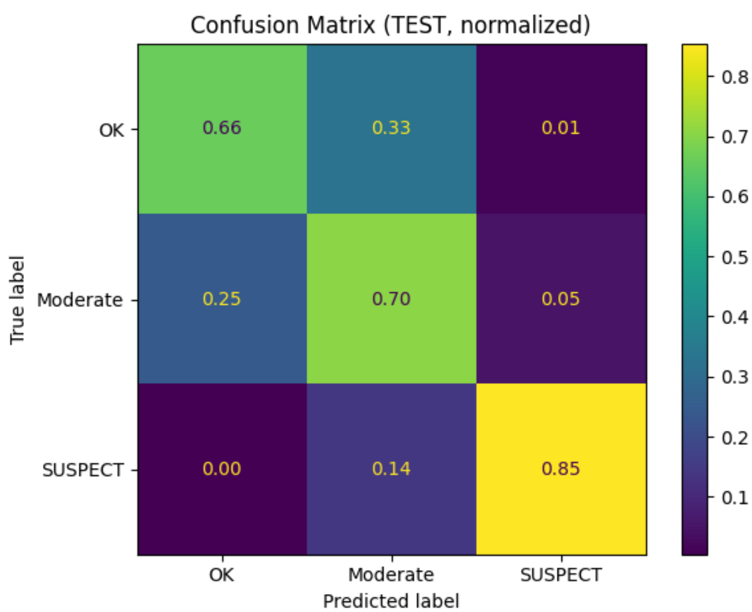


Figure 6-4 Normalized confusion matrix of the sequence classification model on the test dataset.

Confidence Score

In addition to predicting the sensor status class, the system also estimated a confidence score, which represented the probability that the predicted class label was correct. This confidence score provided an additional measure of reliability for each classification result and helped distinguish between highly certain predictions and those that might require further verification. The confidence estimation integrated multiple signals generated during the anomaly detection process, allowing the system to account for both model uncertainty and discrepancies between sensor measurements.

Specifically, three types of information were incorporated when the confidence score was computed. First, the classifier probabilities for each predicted class were included, reflecting the model's internal confidence in assigning a particular label (OK, Moderate, or Suspect). Second, the residual between the predicted occupancy value and the observed radar signal was used, since larger prediction errors could indicate that the sensor reading deviated from the expected temporal pattern. Third, the difference between the radar sensor and the reference camera-based sensor was considered, as disagreement between two independent sensing systems could provide an additional indication of potential sensor errors.

The confidence score was estimated by using a logistic regression model, which mapped these input signals to the probability that the predicted classification was correct. The probability was calculated by using the standard logistic function

$$P(Correct) = \frac{1}{1 + e^{-z}}$$

where z is a linear combination of the input features, including the predicted class probabilities, the prediction residual, and the difference between the two sensors.

$z = \beta_0 + \beta_1$(pred_proba_class0)+β_1(pred_proba_class1)+β_1(pred_proba_class2)+β_1(residual prediction)+β_1(diff two sensors)

The model was trained by using approximately 600 labeled ground-truth observations, which were obtained from manual verification of parking occupancy. Using this dataset, the confidence model achieved an Area Under the ROC Curve (AUC) of 0.664, an Average Precision (AP) of 0.798, a Brier score of 0.240, and a log loss of 0.685, indicating moderate discrimination ability and reasonable probability of estimation performance.

To further improve the reliability of the predicted probabilities, the classifier outputs were subsequently calibrated using isotonic. Probability calibration helped align the predicted confidence scores with the true likelihood of correct predictions. After applying isotonic calibration, the Brier score improved to 0.227, indicating better agreement between predicted probabilities and observed outcomes. This calibrated confidence score was then used to assess the reliability of the classification results and guide subsequent calibration recommendations for the sensor system.

The resulting confidence score could be interpreted as in Table 6-3. This confidence score provides an additional layer of information for system operators. Even when the model classifies a sensor as OK, a low confidence score may indicate uncertainty and prompt further investigation.

Table 6-3. Interpretation of confidence score values used to assess the reliability of predicted sensor status classifications.

Confidence	**Meaning**
> 0.85	Very reliable prediction
0.7–0.85	Reliable
0.5–0.7	Moderate confidence
< 0.5	Unreliable prediction

6.3 CALIBRATION APPROACH

After detecting anomalies in the sensor data, the framework proceeded to a calibration step, in which the system recommended corrections when the likely source of error was clearly identified. The goal of this step is to automatically reduce sensor discrepancies and improve the accuracy of the reported parking occupancy.

The calibration procedure focused on situations in which there was strong evidence that the radar sensor was producing inaccurate measurements. Specifically, automatic correction was applied when the following conditions were satisfied:

- The sensor status was classified as “Suspect.”
- The radar signal exceeded the reference sensor value.

When both conditions occurred simultaneously, this suggested that the radar sensor was overcounting vehicles, which is a common issue for in/out counting systems. This situation often arises from accumulated entry–exit counting errors, where missed detections or double counts gradually increase the estimated occupancy beyond the actual value. In these cases, the system recommended using the reference sensor measurement to correct the radar estimate, as the reference sensor (camera-based detection) was less affected by cumulative counting errors and provided a more reliable snapshot of the actual parking condition.

Figure 6-5 illustrates this calibration process. The blue points represent the corrected occupancy values, where the radar measurements were adjusted using the reference sensor data when the calibration conditions were triggered. As a result of this correction, the overall prediction error was reduced. In the example shown, the MAE decreases from 5.51 to 3.97, demonstrating the effectiveness of the automated calibration approach in improving sensor accuracy.

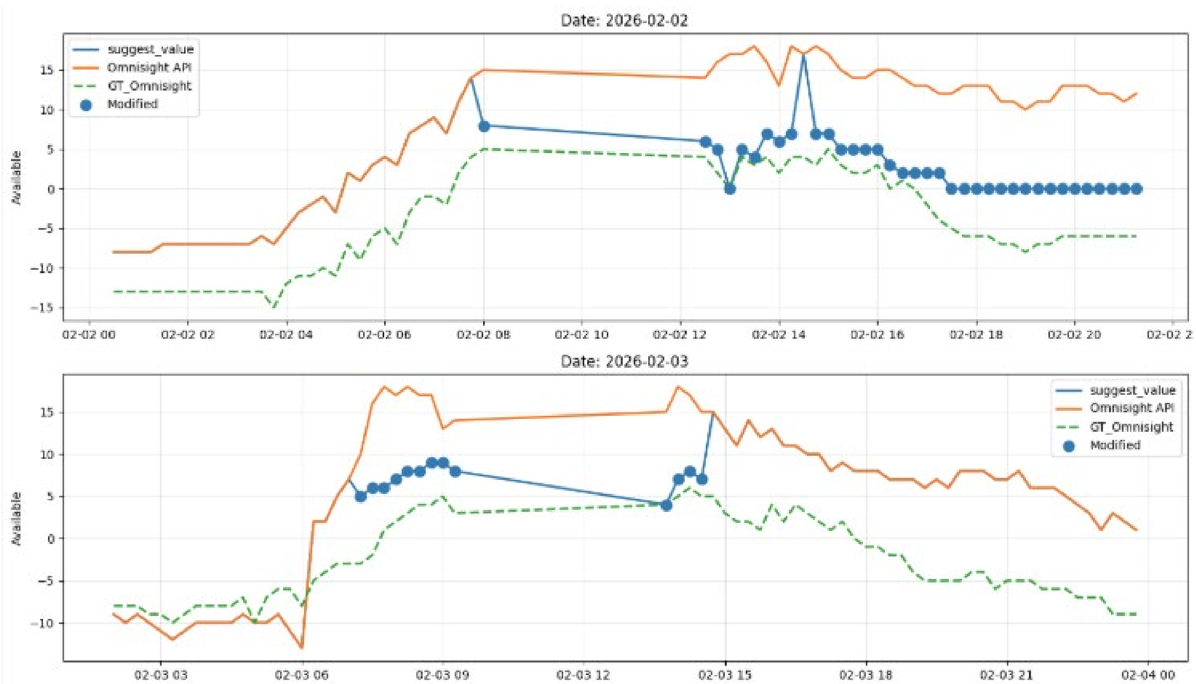


Figure 6-5 Example of the automatic calibration process for radar sensor data

7. SYSTEM IMPLEMENTATION

7.1 INTEGRATION WITH EXISTING TPIMS

During deployment, the framework was integrated into the existing Truck Parking Information Management System (TPIMS) architecture (Figure 7-1). The TPIMS system consists of four main functional components: sensing, real-time data processing and dissemination, prediction, and anomaly detection with calibration.

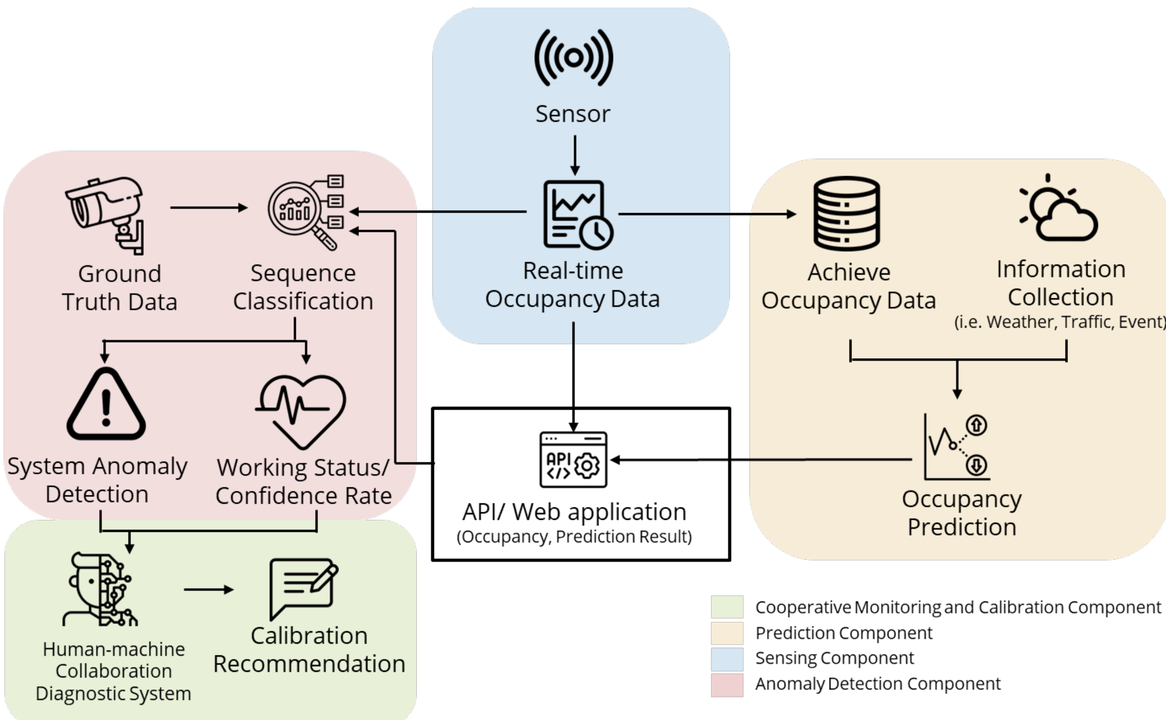


Figure 7-1 Integration of the anomaly detection and calibration framework within the existing TPIMS architecture.

The sensing component received data directly from field sensors deployed at truck parking facilities. These sensors continuously detect vehicle movements and generate real-time parking occupancy information. The incoming sensor data were processed to produce real-time occupancy estimates, which were then disseminated through system interfaces such as APIs and web applications. These real-time data streams served multiple purposes, including operational monitoring, prediction modeling, and anomaly detection.

The prediction module used historical parking occupancy data together with contextual information such as weather conditions and traffic data to forecast future parking availability.

These predictions were generated at regular intervals and were distributed through the same API infrastructure used for real-time occupancy data. This ensured that both current and predicted parking availability could be accessed consistently by downstream applications and users.

The newly introduced anomaly detection and calibration module extended the TPIMS architecture by incorporating additional ground-truth information, or reference sensor data. This module analyzed discrepancies between predicted values, radar sensor signals, and reference measurements to detect potential anomalies in the system. Using a sequence classification approach, the framework determined the current sensor status (e.g., OK, Moderate, or Suspect) and calculated a confidence score representing the reliability of the sensor output.

Once the sensor status has been determined, the system can generate calibration recommendations when strong evidence of sensor error is detected. These recommendations can be used either for automated correction or for supporting human operators in diagnosing sensor issues.

7.2 VISUALIZATION

To support monitoring and operational use of the framework, a visualization dashboard was developed by using a Streamlit-based web application. This dashboard provides an interface for operators to observe sensor status, evaluate anomaly detection results, and review calibration recommendations in real time. A dedicated Calibration Monitor page was created to present the outputs of the anomaly detection and calibration modules (Figure 7-2). The dashboard consists of two main sections.

The first section displays the sensor status over time, including the classification results produced by the sequence classification model. The visualization shows whether the sensor is classified as OK, Moderate, or Suspect at each time-step. In addition, the confidence score associated with each classification is displayed, providing a measure of how reliable the prediction is. The dashboard also includes the last manual verification timestamp, allowing operators to quickly identify when the sensor was most recently checked or recalibrated.

The second section presents the calibration recommendation signals. This component compares the radar sensor measurements, the reference sensor values, and the predicted occupancy trend. When the system detects conditions that indicate likely sensor errors, the dashboard highlights recommended corrected values based on the reference sensor. These recommendations are only displayed when the predefined calibration conditions are satisfied, ensuring that corrections are suggested only when sufficient evidence of sensor discrepancy exists.

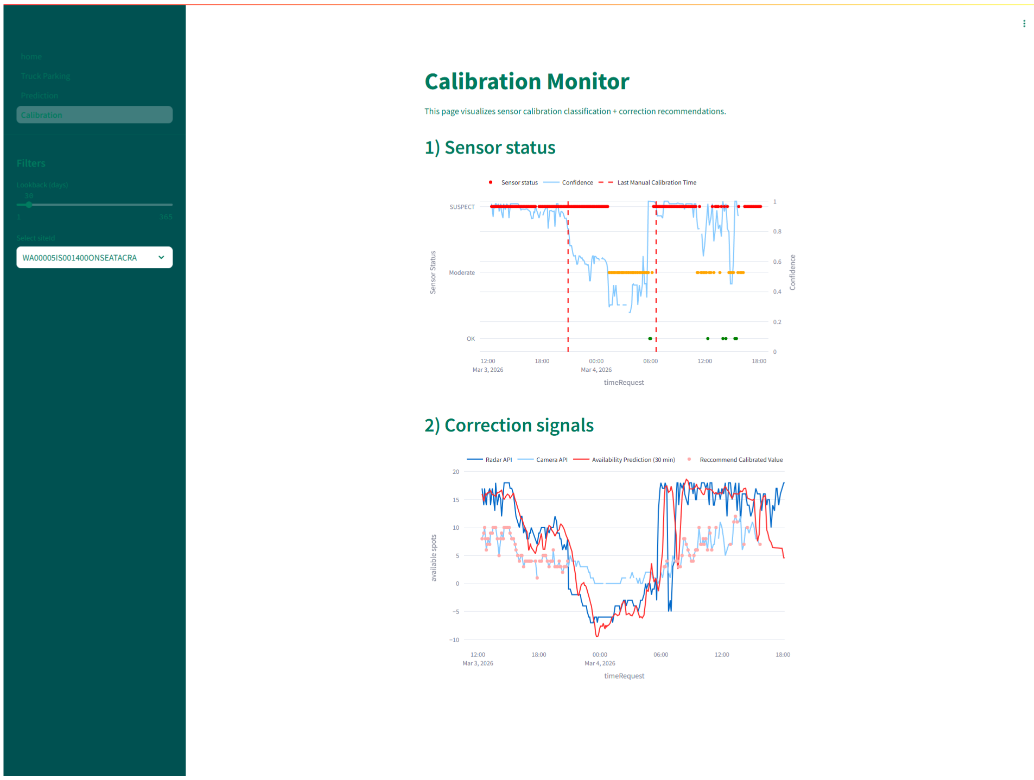


Figure 7-2 Calibration monitoring dashboard showing sensor status classification, confidence levels, and recommended correction signals for improving radar-based parking occupancy estimates

8. OUTREACH AND TECHNOLOGY TRANSFER

8.1 CODE AVAILABILITY

To support technology transfer and future implementation, the anomaly detection and calibration code developed in this project has been made available through the WSDOT TPIMS GitHub repository as a component of the broader TPIMS system. The code was publicly posted to facilitate review, reuse, and future extension by project partners and technical staff involved in deployment and maintenance. The repository is available at: https://github.com/nutvaraj/WSDOT_TPIMS/tree/main/AD

The released code implements a continuous inference pipeline for truck parking sensor calibration. The script was designed to run every five minutes and perform a full sequence of data retrieval, prediction, classification, confidence scoring, and database writing. Specifically, the pipeline loads trained regression and classification models; retrieves recent sensor, weather, and traffic data from the database; prepares time-series features; generates short-term occupancy predictions; evaluates sensor status; computes an optional confidence score; and applies rule-based logic to identify cases that may require correction before the final outputs are written to the calibration table.

The main inference script showed that the implementation is intended for near-real-time monitoring rather than offline analysis only. The script runs in a continuous loop and is scheduled to execute at 5-minute intervals (minutes 0, 5, 10, …, 55 of each hour). For each configured site, it loads the regression model, classification model, and optional confidence model; retrieves historical records from the *dynamic_archive* and *dynamic_public3* tables; engineers time-based and sensor comparison features; performs regression-based forecasting; then classified the latest sensor condition and writes results into a calibration table using an insert-or-update operation keyed by *siteId* and *timeRequest*.

The code also reflects the structure of the calibration framework. The regression portion predicts future parking availability using recent historical observations, while the classification portion uses recent sequential patterns to categorize the sensor condition. The implementation further includes confidence scoring and a rule-based auto-correction flag. In the current logic a correction is recommended when the predicted status is suspect, and the radar reading is higher than the reference signal under predefined conditions. The output includes not only the predicted class and class probabilities, but also supporting fields such as residual error, difference between sensors, confidence score, and recommended corrected value.

A configuration file was included to support practical deployment. The example configuration specifies the database connection, site identifier, model file paths, reference sensor timestamp, site capacity, and confidence model assets for the SeaTac demonstration site. This design allows the same inference framework to be extended to additional sites by editing a JSON configuration file rather than rewriting the main script.

In addition to the code itself, the repository includes setup guidance for future users. The README file provides instructions for preparing trained model files, creating the configuration file, creating the PostgreSQL calibration table, installing required Python (programming) packages, and running the inference script. This documentation helps move the project from a research prototype toward a transferable software component that can be reviewed, reproduced, and adapted by project partners or operational teams.

8.2 TMS SOFTWARE TEAM INTRODUCTION MEETING

As part of outreach and coordination for long-term deployment, the University of Washington team introduced the Truck Parking Information Management System (TPIMS) and the calibration framework to the WSDOT Transportation Management Systems (TMS) software team during a meeting held on March 13, 2026. The purpose of the meeting was to present the current TPIMS architecture; describe its sensing, prediction, anomaly detection, and calibration components; and discuss future software support and system transition needs.

During the presentation, the UW team provided an overview of the statewide TPIMS deployment, including the use of truck parking sensors, real-time data collection, occupancy prediction, API/web application services, and the new anomaly detection and calibration module. The system currently integrates real-time sensing and predictive analytics and has been deployed at sixteen sites along the I-5 corridor, including 273 parking stalls. The presentation also highlighted that the prediction component uses an LSTM-based model to forecast parking availability from ten minutes to eight hours ahead, and that the anomaly detection component combines radar data, reference sensor data, and prediction outputs to classify sensor status as OK, Moderate, or Suspect. Calibration recommendations are then generated to support corrective action.

The meeting discussion focused on both immediate software support needs and longer-term transition planning. Key topics included the need for TPIMS operational continuity, possible takeover and hosting arrangements, front-end support, and the role of the TMS software team in future maintenance. The discussion noted that reliable hosting and system redundancy will be important considerations, especially as the project moves away from temporary or limited hosting arrangements. The team also discussed the value of reducing manual intervention through AI-assisted validation and calibration, and we recognized that more automated ground-truth generation would make long-term implementation easier and more scalable.

The meeting also provided an opportunity to discuss future expansion and sustainability. Notes from the discussion indicated that this work aligns with broader investment and expansion efforts, including an upcoming Infrastructure for Rebuilding American (INFRA) grant and extension of truck parking efforts to additional corridors such as I-90. The TMS team's involvement was discussed as an important step toward building a more reliable, maintainable, and operationally supported TPIMS platform.

9. CONCLUSIONS AND FUTURE WORK

9.1 CONCLUSIONS

This study developed a machine learning–based framework for anomaly detection and calibration to improve the reliability of truck parking occupancy estimates in in/out counting-based Truck Parking Information Management Systems (TPIMS). While in/out counting sensors provide a cost-efficient and scalable solution, counting errors may accumulate over time as a result of missed detections, congestion, or irregular vehicle behavior, leading to inaccurate occupancy estimates.

To address this issue, the framework integrates prediction-based anomaly detection, multi-sensor comparison, and sequence classification. The prediction model learns the normal temporal pattern of parking occupancy and identifies anomalies when observed values deviate significantly from predicted trends. The system evaluates discrepancies between radar and reference camera sensors and classifies the sensor status into three categories: OK, Moderate, and Suspect. In addition, a confidence score estimates the reliability of the classification.

When strong evidence of sensor error is detected, the system provides calibration recommendations by leveraging reference sensor measurements to correct radar-based estimates. The framework was integrated into the existing TPIMS architecture and implemented through a Streamlit-based monitoring dashboard that allows operators to visualize sensor status, confidence levels, and recommended corrections. Overall, the approach improves the reliability of occupancy data and provides a practical tool for monitoring sensor performance in operational environments.

9.2 FUTURE WORK

Although the framework demonstrated promising results, several opportunities remain for further improvement and expansion. Future work should focus on expanding the framework using larger and more diverse ground-truth datasets collected across multiple sites and longer time periods. Additional calibration strategies can also be explored to address different types of sensor discrepancies beyond overcounting errors. Further research may investigate advanced anomaly detection approaches, such as semi-supervised or self-supervised learning methods, to reduce reliance on labeled data. Finally, integrating sensor reliability information into occupancy prediction models could further improve the accuracy and robustness of truck parking information systems.

REFERENCES

Abdul-Aziz, A., Woike, M., Oza, N., Matthews, B., & Baakilini, G. (2010). Propulsion health monitoring of a turbine engine disk using spin test data. In T. Kundu (Ed.), *Health monitoring of structural and biological systems 2010* (Vol. 7650, pp. 431–440). SPIE. https://doi.org/10.1117/12.847574

Analysis Division FMCSA. (2011). *2010–2011 Hours of service rule regulatory impact analysis RIN 2126-AB26*. Retrieved from https://www.fmcsa.dot.gov/sites/fmcsa.dot.gov/files/docs/2011_HOS_Final_Rule_RIA.pdf

Arias del Campo, F., Guevara Neri, M. C., Vergara Villegas, O. O., Cruz Sánchez, V. G., Ochoa Domínguez, H. de J., & García Jiménez, V. (2021). Auto-adaptive multilayer perceptron for univariate time series classification. *Expert Systems with Applications, 181*, 115147.

Bayraktar, M. E., Arif, F., Ozen, H., & Tuxen, G. (2015). Smart parking-management system for commercial vehicle parking at public rest areas. *Journal of Transportation Engineering, 141*(5), 04014094.

Bong, D., Ting, K., & Lai, K. (2008). Integrated approach in the design of car park occupancy information system (COINS). *IAENG International Journal of Computer Science*.

Boniol, P., Liu, Q., Huang, M., Palpanas, T., & Paparrizos, J. (2024c). Dive into time-series anomaly detection: A decade review. *arXiv preprint arXiv:2412.20512*.

Boniol, P., Paparrizos, J., Kang, Y., Palpanas, T., Tsay, R. S., Elmore, A. J., & Franklin, M. J. (2022). Theseus: Navigating the labyrinth of time-series anomaly detection. *Proceedings of the VLDB Endowment, 15*(12), 3702–3705.

Boniol, P., Paparrizos, J., Palpanas, T., & Franklin, M. J. (2021). Sand in action: Subsequence anomaly detection for streams. *Proceedings of the VLDB Endowment, 14*(12), 2867–2870.

Boniol, P., Paparrizos, J., & Palpanas, T. (2023). New trends in time series anomaly detection. In *Proceedings of the EDBT Conference* (pp. 847–850).

Boniol, P., Paparrizos, J., & Palpanas, T. (2024a). An interactive dive into time-series anomaly detection. In *Proceedings of the IEEE 40th International Conference on Data Engineering (ICDE)*.

Boniol, P., Sylligardos, E., Paparrizos, J., Trahanias, P., & Palpanas, T. (2024b). ADecimo: Model selection for time series anomaly detection. In *Proceedings of the IEEE 40th International Conference on Data Engineering (ICDE)*.

Boureau, Y. L., Ponce, J., & LeCun, Y. (2010). A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)* (pp. 111–118).

Breunig, M. M., Kriegel, H.-P., Ng, R. T., & Sander, J. (2000). LOF: Identifying density-based local outliers. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*.

Bressler, A. and Schuster, G., "Telematics-controlled truck parking at the motorway A3 service and rest area 'Montabaur'," Road Authority (LBM) Rheinland-Pfalz, Germany, Tech. Rep., 2008.

Bui, C., Pham, N., Vo, A., Tran, A., Nguyen, A., & Le, T. (2018). Time series forecasting for healthcare diagnosis and prognostics with the focus on cardiovascular diseases. In *Proceedings of the International Conference on the Development of Biomedical Engineering in Vietnam (BME6)* (pp. 809–818). Springer.

Chang, I., Tiao, G. C., & Chen, C. (1988). Estimation of time series parameters in the presence of outliers. *Technometrics, 30*(2), 193–204.

Chang, M., Terzis, A., & Bonnet, P. (2009). Mote-based online anomaly detection using echo state networks. In B. Krishnamachari, S. Suri, W. Heinzelman, & U. Mitra (Eds.), *Proceedings of the International Conference on Distributed Computing in Sensor Systems (DCOSS)* (Lecture Notes in Computer Science, Vol. 5516, pp. 72–86). Springer. https://doi.org/10.1007/978-3-642-02085-8_6

Chauhan, S., & Vig, L. (2015). Anomaly detection in ECG time signals via deep long short-term memory networks. In *Proceedings of the International Conference on Data Science and Advanced Analytics (DSAA)* (pp. 1–7). https://doi.org/10.1109/DSAA.2015.7344872

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. E. (2020). A simple framework for contrastive learning of visual representations. In *Proceedings of the International Conference on Machine Learning (ICML)* (Vol. 119, pp. 1597–1607). PMLR.

Chowdhury, R. R., Zhang, X., Shang, J., Gupta, R. K., & Hong, D. (2022). TARNet: Task-aware reconstruction for time-series transformer. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 14–18).

Cook, D. J., Morris, T., Morellas, V., & Papanikolopoulos, N. (2014, May). An automated system for persistent real-time truck parking detection and information dissemination. In *2014 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 3989–3994). IEEE.

Dave, I. R., Gupta, R., Rizve, M. N., & Shah, M. (2022). TCLR: Temporal contrastive learning for video representation. *Computer Vision and Image Understanding, 219*, 103406.

Dierke, J., Kleine, J., and Lehmann, R., "Intelligent controlled compact parking for modern parking management on german motorways," Transp. Res. Proc., vol. 15, pp. 620–627, 2016. doi: 10.1016/j.trpro.2016.06.052.

Du, J., & Chen, D. (2025). Position: Untrained machine learning for anomaly detection by using 3D point cloud data. *arXiv preprint arXiv:2502.03876*.

EasyWay Project. (2015). *Freight & logistics services intelligent and secure truck parking deployment guideline* (Tech. Rep. F&L-DG01, Version 02-02-00).

Fallon, J., & Howard, K. (2011). SmartPark truck parking availability system: Magnetometer technology field operational test results (No. FMCSA-RRT-10-041).

Federal Highway Administration. (2018). *Truck parking availability detection and information dissemination*. Retrieved from https://ops.fhwa.dot.gov/freight/infrastructure/truck_parking/workinggroups/technology_data/product/best_practices.htm

Federal Motor Carrier Safety Administration. (2019). *Large truck and bus crash facts 2017*.

Ford Torrey, I. W., & Murray, D. (2017). *Managing critical truck parking tech memo #2: Minnesota case study – Utilizing truck GPS data to assess parking supply and demand*. ATRI.

Fox, A. J. (1972). Outliers in time series. *Journal of the Royal Statistical Society: Series B (Methodological), 34*(3), 350–363.

Foumani, N., Miller, L., Tan, C. W., Webb, G. I., Forestier, G., & Salehi, M. (2024). Deep learning for time series classification and extrinsic regression: A current survey. ACM Computing Surveys, 56(9), 1-45.

Franceschi, J., Dieuleveut, A., & Jaggi, M. (2019). Unsupervised scalable representation learning for multivariate time series. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 4652–4663).

G. H. S. a. G. G. S. Mitsuru Saito. (2016). "Calibration of Automatic Performance Measures," Utah Department of Transportation, Salt Lake City.

Gertler, J., & Murray, J. (2011). *SmartPark truck parking availability system: Video technology field operational test results* (No. FMCSA-RRT-10-002).

Goldstein, M., & Dengel, A. (2013). Histogram-based outlier score (HBOS): A fast unsupervised anomaly detection algorithm. In *Proceedings of the KI-2012: Poster and Demo Track* (pp. 59–63).

Gu, J., Zhang, Z., Yu, F., & Liu, Q. (2012, July). Design and implementation of a street parking system using wireless sensor networks. In *IEEE 10th International Conference on Industrial Informatics* (pp. 1212–1217). IEEE.

Guan, Y., Feng, W., Palacios-Garcia, E. J., Vásquez, J. C., & Guerrero, J. M. (2019, May). Vicinity platform-based load scheduling method by considering smart parking and smart appliance. In *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)* (pp. 248–254). IEEE.

Haque, K. et al., "Truck parking utilization analysis using truck GPS data," in Proc. TRB 95th Annu. Meeting Compendium of Papers, 2016.

Haque, K., Mishra, S., Paleti, R., Golias, M. M., Sarker, A. A., & Pujats, K. (2017). Truck parking utilization analysis using GPS data. *Journal of Transportation Engineering, Part A: Systems, 143*(9), 04017045.

Hawkins, D. M. (1980). *Identification of outliers*. Springer Netherlands.

Horn, K. (2015). *Virginia truck parking study*. Virginia Department of Transportation. Retrieved from https://vtrans.virginia.gov/resources/VDOT_2022_Truck_Parking_Study.pdf

Huynh, T., Kornblith, S., Walter, M. R., Maire, M., & Khademi, M. (2022). Boosting contrastive self-supervised learning with false negative cancellation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (pp. 986–996). IEEE.

I-95 Corridor Coalition, 2018, "I-95 corridor coalition truck parking workshop – Final report," Tech. Rep., Retrieved from https://tetcoalition.org/wpcontent/uploads/2018/07/I95CC_TruckParkingWorkshop_FinalReport_fordistribution.pdf, Dec 01, 2022

Image Sensing. (2024). *Solutions | Image Sensing*. Retrieved from https://www.imagesensing.com/solutions.html

Kampakis, S. (2022). *3 types of anomalies in anomaly detection*. Retrieved from https://hackernoon.com/3-types-of-anomalies-in-anomaly-detection

Ke, R., Zhuang, Y., Pu, Z., & Wang, Y. (2020). A smart, efficient, and reliable parking surveillance system with edge artificial intelligence on IoT devices. *IEEE Transactions on Intelligent Transportation Systems, 22*(8), 4962–4974.

Lei, T., Gong, C., Chen, G., Ou, M., Yang, K., & Li, J. (2023). A novel unsupervised framework for time series data anomaly detection via spectrum decomposition. *Knowledge-Based Systems, 280*, 111002.

Leslie, C. S., Eskin, E., & Noble, W. S. (2002). The spectrum kernel: A string kernel for SVM protein classification. In *Proceedings of the Pacific Symposium on Biocomputing* (pp. 566–575).

Leslie, C. S., & Kuang, R. (2004). Fast string kernels using inexact matching for protein sequences. *Journal of Machine Learning Research, 5*, 1435–1455.

Li, D., Chen, D., Jin, B., Shi, L., Goh, J., & Ng, S.-K. (2019). MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks. In *Proceedings of the International Conference on Artificial Neural Networks (ICANN)* (pp. 703–716). Springer.

Lin, T., Rivano, H., & Le Mouël, F. (2017). A survey of smart parking solutions. *IEEE Transactions on Intelligent Transportation Systems, 18*(12), 3229–3253.

Lopez-Jacobs, V., Ellerbee, J., Hoover, M., & Fleming, G. (2013). *SmartPark technology demonstration project* (No. FMCSA-RRT-13-054). United States, Federal Motor Carrier Safety Administration.

Luley, D., Comer, B., Moore, D. C., & Boris, C. (2018). *MAASTO Regional Truck Parking Information Management System (TPIMS)*.

Ma, J., & Perkins, S. (2003). Time-series novelty detection using one-class support vector machines. In *Proceedings of the International Joint Conference on Neural Networks* (Vol. 3, pp. 1741–1745). IEEE.

Ma, Q., Zheng, J., Li, S., & Cottrell, G. W. (2019). Learning representations for time series clustering. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 3776–3786).

Malhotra, P., TV, V., Vig, L., Agarwal, P., & Shroff, G. (2017). TimeNet: Pre-trained deep recurrent neural network for time series classification. In *Proceedings of the European Symposium on Artificial Neural Networks (ESANN)*.

Malhotra, P., Vig, L., Shroff, G., & Agarwal, P. (2015). Long short term memory networks for anomaly detection in time series. In *Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)* (Vol. 23). http://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2015-56.pdf

Meng, Q., Qian, H., Liu, Y., Xu, Y., Shen, Z., & Cui, L. (2023). Unsupervised representation learning for time series: A review. *arXiv preprint arXiv:2308.01578*.

Modi, P., Morellas, V., & Papanikolopoulos, N. P. (2011). *Counting empty parking spots at truck stops using computer vision*. University of Minnesota.

Mohammadi Foumani, N., Miller, L., Tan, C. W., Webb, G. I., Forestier, G., & Salehi, M. (2024). Deep learning for time series classification and extrinsic regression: A current survey. *ACM Computing Surveys, 56*(9), 1–45.

Morris, T., Murray, D., Fender, K., Weber, A., Morellas, V., Cook, D., & Papanikolopoulos, N. (2017). *A comprehensive system for assessing truck parking availability*. Center for Transportation Studies, University of Minnesota.

Morris, T., Pananikolopolous, N., Henderson, T., & Morellas, V. (2018). *A realtime truck availability system for the state of Wisconsin*. University of Minnesota.

Nedap, "Guiding truckers in port of Hamburg." Accessed on: Oct. 10, 2021. [Online]. Available: https://www.nedapidentification.com/cases-archive/

Niu, Z., Yu, K., & Wu, X. (2020). LSTM-based VAE-GAN for time-series anomaly detection. *Sensors, 20*(13), 3738.

Omnisight. (2024). *Fusion sensor | AI sensor for smart solutions Omnisight*. Retrieved from https://omnisightusa.com/

Rodier, C. J., & Shaheen, S. (2007). *Commercial vehicle parking in California: Exploratory evaluation of the problem and possible technology-based solutions* (California PATH

Research Report UCB-ITS-PRR-2007-11). Retrieved from https://itspubs.ucdavis.edu/download_pdf.php?id=1235

Rousseeuw, P. J. (1984). Least median of squares regression. *Journal of the American Statistical Association, 79*, 871–880. https://doi.org/10.1080/01621459.1984.10477105

Saigo, H., Vert, J.-P., Ueda, N., & Akutsu, T. (2004). Protein homology detection using string alignment kernels. *Bioinformatics, 20*(11), 1682–1689.

Sander, J., Ester, M., Kriegel, H.-P., & Xu, X. (1998). Density-based clustering in spatial databases: The algorithm GDBSCAN and its applications. *Data Mining and Knowledge Discovery, 2*(2), 169–194. https://doi.org/10.1023/A:1009745219419

Siemens A. G. (2013). *Sitraffic Wimag*. Retrieved from https://www.mobility.siemens.com/mobility/global/SiteCollectionDocuments/en/road-solutions/urban/infrastructure/Sitraffic-Wimag-en.pdf

Smartmicro. (2024). *Smartmicro | Sensor technology*. Retrieved from https://www.smartmicro.com/

Sun, W., Stoop, E., & Washburn, S. S. (2018). Evaluation of commercial truck parking detection for rest areas. *Transportation Research Record, 2672*(9), 141–151.

Tsay, R. S., Peña, D., & Pankratz, A. E. (2000). Outliers in multivariate time series. *Biometrika, 87*(4), 789–804.

Tukey, J. W. (1977). *Exploratory data analysis* (Vol. 2). Addison-Wesley.

Vital, F., Ioannou, P., & Gupta, A. (2020). Survey on intelligent truck parking: Issues and approaches. *IEEE Intelligent Transportation Systems Magazine, 13*(4), 31–44.

Wang, Y., Jantarathaneewat, N., & Chen, K. (2025). *Truck Parking Information and Management System (TPIMS)*. Final report for WSDOT.

Wang, Y., Yang, H., Liu, C., & Zhuang, Y. (2022). *Truck parking detection sensor and system review*. Research report for WSDOT.

Washburn, S. S., Sun, W., & Stoop, E. (2016). Commercial truck parking detection technology evaluation for Columbia County rest areas.

Washington State Department of Transportation. (2016). *Washington State Truck Parking Study*. Retrieved from https://www.wsdot.wa.gov/NR/rdonlyres/A72C532D-B825-4757-B4BE-F00ABF93A6D6/0/TruckParkingStudyfFinal.pdf

Wavetronix LLC. (2016). "SmartSensor Matrix Case Study - Tampa, Florida,". Accessed on: Nov 28, 2022. [Online]. Available: https://www.youtube.com/watch?v=jp5rbITUv18.

Woodrooffe, J., Blower, D., & Sullivan, J. (2016). *Evaluation of MDOT truck parking information and management system*. University of Michigan Transportation Research Institute.

Williams, L. (2019). The whole story of… parking: The world of parking is no longer stationary. Engineering & Technology, 14(2), 56-61.

Wu, W., He, L., Lin, W., Su, Y., Cui, Y., Maple, C., & Jarvis, S. (2020). Developing an unsupervised real-time anomaly detection scheme for time series with multi-seasonality. *arXiv*. http://arxiv.org/abs/1908.01146

Xing, Z., Pei, J., & Keogh, E. (2010). A brief survey on sequence classification. *ACM SIGKDD Explorations Newsletter, 12*(1), 40–48.

Xu, B., et al. (2013). Realtime street parking availability estimation. In *Proceedings of the IEEE International Conference on Mobile Data Management* (Vol. 1, pp. 16–25). IEEE. https://doi.org/10.1109/MDM.2013.12

Yang, S., Ma, W., Pi, X. and Qian, S., 2019. A deep learning approach to real-time parking occupancy prediction in transportation networks incorporating multiple spatio-temporal data sources. Transportation Research Part C: Emerging Technologies, 107, pp.248-265.

Yang, H., Liu, C., Zhuang, Y., Sun, W., Murthy, K., Pu, Z., & Wang, Y. (2021). Truck parking pattern aggregation and availability prediction by deep learning. *IEEE Transactions on Intelligent Transportation Systems*.

Yoon, J., Jarrett, D., & van der Schaar, M. (2019). Time-series generative adversarial networks. In *Advances in Neural Information Processing Systems (NeurIPS)* (pp. 5509–5519).

Zerveas, G., Jayaraman, S., Patel, D., Bhamidipaty, A., & Eickhoff, C. (2021). A transformer-based framework for multivariate time series representation learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 2114–2124).

Zhao, H., Wang, Y., Duan, J., Huang, C., Cao, D., Tong, Y., Xu, B., Bai, J., Tong, J., & Zhang, Q. (2020). Multivariate time-series anomaly detection via graph attention network. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)* (pp. 841–850). IEEE.

Zheng, Y., Liu, Q., Chen, E., Ge, Y., & Zhao, J. L. (2014). Time series classification using multi-channels deep convolutional neural networks. In *Proceedings of the International Conference on Web-Age Information Management (WAIM)* (pp. 298–310). Springer.

Zhong, S. (2005). Semi-supervised sequence classification with HMMs. *International Journal of Pattern Recognition and Artificial Intelligence, 19*(2), 165–182.

Title VI Notice to Public

It is the Washington State Department of Transportation's (WSDOT's) policy to assure that no person shall, on the grounds of race, color, or national origin, as provided by Title VI of the Civil Rights Act of 1964, be excluded from participation in, be denied the benefits of, or be otherwise discriminated against under any of its programs and activities. Any person who believes his/her Title VI protection has been violated, may file a complaint with WSDOT's Office of Equity and Civil Rights (OECR). For additional information regarding Title VI complaint procedures and/or information regarding our non-discrimination obligations, please contact OECR's Title VI Coordinator at (360) 705-7090.

Americans with Disabilities Act (ADA) Information

This material can be made available in an alternate format by emailing the Office of Equity and Civil Rights at wsdotada@wsdot.wa.gov or by calling toll free, 855-362-4ADA(4232). Persons who are deaf or hard of hearing may make a request by calling the Washington State Relay at 711.