Aalto-yliopisto
Aalto-universitetet
Aalto University

Master's Programme in Spatial Planning and Transportation Engineering

A Machine Learning Framework for Public Transport Ridership Estimation Using Multi-Source Data Fusion with Low-Cost Bluetooth Data

Chris Yung

Master's thesis 2024

Aalto-yliopisto
Aalto-universitetet
Aalto University

Author Chris Yung

Title of thesis A Machine Learning Framework for Public Transport Ridership Estimation Using Multi-Source Data Fusion with Low-Cost Bluetooth Data

Programme Spatial Planning and Transportation Engineering

Major Spatial Planning and Transportation Engineering

Thesis supervisor Prof. Claudio Roncoli

Thesis advisor(s) Charalampos Sipetas, PhD

Date 31.12.2024 **Number of pages** 59 + 6 Appendix **Language** English

Abstract

Accurate information about demand volumes at certain locations within public transport networks is critical to making informed decision by transportation planners. Traditional manual counts to collect volumes, while accurate, are costly and labour intensive. Existing automatic passenger counting systems also face limitations in terms of cost, accuracy, or compatibility.

This paper proposes a multi-source data fusion framework to improve the capability of passenger counting using a low-cost Bluetooth sensor. The frameworks combine otherwise independent and unrelated raw Bluetooth counts, novel drone data and freely available General Transit Feed Specification - Real-Time and ferry schedule data into a unified and comparable format. The proposed framework leverages the advance capabilities of various machine learning models, K-Nearest Neighbour, XGBoost and Random Forest to estimate ridership of public transport vehicles in an area affected by nearby ferry operations.

The results demonstrate the models generated using the framework achieve high accuracy and low errors when compared to the ground truth of manual counts. Machine learning model vastly outperform standard Linear Regression model with a R^2 value of 0.86 compared to 0.62. Models incorporating variables develop from the framework significantly outperform those that rely solely on Bluetooth data (R^2 of 0.86 vs -0.49).

Notably, the framework is still able to draw similar conclusion when utilizing the drone counts as the ground truth which expose the model with significantly more data points then manual counts. However, discrepancy between manual count and drone count highlight the need for further validation to enhance the reliability of this approach. Nevertheless, the framework highlights the value of multi-sensor data fusion as a necessary enhancement to improve the utility and accuracy of the low-cost Bluetooth count.

Keywords Data Fusion, Machine Learning, Public Transport, Bluetooth Data, Drone Data, Automated Passenger Counting

Contents

List of Figures

List of Tables

Acknowledgements

This work was supported by Horizon Europe project ACUMEN (no. 101103808). The author thanks Forum Virium Helsinki for access to Bluetooth data and MobiLysis for access to drone data. The author thanks representatives of both entities for discussions and exchange of ideas on this work.

The author would like to thank Dr. Charalampos Sipetas, and Dr. Claudio Roncoli for their support and guidance throughout the writing process.

Helsinki, 31 December 2024
Chris Yung

Symbols and abbreviations

Abbreviations

APC	Automatic Passenger Counting
GDPR	General Data Protection Regulation
GTFS	General Transit Feed Specification
GTFS-RT	General Transit Feed Specification - Real-Time
HSL	Helsingin Seudun Liikenne
IoT	Internet Of Things
ITS	Intelligent Transportation Systems
KNN	K-Nearest Neighbour
LOOCV	Leave-One-Out Cross-Validation
LR	Linear Regression
MA	Moving Average
MAC	Media Access Control
MAE	Mean Absolute Error
ML	Machine Learning
MSE	Mean Squared Error
OD	Origin-Destination
PTA	Public Transport Agency
RAM	Random Access Memory
RF	Random Forest
RMSE	Root Mean Squared Error
RSSI	Received Signal Strength Indicator
SMAPE	Symmetric Mean Absolute Percentage Error
TEN-T	Trans-European Transport Network
UAV	Unmanned Aerial Vehicles
XGBoost	Extreme Gradient Boosting

1 Introduction

Accurately obtaining ridership information for public transport networks enhances transport management to achieve efficient and effective transport services. Stop-level passenger counts are essential for developing an effective transportation network, with adequate routes and schedules that efficiently match travel demand in the long term (Marques and Pitombo, 2022). Continuous collection of stop-level passenger counts enables the capture of congested time and seasonal trends that negatively impact the user's perception of service reliability, especially if crowding is experienced (Jenelius, 2018) and allows for dynamic adjustment to meet unexpected passenger demand. Overall, having stop-level passenger information is critical for developing advanced models, methods, and tools to support decision-making and improvements in transport networks. This is even more critical while pursuing seamlessness in multimodal networks, where aspects such as comfort and service characteristics play a pivotal role in the attractiveness and usability of the various modes.

Traditional manual methods of collecting ridership information at stops are labour-intensive, costly, and therefore obtained solemnly. Public Transport Agencies (PTA) have employed various traditional Automatic Passenger Counting (APC) to collect this information using various detection technologies. These include load sensors, treadle mat sensors, infrared sensors (Ivano Pinna, 2010). While effective in predicting passenger flows, traditional APCs require large upfront investment as it requires installation of sensors at each door in addition to continuous maintenance and potentially, ongoing subscription fees. This financial burden often limits their widespread adoption, particularly for small or cash-constrained PTAs warranting the need to explore other cost-effective alternatives.

With the emergence of the Internet of Things (IoTs) and the decrease cost of computer power, there has been a shift to develop new APCs methods. One relatively new technology utilizes Wi-Fi or Bluetooth to track or estimate the presence of a device, such as a smartphone, smartwatch, or tablet. An inherent problem of such counting methods is the dissociation of the ground truth from the detected number of devices (Sharifi et al., 2010). For example, the Wi-Fi or Bluetooth detection methods might end up overcounting or undercounting numbers of people due to one person carrying multiple or no detectable devices. Furthermore, the increasing prevalence of Media Access Control (MAC) address randomization introduced by various mobile operation systems to increase user privacy creates additional challenges by obscuring device identifiers, complicating the process of accurately tracking and counting individuals (Jouans et al., 2020). The use of MAC addresses is often associated with strict privacy regulations that might obscure part of the necessary information. Despite these limitations, the low cost of acquiring and maintaining IoT devices still makes this an attractive option for adoption.

Another novel technology for counting pedestrians involves the usage of Unmanned Aerial Vehicles (UAV) or drones to collect video footage which are increasingly used for determining vehicle (Barmpounakis et al., 2017) and pedestrian traffic parameters (Sutheerakul et al., 2017) with cost and flexibility cited as advantages over fixed camera systems. However, the benefits of drone technology in passenger related studies are yet to be explored.

With these limitations and developments in mind, this paper seeks to present a multi-source data fusion technique by integrating IoT Bluetooth sensor data, General Transit Feed Specification Real-Time (GTFS-RT) and ferry data, and aerial drone imagery to analyse passenger boarding and alighting at a specific tram stop. This integration addresses inherent gaps in individual data sources while ensuring compliance with privacy standards by avoiding the identification of MAC addresses. The goal of the data fusion approach is to extract the most relevant features from each data source: Bluetooth sensors provide anonymized, proximity-based movement data; GTFS-RT data offers real-time updates on vehicle locations and their arrivals and departures; and aerial drone imagery supplements manual counts. To achieve robust predictions, the paper evaluates various machine learning models to effectively integrate these diverse data sources, aiming to accurately predict passenger flow at the specified stop.

2 Literature review

This section provided a review of existing APC systems, examining the different available technologies available on the market and their respective benefits and drawbacks. Particular attention will focus on Bluetooth and Wi-Fi, as well as drone technology within APC systems. Research development utilizing data fusion techniques within public transportation is explored highlighting its potential to address the limitations of each individual APC technology. Lastly, the identified research gaps are discussed providing the foundation and relevance of this study.

2.1 Measuring Passenger Volumes

As discussed previously, determining ridership information provides valuable insights into the public transportation network and determines if the current allocation of resources is adequate to meet travel demand and react if an adjustment is needed.

In general, ridership can be estimated from ticket data and the adoption of manual or automatic collection method as identified by Barabino et al. (2014). The four identified classifications are shown in Table 1, which is directly adopted from its research.

Table 1: Classification of Measurement of Passenger Volume

Collection Method	**Transaction Dependent**	**Transaction Independent**
Manual	Travel Document Sold	Manual Check
Automatic	Electronic Farebox	APCs

On transaction strategies, a PTA can estimate the total ridership by counting the total number of transactions over a particular time period. Attempting to disaggregate this data presents significant challenges. It, for example, cannot provide the total number of passengers on a particular route at a particular time making this less than ideal. The usage of automatic electronic fareboxes can mitigate these limitations by recording where each validation of tickets occurs, on which route and time stamping each transaction. This additional layer of data can make each transaction more granular allowing PTA to estimate ridership across route and time.

However, there are numerous drawbacks to this approach due to its dependency on transactions. Depending on the PTA's policy, some operate on fare collection upon boarding only and passengers do not tap out missing passenger exit transactional records. Operators can also operate on an open system where passengers are not required to tap in or out if they possess a valid ticket while onboard. Passengers with passes do not tap in or out missing these passenger's transactions. Fare evaders, passengers traveling without a valid ticket, are not accounted with transactions dependent methods. Additionally, PTA is increasingly opting for alternative ticketing and payment systems like mobile ticketing citing increased revenues, strengthening brand reputation and reduction of operational costs as reasons for adoption (Apanasevic and Markendahl, 2018). Depending on how each PTA adopts mobile ticketing, the attachment of transactional data to boardings is not guaranteed as passengers can purchase and validate their tickets remotely before boarding. With these issues and future developments, the ticket dependent system is likely to be less attractive and unreliable, especially on disaggregated data.

Avoiding the limitations of utilizing transaction data, transaction-independent collection methods are envisioned. Manual methods of counting, whether through ticket inspections or trained human counter onboard can be employed to measure passenger volumes and are commonly used by many PTAs for data collection. However, employing manual counting comes with high labour cost and PTA with its limited budget is often hesitant to commit to. Data, when and if they are collected, are scarce, typically focusing on peak hours or at problematic areas. Collection spans are often limited, typically restricted to a few hours over a couple of days. Continuous data or large-scale data is rarely feasible. The limited dataset fails to provide the necessary scale to accurately measure passenger volumes and trends across all routes and times. PTA constrained by finance yet seeking to obtain this valuable information continuously turn towards automated collection methods. This is where APC solutions present a viable solution to address these needs.

2.2 APC Systems

A wide range of APC systems has been used over the past decades as highlighted in various APC literature reviews by Barabino et al. (2014), Grgurevi et al. (2022) and Radovan et al. (2024). As highlighted by all, each system presented has numerous advantages and disadvantages. Typically, each system can fall under the categories of traditional systems like optical sensor and pressure sensor technology which has been adopted earlier then the emerging and novel categories such as computer vision and Wi-Fi or Bluetooth based systems which represent newer approaches.

The usage of treadle mat sensing technology is a type of pressure sensor technology that has been adopted in APC systems. This is one of the simplest technologies and utilizes an array of sensors arranged in a grid overlayer with a plastic mat on top and triggers the sensors when an external force, typically stepped upon by a passenger, is applied (Ivano Pinna, 2010). Although cost-effective to install, miscounting errors, especially in crowded vehicles and the inability to differentiate between boarding and alighting makes this technology unideal.

Weighting-based APC systems meanwhile are more complex and derives its estimation by calculations from mass in motion. For example, research conducted by Kovacs et al. (2009) calculates the total weight of the vehicle by utilizing driving dynamics, traction, track geometry and energetic data which in turn can predict the specific capacity utilization of a tramline in Budapest. In a practical example in Copenhagen, researchers exploited installed weighing systems onboard that control braking to determine the number of passengers counts. A comparison to the onboard infrared system shows more accurate passenger counts with the new system and was fully adopted by the operator. (Nielsen et al., 2013) There are limitations of this APC technology as weight sensors onboard vehicles are not sensitive to small changes passenger load as a passenger's weight is much lighter than the vehicle's weight making detect low boarding values difficult. The sensors also cannot differentiate between the weight of luggage and passenger. Most research in this field has been related to rail based modes. (Radovan et al., 2024) In addition, as noted by Grgurevi et al. (2022), typical research into this topic leverages existing sensors and there are few papers on this topic due to significant infrastructure investment required if moving beyond existing sensors. Translation of this APC of this type to other modes of transport also remains unclear.

Another commonly used APC system is optical sensors placed at entry and exit points to count the total number of passengers boarding. This system typically uses infrared beams are often the most prevalent APC technology used by PTAs. Two distinct systems active and passive are used (Ivano Pinna, 2010). An active optical sensor consists of a transmitter and receiver and placed horizontally across a door. When the infrared beam admitted from the transmitter is interrupted, the receiver senses a breakage of the beam and counts the number of passengers. A passive sensor meanwhile utilizes infrared cameras to determine changes in heat signature at the doors. Since body temperature are higher than the ambient temperature, the detector can determine the changes in heat signature to detect changes in passenger volumes onboard a vehicle (Radovan et al., 2024). The accuracy of such system is generally high as determined by numerous studies such as Olivo et al. (2019), Barabino et al. (2014) and Kimpel et al. (2003) but all research observed either undercounting and overcounting in real life environment. The mentioned inaccuracy typically mitigated with statistical adjustments. Despite the accuracy, outfitting each door across the entire fleet can be cost-prohibitive for PTAs. PTAs often equip a subset of its entire fleet with this technology to give a statically sufficient sample but not complete picture within the network. This can limit the comprehensiveness of passenger flow data across the entire network and limit real time understanding of passenger flow.

Moving towards the realm of newer APC solutions, computer vision is one type of newer APC solution using optical cameras to count the number of passengers onboard by a software algorithm. As identified by Grgurevi et al.

(2022), the process identifies the shape of each passenger by the contrast from the background. Operationally, the concept is simple but shifts the processing to software and the accuracy is dependent on the camera's resolution and computer vision algorithms. Currently, this technology's accuracy is directly proportional to the number of cameras which hinders its adoption from the cost perspective. Khoudour (2010) was able to provide real-time APC system with high accuracy using dense stereovision at the expense of installing multiple cameras for adequate coverage. Nasir et al. (2018) meanwhile developed a method to capture the total number of passengers utilizing a single camera pointed toward seating passengers based on the detection of skin colour. In this case, the practitioner mentions the difficulty of counting passengers at the back of the bus and the impossibility of counting during overcrowded situations and overlapping passengers. Sipetas et al. (2020) also encounter similar undercounting accuracy problems when using automated video counting technology to detect left behind passengers on a metro platform. The tendency of the cameras to count foreground passengers, where each person is larger than the background passengers and the difficulty of separating bodies during crowded events were cited as reasons for discrepancies between the automatic counts to ground truth. The discrepancy was however mitigated with a logistic regression model. McCarthy et al. (2025) which deployed a video-based passenger counting on board a bus showed good accuracy although difficult capturing counts and prevalent false positives were observed when buses approached high passenger density. An outstanding issue of computer vision is the question of privacy. Data protection laws like the EU's GDPR can limit what kind of data this system can collect and train upon.

2.3 Bluetooth and Wi-Fi based APC Systems

Bluetooth and Wi-Fi APC systems operate on the ability to sense and track the unique signal admitting from a various devices like smartphones, smartwatches and tablets. Each device with the Bluetooth or Wi-Fi capabilities admits a unique MAC address to search for a valid connection continuously. Scanners are placed in various locations to record the presence of devices by storing their unique MAC addresses. Both Bluetooth (Kostakos et al., 2013) and Wi-Fi (Myrvoll et al., 2017) APC systems operates similarly to the above concept and the two distinct emitting technology are used interchangeably. To the author's knowledge, no research to date has identified advantages of one admitting technology over the other within APC systems.

A problem with Bluetooth and Wi-Fi based APC system is the dissociation of ground truth and raw count. A sensor is unable to differentiate between passengers and non-passenger's results resulting in erroneous counting when utilizing the raw results. The number of mobile devices that emit data various from person to person. Even if a person carries a device capable of emitting data, the user must enable Wi-Fi or Bluetooth component in order to be detected (Grgurevi et al., 2022).

Early adoption of removing such noise from Bluetooth and Wi-Fi signals within transportation stems from determining vehicular traffic routes. Multiple sensors are periodically spread out for a determined route by the roadside collecting MAC addresses and timestamping them. When a particular MAC address is consecutively logged between two sensors, it is then able to be matched that the device has travelled this particular route, counted it and the difference in time logs is used to estimate travel time and speed. (Bachmann et al., 2013) As noted by research that utilizing MAC matching, there is the inverse problem of utilizing raw data counts, undercounting of the ground truth to the ground truth of manual counts. A study conducted in Maryland and Delaware found sampling rate is between 2.0% and 3.4% when compared to the ground truth (Sharifi et al., 2010). Other similar MAC address matching techniques regard vehicular traffic conducted by Bachmann et al. (2013) and Tayeb et al. (2023) data showed a similar low percentage capture rate. Attempts to translate this technique to public transport by tracking how long a MAC address is detected by the sensor and recreating an Origin-Destination (OD) matrix by Kostakos et al. (2013) also experience a low detection rate of 9.7%.

Even with undercounting challenges, the concept of using MAC address matching to determine route choice is problematic from a privacy perspective although it was originally conceived to be more protective of a user's privacy (Haghani et al., 2010). This is because a device and therefore, a user, can be tracked as it moves without their consent. Apple and Google, the two major developers of mobile operating systems have since released MAC address randomization in their respective operating systems by utilizing a false MAC address while in search of a connection and changes the false MAC address periodically to hinder MAC address matching. This creates uncertainty about the matching method (Myrvoll et al., 2017) and Transport for London has since stated that due to MAC address randomization, they can no longer reliably recreate journeys utilizing MAC addresses (Transport for London, 2017). A recent article by Demetrio et al. (2024) who utilized Wi-Fi and Bluetooth detectors aboard buses to reconstruct passenger flows acknowledges MAC address randomization is a major limiting factor for their methodology as a device's MAC address can periodically change while on its journey. Other researchers like Grenville et al. (2023) attempt to circumnavigate the MAC address randomization by tracking users that have authenticated and connected to a free Wi-Fi network offered by the PTA. An authenticated device provides the same MAC address even during periodical disconnection in between stations reenabling MAC address tracking. This success was temporarily short-lived as reported by Hudes (2024) the free Wi-Fi service this study used will be discontinued by 2025. Pronello et al. (2024) also mitigated this issue by placing Wi-Fi sensors at bus stops and utilizing the features generated from Wi-Fi probe requests raw output and equipping it with various deep learning methods to predict the number of people at the stop. By shifting the focus from those who are onboard to waiting at stops, MAC randomization is avoided altogether. Other attempts attack this issue straight on with studies from Jouans et al. (2020) showing MAC address randomization can still associate a device after randomization. Methods were developed by Nitti et al. (2020) to derandomize the addresses and reenabling MAC address tracking within public transport. Preliminary control testing simulating bus behavior on a car shows an accuracy rate of 93%. It should be noted that new technological developments and changes to the MAC randomization protocol can thwart these approaches creating a never-ending cat and mouse game. Even with technical limitations removed, MAC addresses are considered personal data by the GDPR increasing legal uncertainty with MAC address matching techniques.

2.4 Video Footage Collection by Drone for Pedestrians Analysis

A new technology for data collection for transportation involves the use of UAVs or drones to collect video footage replacing stationary cameras as a mode of collection. Drones are typically less expensive and provides a greater flexibility then stationary cameras (Sutheerakul et al., 2017) although they also face legal limitations on how and where they can be flown (Alamouri et al., 2023). Typically, drone collect video footage is processed later, either manually or automatically. Diogenes et al. (2007) tested the difference of manual counts and video footage collected for vehicle traffic from a stationary camera. The manually processed video demonstrated higher accuracy then manual counts, primary due to the ability to rewind video recordings. Translating this approach to video footage recorded by drones for pedestrians, Hodgson and Chang (2023) conducted a study to collect drone footage at an elementary school to determine the chosen transportation mode by students. Challenges arise during the collection due to drone camera limitation and environmental obstructions by the school's roof and surrounding trees. During manual processing, statistically significant inconsistencies were observed by the footage reviewers unless the video is further edited, and reviewers are given specific guidance on how to handle edge cases.

With automatic video processing, human errors and uncertainty can be removed giving more predictable counts. The first approaches of automatic video processing in a transportation application were used for automated traffic conflict detection (Saunier, 2007). From video footage collected by a camera, feature-based tracking methods are used to create vehicle trajectories at an intersection where trajectories are clustered to identify vehicle conflict. Transferring this approach to video collection by drones, Barmpounakis et al. (2017) were successful in extracting vehicle and pedestrian trajectory data from drone footage from a four-way intersection although some pedestrian trajectories were blocked by foliage. It should be noted from the literature review conducted by Butilă and Boboc (2022) on using UAVs for urban traffic monitoring identified 34 papers of which only 3 also included pedestrian trajectory suggesting the niche of this technology.

2.5 Data Fusion Within the Study of APC

Each APC technology has its distinct advantages and disadvantages. A comparison of three different APC systems, tested on the same bus by Moser et al. (2019) showed good results. Each technology, video, mat sensors and Wi-Fi sensing showed good accuracy with each technology yielding 90%, 84% and 80% accuracy, respectively but noted mat sensors lacked the ability for real-time analysis due to the delay of data obtainment. Although independently decent, there is still room for improvement to achieve higher accuracy, especially on the latter Wi-Fi sensing technology.

One approach to address this limitation is using data fusion which integrates data from multiple sources to increase reliability and accuracy of its prediction. The JDL model, a widely recognized framework for data fusion begins by first filtering data from numerous streams to remove noise and errors. It is then followed by transforming the heterogeneous data streams into a transferable format through a fusion algorithm. Using this data, the framework moves to predict the relationship between each data stream enabling the prediction of future state. For further information about the JDL model, refer to Azimirad and Haddadnia (2015).

An example of data fusion in public transportation is demonstrated by Giraud et al. (2016) who utilized three different datasets, smart card transaction records, APC and General Transit Feed Specification (GTFS) data to find the origin and destination of each passenger trip. First, smartcard transaction data and APC data are fused and supplemented with GTFS data to estimate where the transaction has taken place. It is then put into a destination estimation algorithm, to link the trip's origin and destination. Using this method, a total of 71.5% of linked trips were successfully calculated within this research. It is unknown what kind of APC technology is used to collect the APC data.

Another example is Pu et al. (2021) who utilized MAC address data collected from Wi-fi and Bluetooth sensor and GPS data onboard buses to recreate ridership flow and its associated ridership OD matrix. The study fused the two sources and extracted features related to the MAC address such as, RSSI and detection time and duration along with trip related metrics like speed and distance travelled. A data clustering algorithm was applied to filter out non-passenger MAC addresses from passengers. The resulting filtered passenger MAC address is then inputted into a Random Forest regression machine learning model to predict the passengers onboard. When compared to manual counts which act as the ground truth, the model achieved high accuracy of an MSE of 0.86 and 0.96 for boarding and alighting, respectively. It is, however, unclear if MAC address randomization resulted in any issues as MAC address tracking was used to derive trip related metrics.

Pronello et al. (2024) meanwhile, set up a data fusion-like technique using Wi-Fi probe requests data to develop a low-cost APC system to count the number of passengers at a stop with deep machine learning. By sniffing probe request data, 14 different types of information including signal strength from RSSI, noise floor and supported transmission speeds can be obtained while bypassing the issue of MAC address randomization. This data is then inputted into a deep machine learning model and a prediction is derived. The results obtained are promising with a mean squared error (MSE) of 1.2 persons compared to the manual counts. While this research did not involve the fusion of multiple data sources, it still aligns with the concept of data fusion by obtaining diverse features within a single data source to create a more reliable input for predictive modelling.

There are few relevant papers with data fusion that are explicitly related to public transportation. The literature review conducted by Ounoughi and Ben Yahia (2023) who searched for papers using the terms data fusion and ITS and Fadhel et al. (2024) who searched with data fusion and smart city showed many articles related to data fusion and transportation. However, the paper was dominated by articles related to vehicles or traffic states. Although some research that employed data fusion methods was eventually found by focusing on APC more broadly, data fusion was rarely explicitly mentioned within the papers. This suggests utilizing multi-source data fusion within APC is an unexplored concept.

2.6 Research Gaps and Goals

Given the desire from PTAs to measure ridership flow at a more aggregated level to drive better decision making, an APC system is preferred to collect such data automatically due to its lower cost than manual collection as previously discussed. Although many different APC technology exists within the market each with its own distinctive advantage, several challenges - accuracy, cost, and real-time processing capabilities – exist within each technology. No single APC technology can fully address the diverse requirements of PTAs and still address the deficiency of each technology, highlighting the need for innovative approaches.

The trend from PTAs to replace traditional fare collection with mobile ticketing systems due to reduced cost appeal as mentioned by Apanasevic and Markendahl (2018) has further complicated ridership data collection. Methods that utilize traditional transactional dependent data collection methods are becoming less reliable as fare collection system evolves. Coupled with the increasing adoption of smart devices, maybe as a result from a PTA's push towards mobile ticketing, of which many can emit Bluetooth and Wi-Fi signals, Bluetooth and Wi-Fi sensors-based APC systems will likely gain more predominance in the future.

Bluetooth and Wi-Fi APC systems are not without drawbacks. The accuracy of the raw count is hindered by the detection capabilities of the sensors and the unknown relationship between the passenger and the number of emitting devices. Data fusion presents a potential solution to this problem by combining complementary datasets to mitigate the weakness of only one Bluetooth and Wi-Fi sensor to increase its accuracy, especially when equipped with a suitable regression model. Furthermore, Bluetooth and Wi-Fi sensors that rely on tracking MAC addresses are hindered by the introduction of MAC address randomization due to privacy concerns. These challenges present an opportunity to explore alternative ways to enhance accuracy while maintaining privacy.

With these limitations and developments in mind, this paper seeks to propose a multi-source data fusion technique by integrating demand and supply related data sources to predict passenger flow estimation using existing machine learning methods. This data fusion approach proposes the combination of low-cost Bluetooth sensor data with other existing complementary data sources such as GTFS-RT, drone footage and ferry schedule while ensuring compliance with privacy standards by avoiding direct usage of MAC addresses. This study aims to provide PTAs with a cost effective and scalable solution to accurately predict passenger flow using Bluetooth sensors.

3 Methodology

The proposed methodology and its associated available data for this case study are presented in Figure 1. The methodology is divided into three main steps, dataset creation, variable creation, and estimation of flows, which will evaluate the feasibility of using diverse data sources to predict passenger volumes at a stop level. Each major step, it is further broken down into smaller steps to further explain the process as outlined within this section. A brief overview of the methodology is provided here, followed by detailed explanations in subsequent subsections.

The first step, dataset creation, involves the defining of the study area to identify the potential data sources that can be collected. Within this study, Jätkäsaari in Helsinki was selected. The available data that were identified to be collected within this study were manual counts, drone footage, Bluetooth data, GTFS-RT and ferry schedule data.

The second step, variable creation, uses the raw data collected from the previous step and converting it to a uniform and interpretable format across all data sources. To achieve consistency, a time series with 30 second interval bins were constructed as a compromise between accuracy and varying frequency of each data source. As Bluetooth, manual counts, and drone counts data were not collect or recorded at the same time, linear interpolation for Bluetooth, unform arrival for manual count and cumulative sum for drone count were used to align the data points at the start of each 30 second bin. GTFS-RT and ferry schedule were incorporated within the event drive data fusion based. When a tram departs based on data from the GTFS-RT, the cumulative sum of drone and manual counts is reset to zero assuming all passenger on the platform were able to board. Additionally, the time until a ferry and tram arrival and departure are recorded at the start of each bin.

In the final step, estimation of passenger flows, various machine learning models were employed to predict passenger flows. All variables created from the previous step that may supplement the Bluetooth data were inputted into 4 different machine learning model: Linear Regression (as the baseline), K-Nearest Neighbour, XGBoost and Random Forest are evaluated against the ground truth using two scenarios. Scenario 1 utilize the ground truth of manual count supplemented by the drone counts while Scenario 2 utilize drone counts as the ground truth without drone counts as a dependent variable. Each model's performance is evaluated using various statistical metrics, with the best performing model determined by the highest R^2 value.

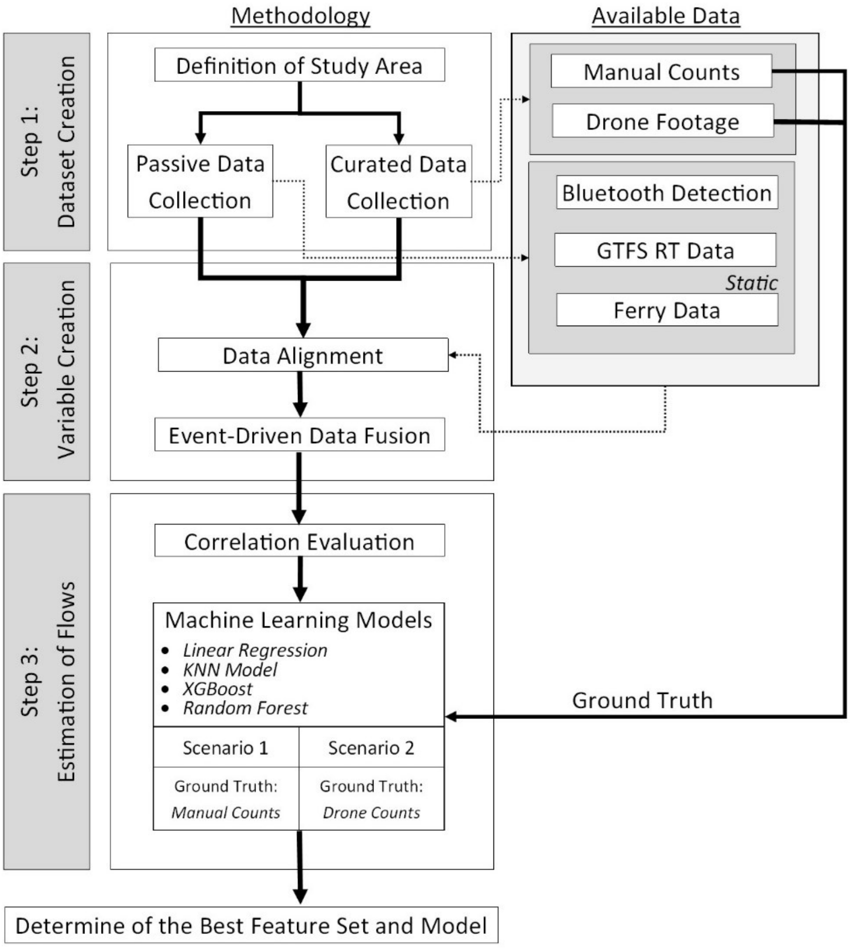

Figure 1: Plot of Proposed Methodology and Data Source

3.1 Dataset Creation

The foundation of this research is to identify which data is useful for estimating passenger flows at the stop level. A well-structured and thoroughly documented data creation process ensures the data collected is of high quality and lays the groundwork for successful data fusion and training of machine learning models later in the process.

3.1.1 Definition of Study Area

Understanding the study area is important as the underlying characteristics of the area influence the behaviour of the passenger flow of a particular stop. The study area also defines the availability of the data sources and how the physical make up indirectly influences the data that is collected.

The research is part of the Ai-aided deCision tool for seamless mUltiModal nEtwork and traffic management (ACUMEN) project, which seeks to influence traveller's route and mode choices to more sustainable modes at

Helsinki's West Harbor Area, known as Länsisatama in Finnish. The study area for this project is chosen to be within Länsisatama, a key transportation hub located in Jätkäsaari district southwest of Helsinki's downtown. The area experiences high traffic volumes as one of the region's main ferry terminals, West Terminal 2, is located within Länsisatama.

West Terminal 2 serves the ferry route between Helsinki, Finland and Tallinn, Estonia and is a vital link between Finland and the rest of Europe. The route is part of the EU's Trans-European Transport Network (TEN-T) North Sea-Baltic corridor and is identified as an important link for both passengers and freight (European Commission, 2018). The terminal sees 10 daily ferry departures and is serviced by cruiseferry ferries at various times of the day which combines the features of a cruise and a roll-on/roll-off ferry. Ferries serving West Terminal 2 are designed to accommodate trucks and private vehicles in addition to providing significant passenger carry capacity. A review of the ferries serving the terminal shows all ferries capable of carrying at least 2,000 passengers, far exceeding the carrying capacity of trams.

Helsingin seudun liikenne (HSL), the region's PTA services the terminal by the two tram stops near the terminal. During weekdays, routes 7 and 9 terminate at their respective tram stops but operate as a through service. Figure 2 shows an aerial photo of West Terminal 2, the tram stops serving the terminal and tram routes near West Terminal 2 during weekdays. The surrounding areas of the relevant area are terminal parking to the north, pick up and drop off to the west and vehicle loading bays to the east and south.

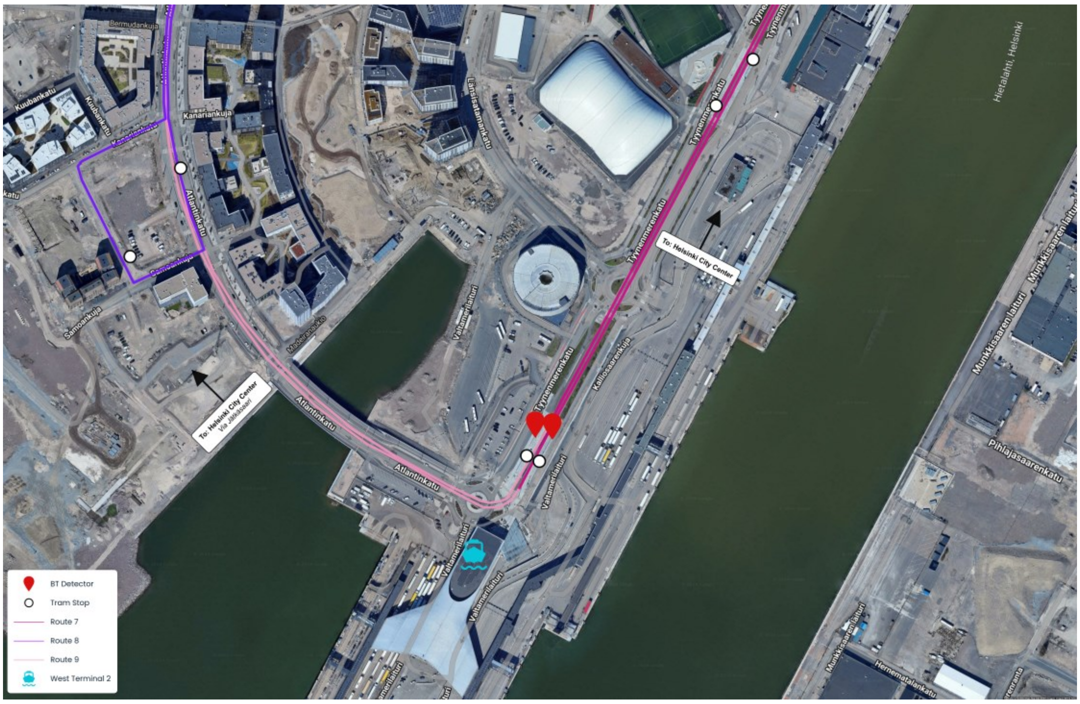

Figure 2: Chosen Study Area at Länsisatama, Helsinki, Finland

Given the potential high passenger volumes generated by the ferries and the limited carry capacity of the trams, overcrowding and unmet demand at the stops can frequently happen. This makes this area an ideal candidate for examining passenger flows of how many passengers are boarding and alighting at each tram. This valuable insight will allow the PTA to know if existing supply is sufficient to meet transportation demand.

3.1.2 Data Collection

After defining the study area, the collection of data can commence. The data to be collected depends on the source, passive or curated data. Passive data is data that can be collected continuously without the need for extensive infrastructure or operational expenditure. Curated data meanwhile are data that only can be collected intermittently, and often require a substantial delay to procure due to the need for specialized processes. The cost of curated data is also typically higher cost than passive data due to either labour, equipment or recurring expenses. Within the area, five data sources are identified, Bluetooth detection, GTFS-RT and ferry data are considered passive data while drone footage and manual counts are considered curated data.

3.1.2.1 Bluetooth Data Collection

Two LILYGO® LoRa V 2.1 Paxcounter, an IoT Bluetooth detector was installed on top of each platform's tram stop. The exact location of the detector is shown in Figure 3.

Figure 3: Location of Bluetooth Detector (Image Source: Google Earth)

As advertised by the developer, the device detects the presence of emitting devices by detecting Bluetooth signals within its range. The device would enter scanning mode every 60 seconds to detect the presence of any Bluetooth signals and their associated MAC addresses. The collected MAC addresses are stored temporarily in its RAM and then salted and hashed with a random seed value to generate unique identifiers. Each unique identifier is then counted and a raw number of the total number of detections is exported to a database via LoRaWAN. Importantly, neither the identifiers nor the MAC address is transferred to the server nor are stored permanently on the device. The identifiers and MAC address are purged from its RAM and storage each time the device enters a new scanning cycle. This process combined the salting and hashing and restricted access to the MAC addresses ensures the system cannot be used for identifying or tracking devices thus maintaining data privacy and anonymization. Similarly, by collecting Bluetooth data at fixed intervals, the issue of MAC address randomization is mitigated as the detector captures all available MAC addresses within each interval preserving the privacy of the user.

The detection range of the Bluetooth detectors is not specified. The device is designed to operate using Bluetooth V4.2 and Bluetooth Low Energy which has a theoretical range of up to 50m and 100m, respectively. The actual detection range of the device is highly dependent on numerous factors like the receiver sensitivity, transmitting power, transmitter and receiver antenna gain and path loss (Bluetooth SIG Inc., 2024) which are beyond the scope of control for this study. Consequently, regardless of the range, the data collected will encompass passenger's devices on the platform and devices outside of the platform.

To counteract the noise and help remove pedestrians passing through the detector, a moving average (MA) filter was applied to the raw Bluetooth count to help better understand the patterns. MA is a method that averages a defined number of consecutive data points to mitigate short-term variations. The MA is calculated using Equation 1:

$$MA_t = \frac{1}{N} \sum_{i=0}^{N-1} x_{t-i} \qquad 1$$

Where MA_t respective to the MA at time t, N is the averaging window of the MA and x_{t-i} denotes the data point at time $t - i$. A window size of 5 minutes was used which is half of the tram's headway of 10 minutes. It is unknown if a filter is necessary nor if an MA filter is the most effective filter as it is dependent on the character of the Bluetooth noise. To account for this, both smoothed and unsmoothed Bluetooth data were retained for analysis.

3.1.2.2 GTFS-RT

The GTFS data provides static information on planned stops, timetables, and routing of the transport system. GTFS-RT is an extension of the GTFS that adds to the feed by providing dynamic real-time tracking of public transport vehicles by using already available sensors onboard vehicles such as its GPS and speedometers. This data is available openly online and can be obtained in real time.

The data, which is provided to us by HSL includes scheduled arrival and departure times, observed arrival and departure time, route number, and vehicle number at the two stops of interest. The data sent deviates slightly from what can be obtained openly online as the open data request for GTFS-RT takes a screenshot of all public transit vehicles' state within the network while the data provided is already processed to reflect stop level information. Regardless, the source of the data still stems from GTFS-RT and methods can be developed to utilize the outputs online to derive the same information sent to us by HSL by tracking the changes of the open source's data next stop output to obtain observed arrival and departure times of the relevant tram stop. As the data is openly available and sensors are already in place, this is a passive data source due to its low cost and continuous collection.

3.1.2.3 Ferry Schedule Data

The ferry arrivals and departures represent known events that influence the demand for the stops of interest. The static data is readily available as the schedule is pre-planned by the ferry's operator and published by the terminal and was extracted to be used in this study. Although not used within this study, vessel traffic service can also be used to determine the position of the ferries in real-time and track the exact moment of the ferry's arrival providing continuous updates throughout the day. Given its pre-planned nature and potential for real-time tracking using existing systems, this data qualifies as a passive data source due to its low cost and ongoing availability.

3.1.2.4 Manual Counts

Manual collection of boarding and alighting per tram vehicle is collected by utilizing a team of volunteers. Each volunteer is stationed at a platform and assigned doors, and they count the total number of passengers boarding and alighting at the assigned doors at the respective stop. In addition to passenger counts, the tram's arrival and departure times and its route are also documented. Additionally, a volunteer is positioned at the south entrance of each stop which is the primary path between the ferry terminal and the stop. A count of the number of people passing through at five-minute intervals is recorded. The recorded data is subsequently aggregated to calculate the total boarding and alighting at each tram departure. The manual counts are collected on September 17th, 2024, between 14:00 – 15:30.

This data is considered a curated source due to its high labour and planning effort. It is cost-prohibitive to collect data over extended periods and the requirement for post-processing adds substantial delay to the availability of the data. However, this data is one of the ground truths in this study since it directly reflects the output that is expected to get from the modelling approach.

3.1.2.5 Drone Footage and Drone Counts

Drones are set up to fly and record footage that encompasses the tram stops and the terminal. This footage is subsequently processed to determine the trajectories of each pedestrian within the captured area. Two virtual gates are placed at the south entrance of each stop, which mimics the positions of the volunteers from the manual count as illustrated in Figure 4. The program first determines the object type captured by the footage and tracks only the trajectories produced by pedestrians. The program then counts the total number of pedestrians that cross a "gate" by tracking how many pedestrian trajectories cross these gates. The program records each instance of a crossing and generates an output of the total number of pedestrians that cross every 15 seconds. This output is referred to as drone counts. A total of 12.75 hours of footage is recorded spread out over three days as shown in Table 2. Footage gaps of around 5 minutes of every 25 minutes of footage within the collection period were observed due to battery swaps of the drone. Irregular gaps from other environmental factors were also observed. The drone footage and manual counts overlap on September 17th, 2024, between 14:30 and 15:30.

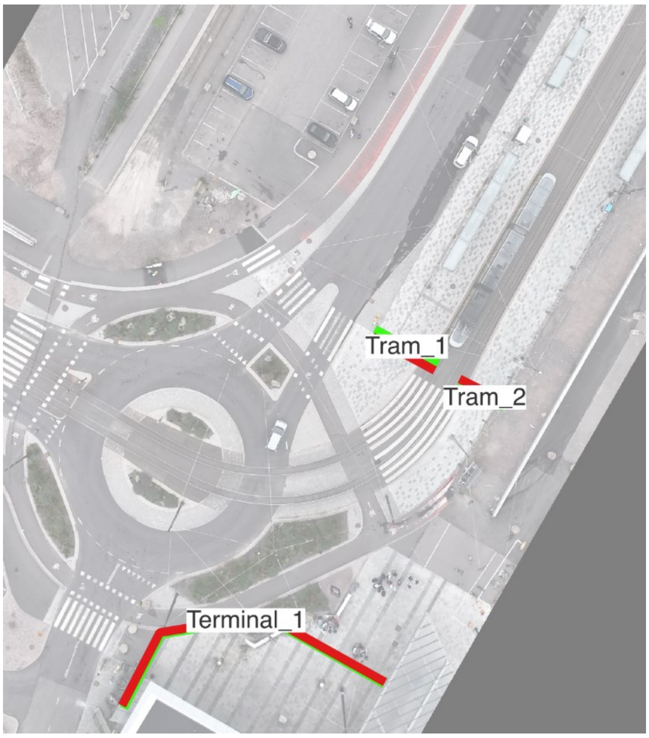

Figure 4: Location of Virtual "Gates”

Table 2: Periods of Drone Footage Collection

Date	Morning Collection	Afternoon Collection
September 16th ,2024	8:00 – 10:30	15:00 – 17 :30
September 17th ,2024	8:00 – 8:15	14:30 – 17 :00
September 18th ,2024	8:00 – 10:30	15:00 – 17 :30

The processing of the drone footage into drone counts is computationally expensive often requiring significant time for processing for the counts to be available. While drones are less labour-intensive than manual counts, a qualified operator is still required to fly the drone to collect footage, and environmental factors can limit footage collection. Drone counts are considered a procured data source due to the requirement for post-processing and the difficulty of continuously collecting data.

3.2 Variable Creation

Since the data itself is basic and incompatible with each other, combining the data sources temporally will allow for meaningful comparison and analysis. This process establishes a framework for analysing relationships between the datasets and passenger flow, which will be detailed in the following subsections.

3.2.1 Data Alignment

The process of data aligning involves aligning the raw data collected from data collection to a format that is standardized so data from different sources can be used together. The existing raw data that were collected is in different frequencies and formats, as summarized in Table 3, making comparison impartial.

Table 3: Dataset's Nature, Frequency and Format

Data Source	Nature	Frequency	Format
Bluetooth Data	Asynchronous	Every 60s – 180s	Total # of Detected Devices
GTFS-RT Data	Synchronous	Real Time Updates	Actual Tram Arrival and Departure times
Ferry Static Schedule	Asynchronous	Static Predefined Schedule	Scheduled Ferry Arrival and Departure times
Manual Count	Asynchronous	Tram Headway	Total # of boarding and alighting
Drone Count	Synchronous	Every 15s	Total # of pedestrians crossing a designated line within frequency

A binned time series for each stop is chosen to replicate the number of people on their respective platforms to ensure compatibility across datasets and reduce potential data loss. The specific implications of choosing a time series are discussed in Section 3.3.2 regarding the fusion of drone data. The width of each bin is standardized to 30 second intervals as a compromise between drone data frequency and scanning frequency of the Bluetooth detectors.

The two Bluetooth sensor's data are not temporally synchronized due to their independent operation. In addition, data is sent at irregular intervals due to periodic cellular disconnection or processing delays under heavy loads even when screening intervals are set for 60 seconds. Linear interpolation is applied to estimate the missing Bluetooth data of the start time of each bin using Equation 2.

$$BT_{est} = BT_1 + \frac{(t - t_1)}{(t_2 - t_1)} \cdot (BT_2 - BT_1) \qquad 2$$

where BT_{est} is the estimated Bluetooth value and BT_1 and BT_2 are the known values at times t_1 and t_2, respectively; t is the time of the estimated BT_{est} Linear interpolation was used due to the low frequency of the data. The average of the two separate Bluetooth detectors is used to increase the resolution for both the raw Bluetooth (BT_{raw}) and smoothed Bluetooth data ($BT_{smoothed}$) and are inputted as relevant dependent variables.

For GTFS-RT and static ferry schedules, determining the bin in which an arrival and departure event occurs is critical to reflect the supply of the trams and passenger demand from the ferry. If a departure or arrival were to occur within a bin, the observed time is recorded in the corresponding bin denoted if it is either an arrival or departure.

For the drone count data, as the data collected frequency of 15 seconds is narrower than the time bin of 30 seconds, the boarding drone counts within the wider time bin are summed to produce the total counts for the corresponding time interval.

3.2.2 Event-Driven Data Fusion

Estimation of passenger flows requires integration of multiple data sources, accounting how one data source can be a trigger for changes in other sources or influence actual passenger flow. Building upon the previously aligned data in a time series format, an event-driven data fusion is conducted to integrate all sources and derivation of potential variables. These variables are used as potential inputs to estimate passenger flow with a machine learning method.

The time series created from Section 3.2.1, which includes a ferry arrival or departure within a bin, the temporal span of a ferry event is fused by associating the ferry events with the start time of each bin. Two variables, *time since ferry arrival* ($t_{\text{since\_ferry\_arrival}}$) and *time until departure of ferry* ($t_{\text{until\_ferry\_departure}}$), are created by utilizing the start time of each bin in relation to the most recent ferry schedule arrival and next closest schedule ferry departure, respectively. Equation 3 and 4 represents the creation of the two variables.

$$t_{\text{since\_ferry\_arrival}} = (t_{\text{bin\_start}} - t_{\text{last\_ferry\_arrival}}) \text{ seconds} \qquad 3$$

$$t_{\text{until\_ferry\_departure}} = (t_{\text{next\_\_ferrydeparture}} - t_{\text{bin\_start}}) \text{ seconds} \qquad 4$$

Similarly for the tram departure and arrival, data is calculated using GTFS-RT dataset by referencing tram arrivals and departures relative to the start time of each bin, two additional variables, *time since last tram departure ($t_{\text{since\_tram\_arrival}}$)* and *time until next tram departure* ($t_{\text{until\_tram\_departure}}$) are created measuring the temporal span of a tram event. Equation 5 and 6 represents the creation of the two variables.

$$t_{\text{since\_tram\_arrival}} = \left(t_{\text{bin\_start}} - t_{\text{last\_tram\_arrival}}\right) \text{seconds} \qquad 5$$
$$t_{\text{until\_tram\_departure}} = \left(t_{\text{next\_\_tram\_departure}} - t_{\text{bin\_start}}\right) \text{seconds} \qquad 6$$

Incorporating manual count into the time series, however, needs key assumptions as the data only provides the total boarding and alighting of each tram. Focusing on the boarding data, the manually recorded tram departure is aligned to the GTFS-RT data and fused with the total number of manually recorded boardings. An assumption of a uniform arrival rate to the platform is applied due to low tram headway. Within the time series dataset, the total number of boarding was distributed across all time bins between two consecutive departures. A cumulative sum was then calculated between each departure interval to estimate the number of passengers on the platform at the start of each time bin. The cumulative sum, denoted as *manual recorded boarding* ($N_{Manual\_Boardings}$), resets to zero when a tram departure event occurs within the GTFS-RT data assuming the passenger on the platform fully dissipates and serves as one of the ground truths.

For alighting passengers, the behaviour of alighting passengers moving quickly into the terminal precludes the assumption of a uniform rate as applied in the boarding counts. This behaviour was observed during manual counts and alighting passengers do not dwell on the platform for long. The alighting passenger variable, *manual recorded alighting* ($N_{Manual\_Alighting}$), is confined to the recorded arrival bin as matched to the recorded tram arrival. This reduces the available data for alighting to one point per tram departure and severely reduces the availability of this variable limiting the capabilities of this data in the prediction model. Considering this, $N_{Manual_Alighting}$is not used as one of the variables.

Fusion of the drone data follows a similar method to the manual counts. A cumulative total of people crossing the gates from the drone data is calculated between two consecutive departures and the count is reset when a departure event is recorded in the GTFS-RT data set. However, special attention is needed to include gaps in the drone footage. If a departure event occurs within the drone footage gap, the proceeding drone counts are disregarded until the next departure event as the drone data does not reflect the total number of passengers on the platform. An additional variable, *drone recorded boarding* ($N_{Drone\_Boardings}$) is treated as dependent variables to supplement manual counts where manual counts are available as in Scenario

1. If manual counts are not available, the variable is treated as independent variables and is assumed as the ground truth as denoted in Scenario 2.

Drone recorded alighting ($N_{Drone\_Alighting}$) can also be obtained using the same process. However, the lack of data surrounding manual alighting $N_{Manual\_Alighting}$ renders obtaining $N_{Drone\_Alighting}$ less attractive. It is therefore concluded that with the dataset obtained within this study, alighting data will not be utilized or tested.

One of the challenges of utilizing drone data is the presence of footage gaps. These gaps were produced because of battery changes and other environmental factors which interrupt continuous data collection. The total number of boarding passengers cannot be estimated for both the preceding and subsequent tram if a departure falls into drone footage gap. Given typical current drone flying times are around 30 minutes as mentioned by Hodgson and Chang (2023), the temporal resolution of data is constrained, and data losses are proportional to the gap frequency.

A time series transformation was chosen over departure and arrival data points to minimize drone count losses and maximize the availability of data. This approach reduces the reliance on capturing individual departure and arrival events but instead leverages the assumption of a uniform passenger arrival rate. Deviation from this assumed behaviour may affect the accuracy of the dataset and subsequent insights.

3.3 Estimation of Flows

Using aligned data sets and their derived variables, passenger flow estimation can commence. The relationship between each variable is preliminary explored and the resulting data are input into machine learning algorithms to estimate boardings. The importance of the input variables is evaluated by utilizing performance metrics within each model to determine if the prediction of boardings by Bluetooth and drone data is feasible. Within this study, only boardings are considered due to limited data as discussed in section 3.2.2.

3.3.1 Correlation Evaluation

The derivation of the variables at this stage assumes all variables are relevant and influence boardings. Correlation evaluation identifies relevant variables that have a strong relation against the target independent variable tests against dependent variables and eliminates non-relevant variables. This process ensures the dataset that are inputted into the machine learning model later in the methodology are relevant and reduce multicollinearity and noise.

Pearson's correlation coefficient (r) as shown in Equation 7 is a statistical measure of the strength of a linear relation between two variables.

$$r = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum(X_i - \bar{X})^2 \sum(Y_i - \bar{Y})^2}} \quad 7$$

Although r measures strength on a linear relationship and not for a non-linear relation, a poor coefficient does not necessarily result in the removal of the variable. The purpose of the r is used as a quick identifier to reveal weak correlated variables to the independent variable and detect dependencies between dependent variables. Utilizing the r quickly flags potentially problematic variables and enables further investigation to determine if the variable should be included, transformed, or excluded. Subsequent machine learning models will account for non-linear relations and the r value is primarily used to help identify outliers and flag potentially weakly correlated or redundant variables to increase certainty in the subsequent machine learning model.

3.3.2 Machine Learning Models

With the variety of different variables produced from the dataset but with an unknown relationship to the ground truth, it is unknown which variables are relevant to predicting the passenger flow of a tram stop. Traditional models, like utilizing the Pearson's correlation coefficient which assumes a linear relationship is insufficient to fully justify which variable is relevant. Unlike traditional models which struggles to capture nonlinear interactions, machine learning (ML) models are more resilient in capturing complex relations and provide more robust results to diverse datasets with each model having its advantages and disadvantages over the other.

Two scenarios were created to address the limitation of the availability of data. Scenario 1 assumes the ground truth as the manual counts are treated as the independent variable with the drone count serving as supplementary to the manual count. Scenario 2 meanwhile assumes the ground truth as the drone counts and disregards the manual counts as drone data typically cost less and can continuously collect data for longer periods of time. This scenario yields significantly more data but introduces a key assumption that the drone count is comparable to the manual count.

For both scenarios, four different types of models are tested: Linear Regression, K-Nearest Neighbour (KNN) model, Extreme Gradient Boosting (XGBoost) and Random Forest. They are trained and validated with the same data produced in the previous sections using the same independent and dependent variables. All models were implemented using Python 3.12, primarily using the scikit-learn library.

Linear Regression

Linear Regression is a model that evaluates trends between the independent variable (ground truth) and dependent variables assuming the relationship is linear. The result of this model serves as a baseline comparison against more complex models.

KNN Model

The k-nearest neighbours (KNN) is a non-parametric ML model that predicts outcomes based on the proximity and similarity of the data points in the feature space. KNN makes predictions by classifying new data point according to the distance between the test data and the training data. Using a specified number of nearest neighbours (k), it predicts by averaging the target values of the neighbours. A value of k = 5 is used to balance overfitting and over smoothing, equal weighting was assumed treating all neighbours to be equally reliable and the default Minkowski distance metric is used.

XGBoost

The eXtreme Gradient Boosting (XGBoost) is an ML model that utilizes an ensemble of decision trees to predict a target value. Utilizing a single decision tree is prone to overfitting, XGBoost mitigates this by utilizing multiple trees to create a more robust model. XGBoost builds trees sequentially and each sequentially tree constructed attempts to correct the previously constructed tree's error. After multiple iterations, each sequential tree's predictions converge making XGBoost particularly effective for complex and structured datasets. The XGBoost model is configured with 100 estimators and a maximum depth of 5 to balance computational efficiency and accuracy. The learning rate was set to 0.1 to stabilize the convergence of each iteration and a seed of 42 was utilized for result replicability.

Random Forest

Similar to XGBoost, Random Forest is an ML model that constructs multiple decision trees during training built individually on a random subset of the given data. Unlike XGBoost which builds trees sequentially, Random Forest constructs multiple decision trees using bootstrap sampling (bagging), where each tree is randomly trained on a random subset of data. The final prediction is made by averaging the predictions from each tree which reduces overfitting. The model was configured with the construction of 100 decision trees and each tree had a maximum depth of 5. These values were chosen to balance computational efficiency and predictive accuracy as higher numbers often yield diminishing returns. Due to the random nature of Random Forest, a seed of 42 was utilized for result replicability.

Given the need to identify the most robust model and uncertainty of the relationship between the variables and the ground truths, all possible variable combinations, except those identified within correlation evaluation as defined in Section 3.3.1, are inputted into all 4-machine learning models. This approach ensures all variables, and their relations are tested.

To determine which combination is the best and provides the most robust results, a set of evaluation metrics is used to compare the ML models. The model performance is evaluated using Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R^2 and Symmetric Mean Absolute Percentage Error (SMAPE). MSE, MAE and RMSE are metrics that evaluate the magnitude of errors produced by the model. R^2 evaluates the variance of the model, or in other words, how well the data fits into the regression model. These four metrics are typically used to evaluate how well an ML model performs. Another typically used metric for evaluation is the Mean Absolute Percentage Error (MAPE), which measures errors relative to the size of the actual value are shown in Equation 8.

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{|y_i|} \times 100 \tag{8}$$

Where n is the total number of observations, y_i and $\hat{y}_i$ is the actual and predicted value of the i-th observation. Within the dataset, ground truth can be zero as there are no passengers on the platform. Inputting zero as the denominator in MAPE will result in a divided by zero error. SAMPE avoids this problem by scaling by the average of the actual and predicted values. SAMPE equation is shown in Equation 9.

$$\mathrm{SMAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{\frac{|y_i| + |\hat{y}_i|}{2}} \times 100 \tag{9}$$

SAMPE is bounded from 0% to 200% with 0% implying a perfect fit. Its bounded nature makes the metric ideal for the prediction of small values and is a good alternative to MAPE to measure relative errors.

While in this study, multiple evaluation metrics were calculated, R^2 was chosen to be the primary criterion for the best combination within each ML model. This decision aligns with the findings from Chicco et al. (2021), which recommends the usage of R^2 to evaluate regression analyses. Their study highlights the unbounded nature of MAE, MAPE, MSE and RMSE metrics which make them less suitable for assessing regression models. Additionally, R^2 was determined by the study to be more reliable evaluation metric to the accuracy of a regression model than SMAPE.

The amount of data points varies significantly from scenario to scenario. Within scenario 1, the available data is small making the traditional train test split method unsuitable due to the risk of introducing biases in sampling. A different approach, Leave-One-Out Cross-Validation (LOOCV) was used to test and evaluate each model for both scenarios.

LOOCV is a cross-validation technique that utilizes n−1 data points as the training dataset while the remaining point is used as the test set. This process is repeated for n times where n is the total number of data points in the dataset. Although computationally expensive, every point is used for both training and testing and averages the test errors in LOOCV as the metrics are aggregated. This ensures the results are reliable and comparable to the traditional split test for small datasets.

3.4 Testing of Assumptions from Scenario 2

As mentioned previously, Scenario 2 assumes the ground truth to be the drone counts assuming the drone count is comparable to the manual count. Knowing the limitation of video collection from drone footage and accuracy concerns from computer vision, the assumption warrants caution. It is not confirmed whether the person crossing the virtual gates is equivalent to boarding and if they will immediately board the next available tram. If significant discrepancies exist, the validity of the assumption and the association of boardings with the drone data could be questioned. To address this, a comparison between the manual counts and drone counts is conducted to assess the reliability of the measurement.

4 Results

4.1 Correlation Evaluation Results

The correlation of the potential variables is shown in Table 4 and Table 5 for stop 1 and stop 2, respectively. Correlations that utilize the same data for stops 1 and 2 are crossed out in the latter table for clarity.

Table 4: Correlation Matrix between Variables for Stop 1

	$N_{manual\ boarding}$	$N_{drone\text{-}boarding}$	BT_{raw}	$BT_{smoothed}$	$t_{since\ tram\ arrival}$	$t_{until\ tram\ departure}$	$t_{since\ ferry\ arrival}$	$t_{until\ ferry\ departure}$
$N_{manual\_boarding}$	1							
$N_{drone\_boarding}$	0.65	1						
BT_{raw}	0.63	0.64	1					
$BT_{smoothed}$	0.65	0.66	0.93	1				
$t_{since\_tram\_arrival}$	0.36	0.21	0.14	0.01	1			
$t_{until\_tram\_departure}$	-0.42	-0.22	-0.20	-0.08	**-0.89**	1		
$t_{since\_ferry\_arrival}$	-0.6	**-0.71**	**-0.71**	**-0.78**	-0.08	0.04	1	
$t_{until\_ferry\_departure}$	0.05	-0.03	0.24	0.28	-0.27	-0.03	0.13	1

Table 5: Correlation Matrix between Variables for Stop 2

	$N_{manual\ boarding}$	$N_{drone\text{-}boarding}$	BT_{raw}	$BT_{smoothed}$	$t_{since\ tram\ arrival}$	$t_{until\ tram\ departure}$	$t_{since\ ferry\ arrival}$	$t_{until\ ferry\ departure}$
$N_{manual\_boarding}$	1							
$N_{drone\_boarding}$	-0.09	1						
BT_{raw}	0.24	0.42	1					
$BT_{smoothed}$	0.24	0.55	~~0.93~~	1				
$t_{since\_tram\_arrival}$	0.44	0.45	-0.03	0.07	1			
$t_{until\_tram\_departure}$	-0.38	-0.42	0.12	0.04	**-0.98**	1		
$t_{since\_ferry\_arrival}$	0.00	**-0.63**	~~-0.71~~	~~-0.78~~	-0.02	-0.02	1	
$t_{until\_ferry\_departure}$	0.00	-0.04	~~0.24~~	~~0.28~~	-0.14	0.18	0.13	1

Highly correlated dependent variable pairs are reviewed, and their combination is removed to reduce the dimensionality of the data. $t_{since\_ferry\_arrival}$ is removed from the list of dependent Variables due to the high negative between the two Bluetooth variables correlation (-0.71 and -0.78) and against the drone counts (-0.71 and 0.62 for stop 1 and 2, respectively). It is likely the high Bluetooth counts reveal sufficient information about the ferry's arrival time. $t_{until\_ferry\_departure}$ and $t_{since\_tram\_arrival}$ have a very high negative correlation with each other and are treated as mutually exclusive variables when fed into the ML model. As expected, and discussed previously, BT_{raw} and $BT_{smoothed}$ are also highly correlated as they stem from the same base data. Both variables are also treated as mutually exclusive. However, pertaining to this research to test the viability of using low-cost Bluetooth data, the Bluetooth variables are mutually inclusive to other variables.

4.2 Scenario 1

4.2.1 Data Set After Event-Driven Data Fusion (Scenario 1)

Scenario 1 constraint of one hour of usable data due to the limited overlap between manual and drone counts. A total of 85 and 77 data points are available for stop 1 and stop 2 respectively for scenario 1. The number of data points is less than the expected 120 data points per stop, which follows the drone's available counts as it has the lowest common resolution available. The drone counts are the lowest common resolution because of the 5-minute drone footage gap for every 25 minutes of continuous drone data collection due to battery changes and the lead time after a footage gap until the next tram departure. Figure 5 and Figure 6 show the respective datasets after the event data fusion for stop 1 and stop 2, respectively for scenario 1 in graphical form.

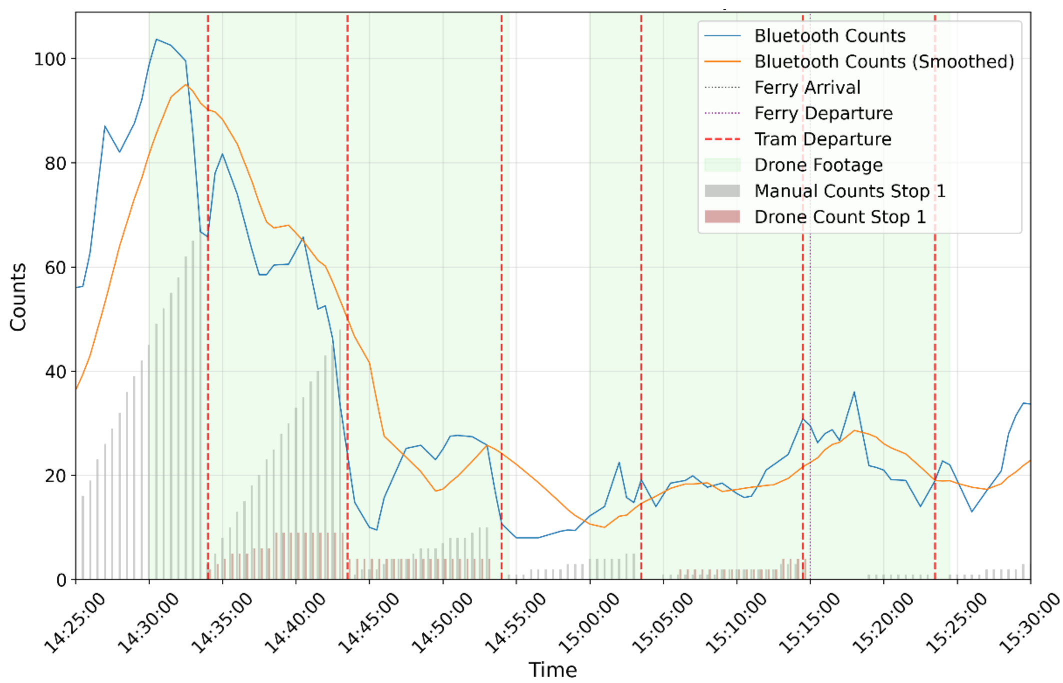

Figure 5: Data Set for Stop 1 After Event Data Fusion for Scenario 1

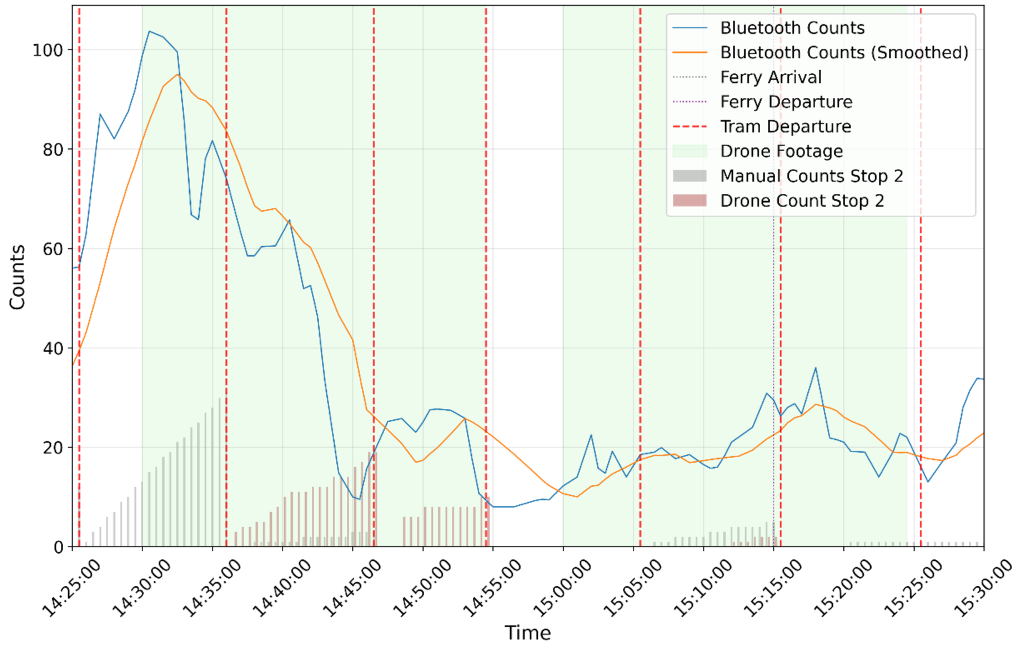

Figure 6: Data Set for Stop 2 After Event Data Fusion for Scenario 1

As shown in the figures, the blue and orange line graph shows the raw Bluetooth and smoothed Bluetooth data, respectively. The vertical red dash line shows each tram departure which will reset the cumulative drone counts to zero as shown in the red bar drone counts. The manual counts, shown by the grey bar follow a uniform linear pattern due to the uniform arrival assumption as previously discussed. No ferry arrival was observed within the overlapping hour, but a ferry departure event took place at 15:15.

The green background shows the collected drone footage. The gaps as shown in the figure highlight the challenges faced by replicating the number of passengers on the platform using the drone footage. The volume of manual counts, as shown by the grey bar, at the start of the drone footage at 14:30 until the next tram departure was missed by the drone count data. This similarly happens at the next gap of 14:55 – 15:00 which results in missing another chunk of data until the next tram departure. Three and four complete departure tram events are recorded for stop 1 and stop 2, respectively.

4.2.2 Machine Learning Models for Scenario 1

Utilizing the dataset described in section 4.2.1, 4 different machine learning models - linear regression, KNN, XGBoost and Random Forest - were developed and evaluated utilizing the LOOCV method. Figure 7 and Figure 8 illustrate the predicted results plotted against manual counts for each model for stop 1 and stop 2, respectively. Similarly, Table 6 and Table 7 show the evaluation performance metrics for stop 1 and stop 2 while Table 8 shows the best variable combination of each model, defined as the highest R^2.

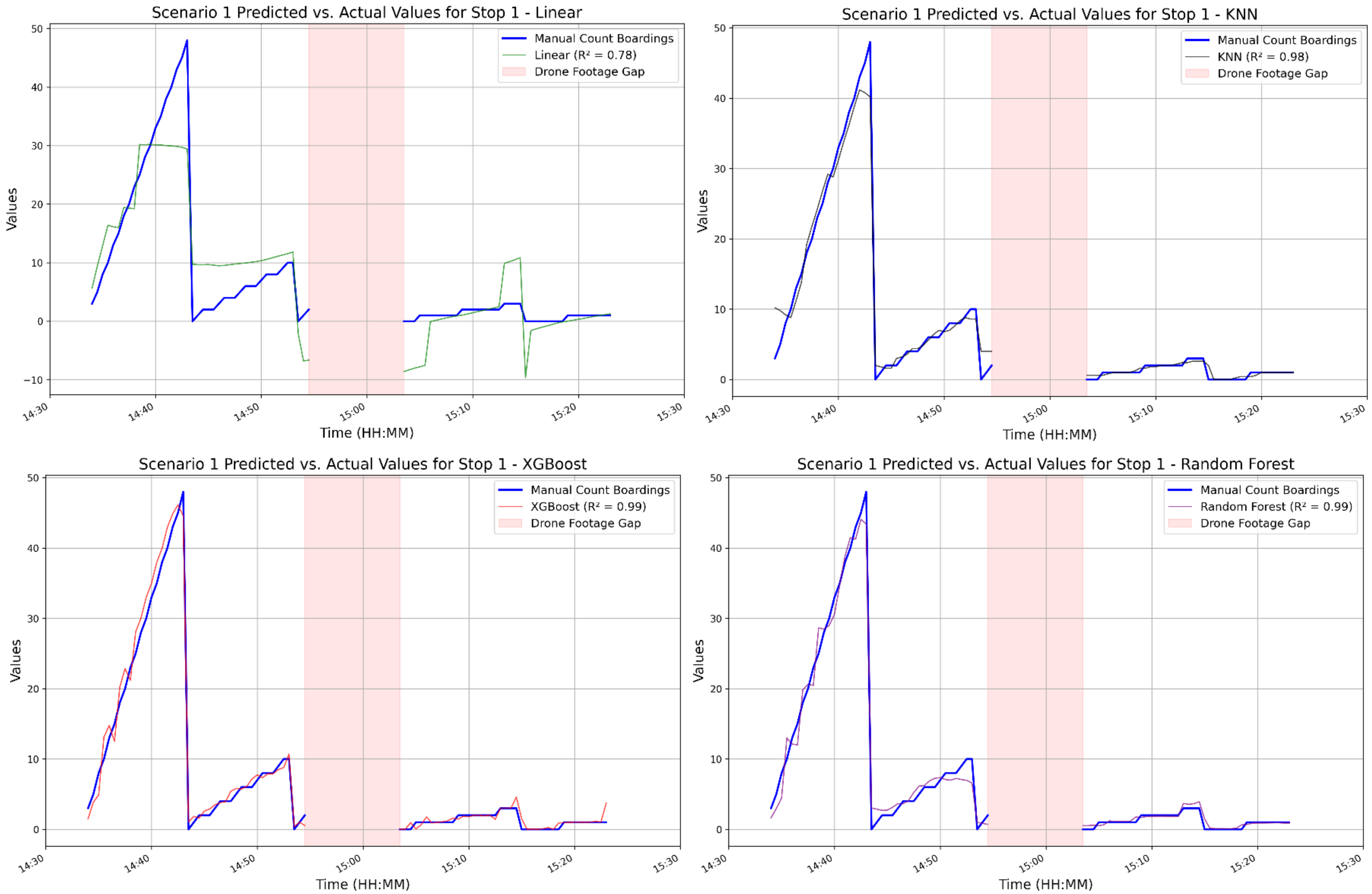

Figure 7: Predicted Boarding Vs Actual Manual Counts for Stop 1 for Scenario 1

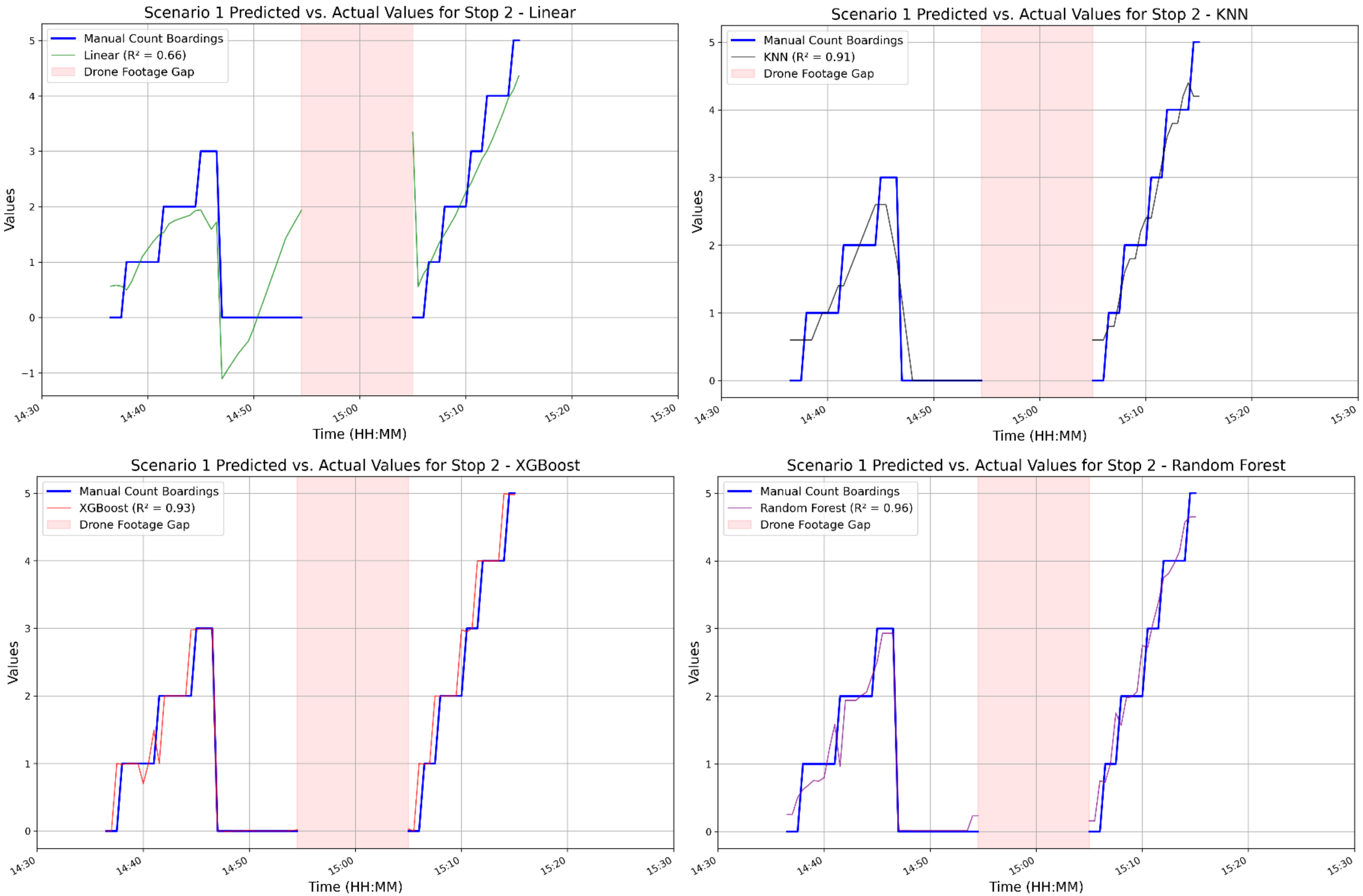

Figure 8: Predicted Boarding Vs Actual Manual Counts for Stop 2, Scenario 1

Table 6: Scenario 1, ML Models Performance Metric, Stop 1

	MSE	MAE	RMSE	SMAPE	R^2
Linear Regression	30.76	3.96	5.55	85.3 %	0.78
KNN	2.88	0.95	1.70	36.2 %	0.98
XGBoost	2.38	0.89	1.54	52.3 %	0.98
Random Forest	1.87	0.92	1.37	48.5 %	0.99

Table 7: Scenario 1, ML Models Performance Metric, Stop 2

	MSE	MAE	RMSE	SMAPE	R^2
Linear Regression	0.74	0.63	0.86	89.4 %	0.66
KNN	0.19	0.31	0.43	38.7 %	0.91
XGBoost	0.12	0.20	0.35	83.1 %	0.94
Random Forest	0.10	0.21	0.31	85.6 %	0.96

Table 8: Dependent Variables Combination, Scenario 1

	Stop 1				**Stop 2**			
	LR	**KNN**	**XGB**	**RF**	**LR**	**KNN**	**XGB**	**RF**
BT_{raw}						■		
$BT_{smoothed}$	■	■	■	■	■		■	■
$N_{Drone\_Boardings}$	■		■	■		■	■	■
$t_{\text{since\_tram\_arrival}}$	■				■		■	
$t_{\text{until\_tram\_departure}}$		■	■					
$t_{\text{since\_ferry\_arrival}}$	High Correlation to Bluetooth Counts							
$t_{\text{until\_ferry\_departure}}$	■	■	■	■	■	■	■	■

LR = Linear Regression, RF = Random Forest

Within Scenario 1, Linear Regression performed the worst compared to the other three models with the model struggling to replicate the ground truth. Negative values were predicted by the Linear Regression model for both stops 1 and 2 but were not observed for the other three more complex models. The three different models, KNN, XGBoost, and Random Forest, performed similarly in Stop 1 while Random Forest has a slight advantage with higher R^2 compared to KNN and XGBoost in Stop 2. All three complex models achieved low error rates.

Focusing the three complex models and their chosen dependent variable, the model consistently utilizes $t_{\text{until\_ferry\_departure}}$ and frequently chooses $BT_{smoothed}$ and $N_{Drone\_Boardings}$. The variables $t_{\text{since\_tram\_arrival}}$ and $t_{\text{until\_tram\_departure}}$ did not show consistent inclusion or exclusion making their impact unclear.

As Random Forest had the highest R^2 of both stops 1 and 2, a plot of all combinations used in the testing and the resulting R^2 is shown in Figure 9. A total of 32 different combinations were tested and plotted in descending order. A special marker, X is used to signify where only the Bluetooth counts were inputted into the model. For the complete set of results for every combination, refer to Appendix A.

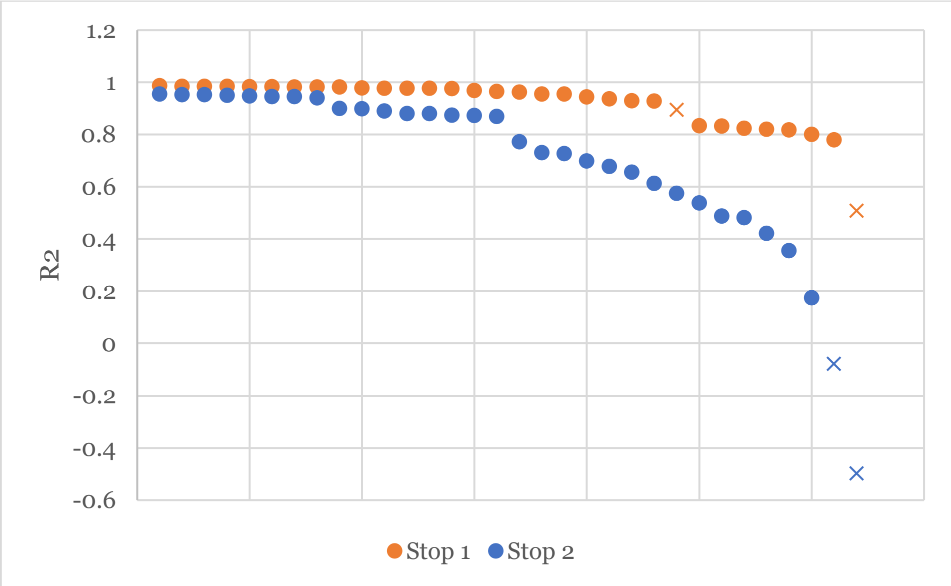

Figure 9: All Variable Combination's R^2 in Scenario 1, Random Forest

Most combinations tested were not substantially different from the best R^2. However, the models whose R^2 is below 0.5, typically used only one other variable and the selected variable is typically a supply variable (tram departures). Models that only utilize the Bluetooth counts perform very poorly as all but one model is ranked last. Stop 2's Bluetooth only models also result in negative R^2 suggesting the model to be predicting worse than the baseline mean.

4.3 Scenario 2

4.3.1 Data Set After Event-Driven Data Fusion for Scenario 2

A total of 873 and 845 data points are available for stop 1 and stop 2 respectively for scenario 2 obtained from 12.75 hours of drone footage. Again, the number of data points is less than expected due to drone footage gaps. Figure 10 and Figure 11 show the respective data set after the event data fusion for stop 1 and stop 2 during the PM drone collection period on 2024-09-17, respectively for Scenario 2 in graphical form. The figures are an excerpt of the input data, the entirety of the data used to train the ML models spans the entirety of the drone data as shown in Table 2.

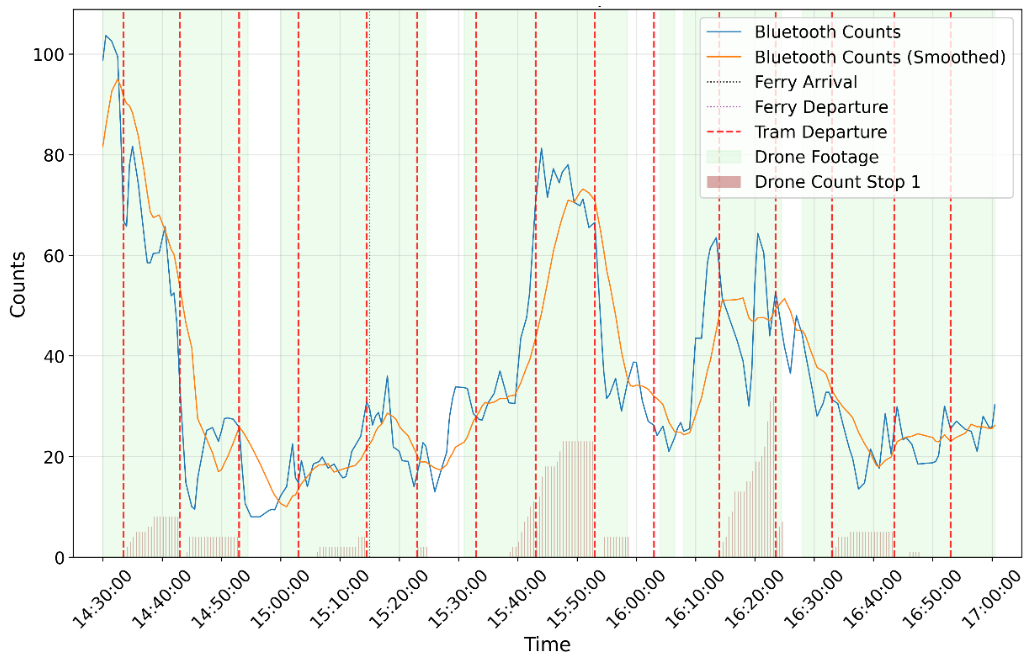

Figure 10: Data Set for Stop 1 After Event Data Fusion for Scenario 2 (2024-09-17, PM)

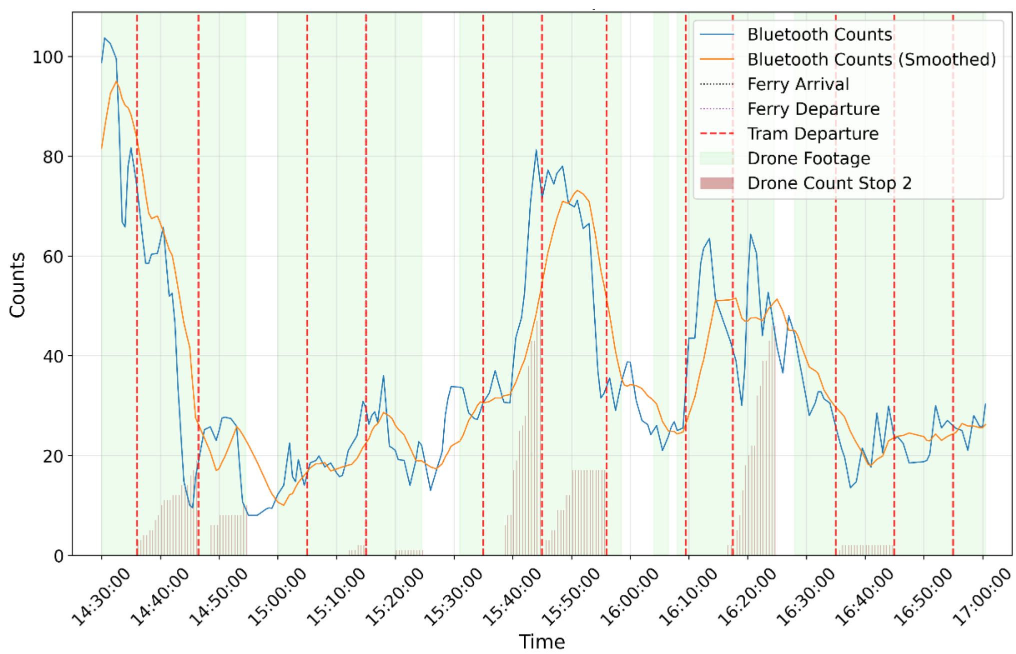

Figure 11: Data Set for Stop 2 After Event Data Fusion for Scenario 2 (2024-09-17, PM)

The figures follow the same convention as described in the previous scenario. Ferry arrival was experienced at 9:30 and 15:30 during drone footage collection while ferry departure was experienced at 9:00, 10:30, 15:15 and

16:30. A total of 61 departure events were recorded within scenario 2 compared to 7 departure events in scenario 1.

4.3.2 Machine Learning Models for Scenario 2

Utilizing the dataset for Scenario 2 the 4 different machine learning models - Linear Regression, KNN, XGBoost and Random Forest – were again developed and evaluated utilizing the LOOCV method for consistency. Figure 12 and Figure 13 illustrate the predicted results plotted against manual counts for each model for stop 1 and stop 2, respectively. Similarly, Table 9 and Table 10 show the evaluation performance metrics for stop 1 and stop 2 while Table 11 shows the best variable combination of each model, defined as the highest R^2.

Table 9: Scenario 2, ML Models Performance Metric, Stop 1

	MSE	MAE	RMSE	SMAPE	R^2
Linear Regression	17.41	2.78	4.17	134.1 %	0.41
KNN	8.22	1.59	2.87	78.1 %	0.72
XGBoost	4.51	1.25	2.12	116.2 %	0.85
Random Forest	8.14	1.73	2.85	119.9 %	0.72

Table 10: Scenario 2, ML Models Performance Metric, Stop 2

	MSE	MAE	RMSE	SMAPE	R^2
Linear Regression	100.19	6.88	10.01	131.0 %	0.62
KNN	63.56	3.89	7.97	73.2 %	0.76
XGBoost	37.69	2.91	6.14	106.6 %	0.86
Random Forest	45.36	3.70	6.73	110.3 %	0.83

Table 11: Dependent Variables Combination, Scenario 2

	Stop 1				**Stop 2**			
	LR	**KNN**	**XGB**	**RF**	**LR**	**KNN**	**XGB**	**RF**
BT_{raw}								
$BT_{smoothed}$								
$t_{\text{since\_tram\_arrival}}$								
$t_{\text{until\_tram\_departure}}$								
$t_{\text{since\_ferry\_arrival}}$	High Correlation to Bluetooth Counts							
$t_{\text{until\_ferry\_departure}}$								

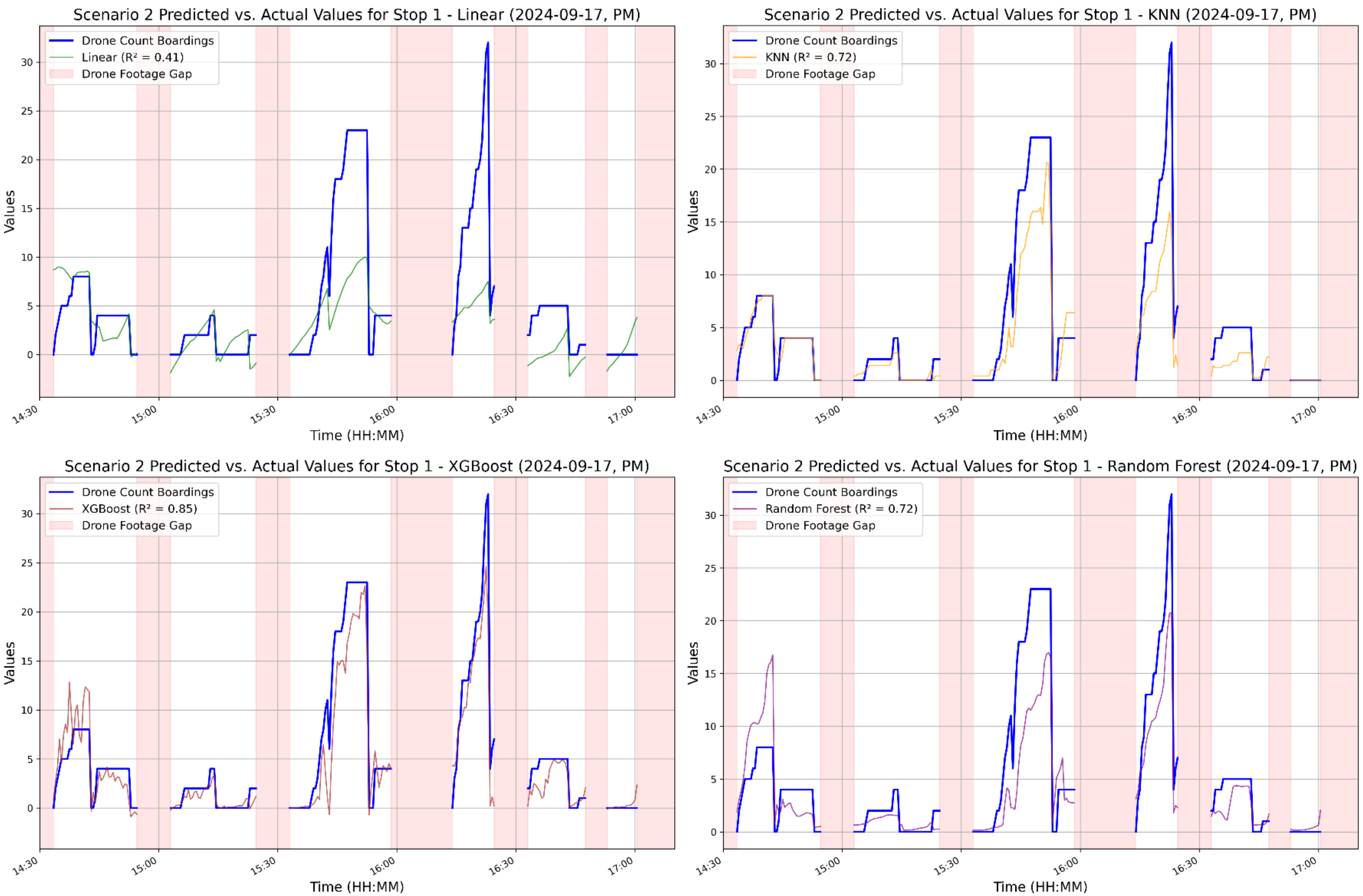

Figure 12: Excerpt of Predicted Boarding Vs Actual Manual Counts for Stop 1 for Scenario 2

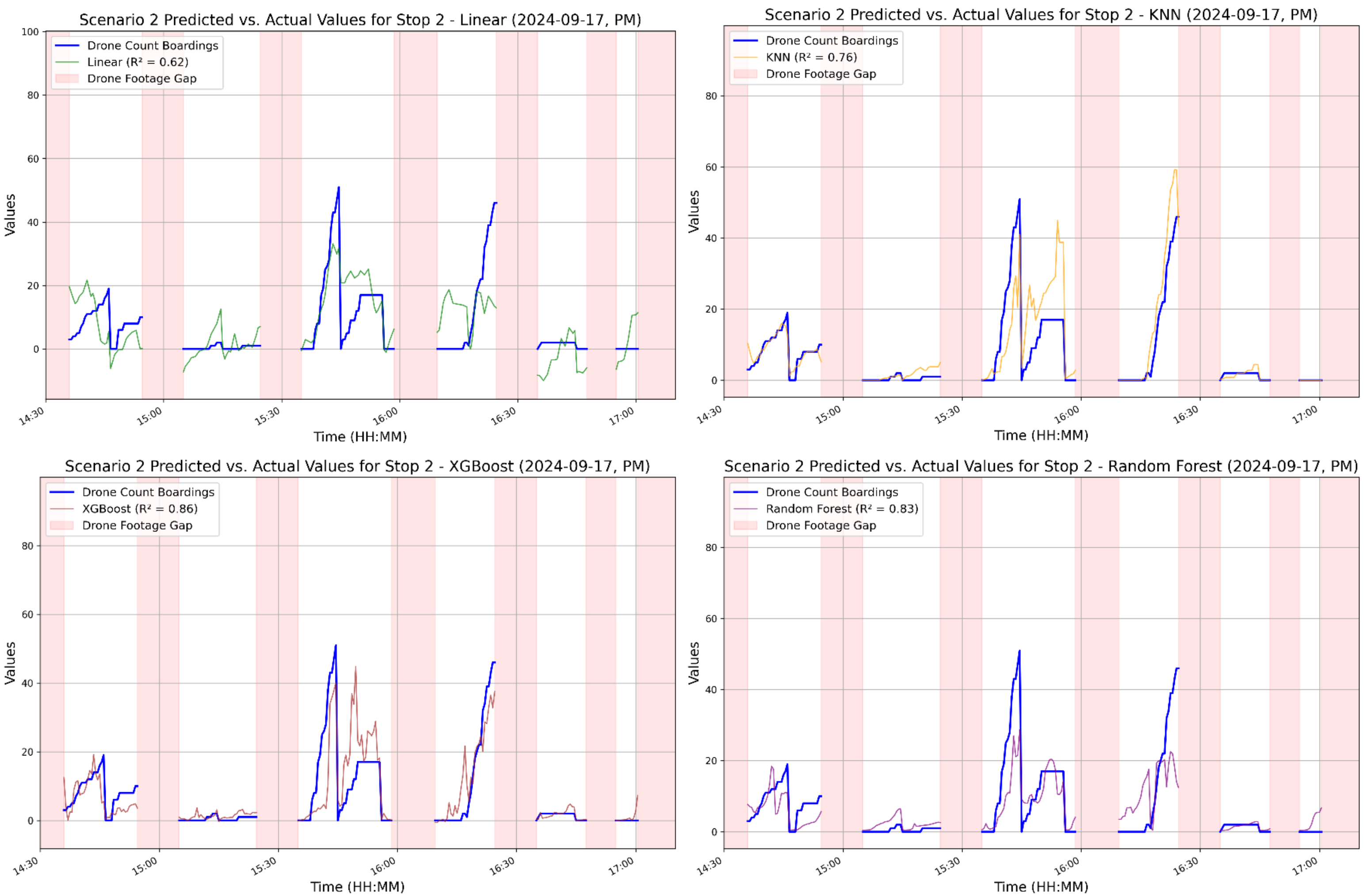

Figure 13: Excerpt of Predicted Boarding Vs Actual Manual Counts for Stop 2 for Scenario 2

Within Scenario 2, Linear Regression again performed the worst compared to the other three models with the model struggling to replicate the ground truth. Again, negative values were constantly predicted in Linear Regression which was not replicated in the other three more complex ML models. XGBoost performance is the highest in Scenario 2 with an R^2 of 0.85 and 0.83 for stops 1 and 2, respectively. KNN and Random Forest also perform well with a good R^2. Errors are significantly higher for stop 2 with all model's MSE showing almost a 10-fold increase and at least a double of MAR and RMSE. SMAPE remains relatively constant between the two stops. In contrast to the metrics from Scenario 1, errors are significantly higher and R^2 is poorer with Scenario 2, Both scenarios, however, show the model metrics are better at predicting boardings at Stop 1 than Stop 2.

Focusing on the chosen dependent variable for the three viable models, the model consistently utilizes $t_{\text{since\_tram\_arrival}}$ and $t_{\text{until\_ferry\_departure}}$. All models utilize $BT_{smoothed}$ while KNN and Random Forest utilize BT_{raw}. The variables $t_{\text{since\_tram\_arrival}}$ was chosen by all three models. Compared to Scenario 1, the utilization of $t_{\text{until\_ferry\_departure}}$ is consistent by both scenarios. Typically, the best performing models utilize $BT_{smoothed}$ against the BT_{raw} for both scenarios. However, in Scenario 1, the inclusion of tram-related variables varied, with some models excluding them entirely, leaving their overall impact inconclusive between the two scenarios.

As XGBoost had the highest R^2 of both stops 1 and 2 in scenario 2, a plot of all combinations used in the testing and their respective R^2 are shown in Figure 14. A total of 12 different combinations were tested and plotted in descending order. A special marker X is used to signify where only the Bluetooth counts were inputted into the model. For the complete set of results for every combination, refer to Appendix B.

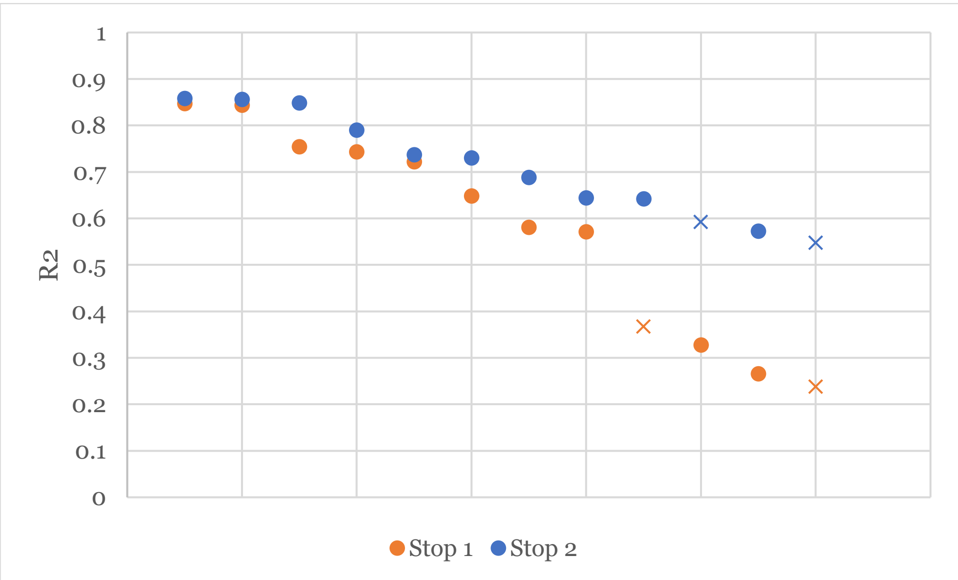

Figure 14: All Variable Combination's R^2 in Scenario 2, XGBoost

The figure shows a negative trend suggesting the results are sensitive to the variables chosen. The worst performing models were again, like Scenario 1, with tram related variables combination only, $t_{since\_tram\_arrival}$ and $t_{until\_ferry\_departure}$, performed the poorly. Models that only utilize the Bluetooth counts also performed poorly ranking last with poor R^2 values.

4.4 Testing of Assumptions from Scenario 2

A comparison of the manual counts to the drone counts to test the assumption from Scenario 2 is shown in this section and is based on the one hour overlap of the drone footage and manual data collection. Within Table 12, the table compares the manual counts, where volunteers count the number of passengers entering each platform against the drone counts, where the number of people crossing the gates. Meanwhile, Table 13 shows the total boarding counts taken from the manual counts and comparing it against the drone counts, which are the total number of people that crosses the virtual gates between two consecutive tram departures.

Table 12: Manual Counts vs Drone Counts for Crossing Virtual "Gates"

Bin	**Gate 1 / Tram Platform 1**			**Gate 2 / Tram Platform 2**		
	Drone Count	Manual Count	Difference	Drone Count	Manual Count	Difference
14:30	32	26	6	49	15	34
14:35	8	12	-4	18	4	14
14:40	6	5	1	8	1	7
14.:45	11	8	3	12	1	11
14:50	0	3	-3	5	0	5
14:55	0	2	-2	0	0	0
15:05	4	0	4	0	0	0
15:10	13	1	12	3	0	3
15:15	1	0	1	3	0	3
15:20	2	0	2	0	0	0
15:25	0	0	0	0	2	-2
SUM	**77**	**57**	**20**	**98**	**23**	**75**

Table 13: Manual Boarding Counts vs Drone Counts

Tram Departure	**Boarding (Drone)**	**Boarding (Manual)**	**Difference**
14:34	21	31	-10
14:46	8	3	5
15:05	1	0	1
15:15	13	5	-8
15:24	0	0	0

There is a significant discrepancy between the drone and manual count where drone counts are counting higher than the actual manual counts. This suggests that either the manual counts or drone counts have some unknown errors to explain the differences. There is also a significant difference between drone and manual boarding counts bringing concern about the accuracy of the assumption that manual counts and drone counts are comparable.

5 Discussions, Limitations and Future Directions

The results of the study suggest that passive Bluetooth sensor data is a good candidate for estimating the number of boarding passengers if the data is supplemented with other relevant dependent variables and utilized in a more complex machine learning regression model. The statues and impact of the drone footage itself are less certain due to issues replicating the drone counts to manual counts.

The proposed framework's importance in integrating multiple data sources to supplement Bluetooth data is shown in Figure 9 and Figure 14 as inputting only the Bluetooth counts results in poor model performance. A drastic improvement in the metrics is observed when additional variables derived from the data fusion methodology are input alone side with the Bluetooth data. This suggests there are inherent gaps with the raw Bluetooth data that need to be supplemented by these variables. The variables that were tested were a combination of supply variables ($t_{\text{since\_tram\_arrival}}$ and $t_{\text{until\_tram\_departure}}$) and demand variables ($t_{\text{since\_ferry\_arrival}}$, $t_{\text{until\_ferry\_departure}}$ $and\ N_{Drone\_Boardings}$). From the variables, supply variable-only models perform poorly compared to the best model which utilizes the demand variables and typically a supply variable. This suggesting demand variables are more likely to be impactful than supply variables. However, supply variables still have importance as most of the best models select a supply variable in addition to the demand variables.

Regarding the machine learning models within Scenario 1, the Linear Regression model struggles to estimate boarding to produce accurate results. The linear models produced negative results and fail to react accurately to the sudden reduction of passengers because of the clearing of passengers from the departure of the tram. The model struggles to capture extended periods of constant low values due to its inability to model non-linear relationships effectively. Other machine models produce much better estimates with the more complex, XGBoost and Random Forest, which are both tree ensemble models producing the best results. This is likely due to the two models being able to replicate non-linear relationships more effectively than the Linear Regression and KNN models.

There are concerns about the potential overfitting of Scenario 1 as indicated by its metrics. This risk is amplified by the low amount of data points as the model can become poor in validity, reliability, and generalizability. While LOOCV was used to mitigate the risk of overfitting, the risk cannot be fully eliminated as the model is trained on the entire dataset except for one data point in each iteration. The test data is not substantially different from the full dataset.

Another limitation is the lack of event diversity within Scenario 1 to ferry arrival, ferry departure and tram departure. For the ferry, only one ferry departure is experienced and no ferry arrivals. Additionally, the same problem is faced regarding tram departure events, where the drone footage captures the complete boarding between two tram departures. Only a total of 7 complete departure events were observed, 3 for stop 1 and 4 for stop 2. This lack of diversity may result in the model performing poorly when exposed to new, unseen data. Spikes related to the ferry arrival and departure which is a large source of passenger demand for the tram stop should be tested to observe if the model is able to replicate passenger demand.

In regards to boarding data, Ansari Esfeh et al. (2020) question the assumption of utilizing the wait time of half the headway, which is widely adopted in waiting time cost estimation if it is valid. They identified the most widely used wait time model, as shown in equation 10, developed by Spiess and Florian (1989) assumes a) uniform distribution of passenger arrivals at stop b) that headways follow an exponential distribution and c) passenger takes the first available bus.

$$Estimated\ Wait\ Time = \frac{a}{f_s} \tag{10}$$

Where a is the headway distribution (1/2 assumes equal interarrival times) and f_s is the frequency of all routes serving the stop. In this paper, the same assumption was applied, the headway distribution is uniform as this stop is the origin of the two trams stop and the frequency is constant at 10 minutes. This paper also implicitly assumes passenger estimates wait to be half of the headway. Ansari Esfeh et al. (2020) identify mean waiting times are influenced by different combinations of passenger types (planned trips or unplanned trips) and service types (schedule-based, frequency-based, high-frequency and low-frequency) as passenger arrival patterns change and result in more accurate behaviour than the assumption. Although this was attempted to be replicated by the supplemental drone counts, challenges with the drone footage limited the ability to replicate actual arrival rates. Identification of the passenger profile and adjusting the ground truth can serve as a foundation for additional investigation.

Another assumption that both Spiess and Florian (1989) and this paper utilize the assumption that a passenger takes the first available vehicle. Currently, the tram stops are served by a distinct tram route for each platform meaning the passengers will likely follow the assumption as they do not have a choice in which route to board. Previously, HSL has served each tram stop by more than one route. Another potential is the investigation of how this framework and other relevant variables can change to differentiate the route choice of each passenger.

A scenario that is not experienced during data collection is passenger left behind events. An assumption was made that all passengers are cleared after a tram departure but as confirmed by the Bluetooth Data, passenger levels are typically higher in the evening. Ferry passengers often takes advantage of the day cruise tickets offered by the ferry operators which call for morning departures and evening arrivals where the frequency of trams is lower. It is unknown how the existing model will react to left behind passengers and how this would affect the results. It is well-known among residences that passengers taking the Helsinki – Tallinn ferry typically have excessive luggage which overcrowds the tram more easily. A future development can be the exploration of models that is able to identify overcrowding events.

The lack of data was attempted to mitigate with Scenario 2 where an assumption of the drone counts is comparable to the manual count and therefore can be utilized as the ground truth. With this assumption, a drastic increase in data points materialized. This in turn mitigated the lack of event diversity from Scenario 1 as a variety of ferry and tram arrival and departure events are observed within the data. The results are promising with a similar conclusion to Scenario 1. Linear Regression struggles to predict drone counts with negatives frequently predicted. Other machine models produce better estimates with the more complex XGBoost resulting in the best model. Demand-related variables were selected more often than the supply variables.

However, the testing of assumption as shown in Section 4.4, did not conclusively demonstrate if the drone counts are comparable to the manual counts. There was a significant difference in the manual gates crossing and boarding data compared to the drone data although the data overlap was limited to only 1 hour. It would be recommended to collect more manual counts to statistically significant levels to cross verify the drone counts and adjust the algorithm if needed. Furthermore, a review of drone footage and comparison to the counting algorithm was unable to be conducted due to potential conflicts with EU GDPR data protection laws.

Numerous factors likely contributed to the difference. Site factors such as obstruction caused by trees and roof awnings could affect results like those faced by Hodgson and Chang (2023) hindering counting efforts. Barmpounakis et al. (2017) who utilize a similar tracheary approach using drone footage identified accuracy is highly dependent on the video's stabilization collected by the drone and its geo-reference. Compounded by known issues of computer vision, especially in the tendency to count passengers close to the camera (Sipetas et al., 2020) and difficulty in counting accurately during overcrowding loads (Nasir et al., 2018) and overlapping passengers (McCarthy et al., 2025), the dissociation of the drone counts to the ground truth can be expected. The assumption within this research is that passengers who cross the virtual gate will board the next available tram may also not hold. Scenario 2 and drone data appear therefore to be on uncertain grounds if counts are taken as is. The ease of collection of data by drones is still an attractive option for data collection resulting in more data points than the manual counts. However, observed discrepancies need further investigation to better understand the relation between the drone data and manual counts. By addressing this limitation, the transition to using drone counts exclusively can be materialized avoiding costly manual counts.

Future practitioners should consider the operational issues of using drone technology for transportation data collection and emulate findings by Hodgson and Chang (2023) where the drone's battery limitation limited the capture of the entire school dismissal time. This issue is similarly faced by this research with the presence of drone footage gaps which interrupt continuous data collection. Within this study, the gaps prove to be problematic as data fusion chooses the data sources with the lowest common denominator which were the drone footage. It is also unknown which environmental conditions are needed to enable night drone footage collection, a task that would be required to replicate overcrowding events that typically happen more often during the evenings within this study's location.

Previous research of similar data fusion about public transportation ridership and APC system has an explicate intent to replicate the origin and destination of ridership patterns onboard vehicles (Kostakos et al., 2013, Pu et al., 2021). The current framework proposes that a method of replicating boardings with Bluetooth sensors installed outside of the vehicle shows promise, but alighting predictions were not conducted due to the lack of data. A future direction of research is to develop methods to estimate alighting passengers and remediate potential challenges of short dwell times of alighting passengers on the platform. Further research can be conducted to determine if the OD matrix can be replicated utilizing similar off-board detection methods if both boarding and alighting can be estimated accurately.

An extension of the framework could involve applying it to a network-level analysis by incorporating variables from adjacent stops, such as predicted boardings from previous stops. It is important to note that data collected within this researched was sourced from a single location which may limit the generalizability of the findings. However, this limitation underscores the value of further expansion of the study to a broader network level. Such an approach would allow for exploration of unobserved relation between stops and trams, which were not evident when focusing on an individual stop level.

Interestingly, Pronello et al. (2024) as discussed previously in section 2 follow a similar methodology to this research. By extracting various variables from each Wi-Fi probe request and inputting it into a deep machine learning model, it was able to predict the number of people at a bus stop with high accuracy. In their conclusion, they noted, "Future work we could consider (is) associating the measurements of our system with the real-time bus passing times to monitor additional metrics".

This aligns with the findings of this study, variables taken from the supply of public transport (GTFS-RT) and demand (ferry data) show a significant influence on the accuracy of the machine learning model. This raises an intriguing question: Could combining the approach by Pronello et al. (2024) and replacing the Bluetooth counts with the Wi-Fi probe data, while retaining demand and supply variables, yield a more accurate and robust model? Such synergy could harness the strength of each dataset and potentially enhancing prediction accuracy.

6 Conclusion

The goal of this paper was to present a multi-source data fusion framework for estimating ridership onboarding tram vehicles in an area where low-cost Bluetooth data are available. The developed framework utilizes existing data sources, GTFS-RT and ferry data which represents the supply of public transport and demand of the passenger, respectively and fusing it to the available Bluetooth data. Drone data, another independent source was also collected and was used for as an independent source purposes for scenario 1 and while used as the ground truth data points for Scenario 2. The analysis showed that obtaining each data source is associated with challenges. For example, Bluetooth data, even though low cost, were very noisy, while drone counts presented some gaps due to battery recharge. Still, both datasets when fused properly show promise for generating high-quality results.

The result from Scenario 1 which utilized manual counts as the ground truth shows very low error rates and high R^2 when inputted into a more advanced ML regression model (i.e., KNN clustering, XGBoost and Random Forests) utilizing variables that refer to public transportation's demand and supply. There was evidence to suggest demand related variables have a greater impact than supply variables. Utilizing only demand variables could yield better results than using supply variables only but utilizing both demand and supply variables generates the best model. Models utilizing only Bluetooth counts results in significantly worse than those incorporating fusion-related variables. This suggests the advantages of using a multi sensor data fusion approach to increase the reliability and accuracy of the model's prediction.

The results from Scenario 2 which utilize the available drone counts as a replacement for manual count to increase event diversification and data size while still adhering to the fusion framework still generate excellent results with identical conclusion to Scenario 1. This suggests the robustness and transferability of the proposed framework. However, uncertainty remains when comparing drone counts and manual counts due to the discrepancies observed between the two measurement methods. This discrepancy could be the result from drone collection technical issue or known issues of undercounting by computer vision. Addressing this mismatch through improved calibration or tuning of algorithm could further enhance the appeal of this framework.

While this paper demonstrates the potential of multi-source data fusion to predict boarding figures, future research could further enhance the framework. Integrating alighting data, exploring additional features from Bluetooth signals, or exploring options to expand this framework to a network level. This additional information can increase the appeal of Bluetooth and drone counts for broader application of these technology within public transportation analysis.

References

ALAMOURI, A., LAMPERT, A. & GERKE, M. 2023. Impact of Drone Regulations on Drone Use in Geospatial Applications and Research: Focus on Visual Range Conditions, Geofencing and Privacy Considerations. *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science,* 91**,** 381-389.

ANSARI ESFEH, M., WIRASINGHE, S. C., SAIDI, S. & KATTAN, L. 2020. Waiting time and headway modelling for urban transit systems – a critical review and proposed approach. *Transport Reviews,* 41**,** 141-163.

APANASEVIC, T. & MARKENDAHL, J. 2018. The value of mobile ticketing from a public transport perspective. *Journal of Payments Strategy & Systems,* 11.

AZIMIRAD, E. & HADDADNIA, J. 2015. The comprehensive review on JDL model in data fusion networks: techniques and methods. *International Journal of Computer Science and Information Security,* 13**,** 53.

BACHMANN, C., ROORDA, M. J., ABDULHAI, B. & MOSHIRI, B. 2013. Fusing a Bluetooth Traffic Monitoring System With Loop Detector Data for Improved Freeway Traffic Speed Estimation. *Journal of Intelligent Transportation Systems,* 17**,** 152-164.

BARABINO, B., DI FRANCESCO, M. & MOZZONI, S. 2014. An Offline Framework for Handling Automatic Passenger Counting Raw Data. *IEEE Transactions on Intelligent Transportation Systems,* 15**,** 2443-2456.

BARMPOUNAKIS, E. N., VLAHOGIANNI, E. I., GOLIAS, J. C. & BABINEC, A. 2017. How accurate are small drones for measuring microscopic traffic parameters? *Transportation Letters,* 11**,** 332-340.

BLUETOOTH SIG INC. 2024. *Understanding Bluetooth Range* [Online]. Available: https://www.bluetooth.com/learn-about-bluetooth/key-attributes/range/ [Accessed 2024].

BUTILĂ, E. V. & BOBOC, R. G. 2022. Urban Traffic Monitoring and Analysis Using Unmanned Aerial Vehicles (UAVs): A Systematic Literature Review. *Remote Sensing,* 14.

CHICCO, D., WARRENS, M. J. & JURMAN, G. 2021. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Comput Sci,* 7**,** e623.

DEMETRIO, A., ELGNER, F., HAMEISTER, H., QUINTING, M., WARZOK, D. & WENDEL, J. 2024. Large-scale Wi-Fi and Bluetooth data collection for reconstructing passenger flows. *Journal of Location Based Services,* 18**,** 185-204.

DIOGENES, M. C., GREENE-ROESEL, R., ARNOLD, L. S. & RAGLAND, D. R. 2007. Pedestrian counting methods at intersections: A comparative study. *Transportation Research Record,* 2002**,** 26-30.

EUROPEAN COMMISSION 2018. North Sea-Baltic Core Network Corridor: 3rd Work Plan of the Coordinator Directorate General for Mobility and Transport.

FADHEL, M. A., DUHAIM, A. M., SAIHOOD, A., SEWIFY, A., AL-HAMADANI, M. N. A., ALBAHRI, A. S., ALZUBAIDI, L., GUPTA, A., MIRJALILI, S. & GU, Y. 2024. Comprehensive systematic review of information fusion methods in smart cities and urban environments. *Information Fusion,* 107.

GIRAUD, A., TRÉPANIER, M., MORENCY, C. & LÉGARÉ, F. 2016. Data fusion of APC, smart card and GTFS to visualize public transit use. CIRRELT, Centre interuniversitaire de recherche sur les réseaux d'entreprise ....

GRENVILLE, A., KLUMPENHOUWER, W. & SHALABY, A. 2023. Using Wi-Fi Connection Data to Analyze Performance of the Subway System in Toronto, Canada. *Transportation Research Record: Journal of the Transportation Research Board,* 2678**,** 764-775.

GRGUREVI, I., JURŠI, K. & RAJICˇ, V. 2022. Review of Automatic Passenger Counting Systems in Public Urban Transport. *5th EAI International Conference on Management of Manufacturing Systems*. University of Zagreb: EAI/Springer Innovations in Communication and Computing.

HAGHANI, A., HAMEDI, M., SADABADI, K. F., YOUNG, S. & TARNOFF, P. 2010. Data Collection of Freeway Travel Time Ground Truth with Bluetooth Sensors. *Transportation Research Record: Journal of the Transportation Research Board,* 2160**,** 60-68.

HODGSON, C. & CHANG, K. 2023. Using drone technology to collect school transportation data. *Travel Behaviour and Society,* 31**,** 1-9.

HUDES, S. 2024. TTC ending free Wi-Fi in subway stations by end of year. *The Canadian Press*.

IVANO PINNA, B. D. C., F DEFLORIO 2010. Automatic passenger counting and vehicle load monitoring. *Ingegneria Ferroviaria,* 65**,** 101-138.

JENELIUS, E. 2018. Public transport experienced service reliability: Integrating travel time and travel conditions. *Transportation Research Part A: Policy and Practice,* 117**,** 275-291.

JOUANS, L., VIANA, A. C., ACHIR, N. & FLADENMULLER, A. 2020. Associating the Randomized Bluetooth MAC Addresses. *HAL Open Science*.

KHOUDOUR, L. 2010. Real-time passenger counting in buses using dense stereovision. *Journal of Electronic Imaging,* 19.

KIMPEL, T. J., STRATHMAN, J. G., GRIFFIN, D., CALLAS, S. & GERHART, R. L. 2003. Automatic Passenger Counter Evaluation: Implications for National Transit Database Reporting. *Transportation Research Record: Journal of the Transportation Research Board,* 1835**,** 93-100.

KOSTAKOS, V., CAMACHO, T. & MANTERO, C. 2013. Towards proximity-based passenger sensing on public transport buses. *Personal and Ubiquitous Computing,* 17**,** 1807-1816.

KOVACS, R., NADAI, L. & HORVÁTH, G. Concept validation of an automatic passenger counting system for trams. 2009 5th International Symposium on Applied Computational Intelligence and Informatics, 2009. IEEE, 211-216.

MARQUES, S. D. F. & PITOMBO, C. S. 2022. Transit Ridership Modeling at the Bus Stop Level: Comparison of Approaches Focusing on Count and Spatially Dependent Data. *Applied Spatial Analysis and Policy,* 16, 277-313.

MCCARTHY, C., GHADERI, H., MARTÍ, F., JAYARAMAN, P. & DIA, H. 2025. Video-based automatic people counting for public transport: On-bus versus off-bus deployment. *Computers in Industry,* 164.

MOSER, I., MCCARTHY, C., JAYARAMAN, P. P., GHADERI, H., DIA, H., LI, R., SIMMONS, M., MEHMOOD, U., TAN, A. M. & WEIZMAN, Y. A methodology for empirically evaluating passenger counting technologies in public transport. Proceedings of the 41st Australasian Transport Research Forum (ATRF), Canberra, Australia, 2019.

MYRVOLL, T. A., HÅKEGÅRD, J. E., MATSUI, T. & SEPTIER, F. 2017. Counting Public Transport Passenger Using WiFi Signatures of Mobile Devices. *2017 Ieee 20th International Conference on Intelligent Transportation Systems (Itsc).*

NASIR, A., GHARIB, N. & JAAFAR, H. Automatic passenger counting system using image processing based on skin colour detection approach. 2018 international conference on computational approach in smart systems design and applications (ICASSDA), 2018. IEEE, 1-8.

NIELSEN, B. F., FRØLICH, L., NIELSEN, O. A. & FILGES, D. 2013. Estimating passenger numbers in trains using existing weighing capabilities. *Transportmetrica A: Transport Science,* 10, 502-517.

NITTI, M., PINNA, F., PINTOR, L., PILLONI, V. & BARABINO, B. 2020. iABACUS: A Wi-Fi-Based Automatic Bus Passenger Counting System. *Energies,* 13.

OLIVO, A., MATERNINI, G. & BARABINO, B. 2019. Empirical Study on the Accuracy and Precision of Automatic Passenger Counting in European Bus Services. *The Open Transportation Journal,* 13, 250-260.

OUNOUGHI, C. & BEN YAHIA, S. 2023. Data fusion for ITS: A systematic literature review. *Information Fusion,* 89, 267-291.

PRONELLO, C., ANBARASAN, D., SPOTURNO, F. & TERZOLO, G. 2024. A low-cost automatic people-counting system at bus stops using Wi-Fi probe requests and deep learning. *Public Transport.*

PU, Z., ZHU, M., LI, W., CUI, Z., GUO, X. & WANG, Y. 2021. Monitoring Public Transit Ridership Flow by Passively Sensing Wi-Fi and Bluetooth Mobile Devices. *IEEE Internet of Things Journal,* 8, 474-486.

RADOVAN, A., MRŠIĆ, L., ĐAMBIĆ, G. & MIHALJEVIĆ, B. 2024. A Review of Passenger Counting in Public Transport Concepts with Solution Proposal Based on Image Processing and Machine Learning. *Eng,* 5, 3284-3315.

SAUNIER, N. Automated Road Safety Analysis Using Video Data. 2007.

SHARIFI, E., HAMEDI, M., HAGHANI, A. & SADRSADAT, H. 2010. Vehicle Detection Rate for Blue-tooth Travel Time Sensors: A Case Study in Maryland and Delaware. *89th Annual Meeting of the Transportation Research Board*. Washington, D.C.

SIPETAS, C., KEKLIKOGLOU, A. & GONZALES, E. J. 2020. Estimation of left behind subway passengers through archived data and video image processing. *Transp Res Part C Emerg Technol,* 118, 102727.

SPIESS, H. & FLORIAN, M. 1989. Optimal strategies: A new assignment model for transit networks. *Transportation Research Part B: Methodological,* 23, 83-102.

SUTHEERAKUL, C., KRONPRASERT, N., KAEWMORACHAROEN, M. & PICHAYAPAN, P. 2017. Application of Unmanned Aerial Vehicles to Pedestrian Traffic Monitoring and Management for Shopping Streets. *Transportation Research Procedia,* 25, 1717-1734.

TAYEB, F., CHIHAOUI, H. & FILALI, F. 2023. Bluetooth-Based Vehicle Counting: Bridging the Gap to Ground-Truth With Machine Learning. *IEEE Access,* 11, 64600-64607.

TRANSPORT FOR LONDON. 2017. *Review of the TfL WiFi Pilot* [Online]. Available: https://content.tfl.gov.uk/review-tfl-wifi-pilot.pdf [Accessed].

Appendix A – Combination Results for Scenario 1 (Random Forest)

Stop 1 Results

Stop	Variables_Used	Bluetooth_Used	MSE	MAE	RMSE	**R2**	SMAPE
Stop 1	('Time_Until_Departure_Ferry', 'Drone_Exit_Gate_Stop 1')	smoothed_cnt_mean	1.87	0.92	1.37	**0.99**	48.45
Stop 1	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 1', 'Drone_Exit_Gate_Stop 1')	cnt_mean	2.24	0.97	1.50	**0.98**	49.06
Stop 1	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 1', 'Drone_Exit_Gate_Stop 1')	smoothed_cnt_mean	2.27	1.01	1.51	**0.98**	51.90
Stop 1	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 1', 'Drone_Exit_Gate_Stop 1')	smoothed_cnt_mean	2.30	1.03	1.52	**0.98**	52.50
Stop 1	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 1', 'Drone_Exit_Gate_Stop 1')	cnt_mean	2.37	1.01	1.54	**0.98**	49.59
Stop 1	('Time_Until_Departure_Ferry', 'Drone_Exit_Gate_Stop 1')	cnt_mean	2.57	1.05	1.60	**0.98**	49.87
Stop 1	('Time_Since_Last_Departure_Stop 1', 'Drone_Exit_Gate_Stop 1')	cnt_mean	3.06	1.14	1.75	**0.98**	50.37
Stop 1	('Time_Until_Next_Departure_Stop 1', 'Drone_Exit_Gate_Stop 1')	smoothed_cnt_mean	3.25	1.24	1.80	**0.98**	56.74
Stop 1	('Time_Since_Last_Departure_Stop 1', 'Drone_Exit_Gate_Stop 1')	smoothed_cnt_mean	3.28	1.26	1.81	**0.98**	57.33
Stop 1	('Time_Until_Next_Departure_Stop 1', 'Drone_Exit_Gate_Stop 1')	cnt_mean	3.45	1.21	1.86	**0.98**	52.58
Stop 1	('Drone_Exit_Gate_Stop 1',)	smoothed_cnt_mean	4.44	1.51	2.11	**0.97**	63.49
Stop 1	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 1')	smoothed_cnt_mean	4.97	1.08	2.23	**0.96**	51.03
Stop 1	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 1')	smoothed_cnt_mean	6.37	1.21	2.52	**0.96**	51.06
Stop 1	('Drone_Exit_Gate_Stop 1',)	cnt_mean	6.43	1.61	2.54	**0.95**	58.36
Stop 1	('Time_Until_Departure_Ferry',)	smoothed_cnt_mean	7.95	1.18	2.82	**0.94**	47.46

Stop	Variables_Used	Bluetooth_Used	MSE	MAE	RMSE	**R2**	SMAPE
Stop 1	('Time_Until_Next_Departure_Stop 1',)	smoothed_cnt_mean	9.98	2.17	3.16	**0.93**	76.79
Stop 1	('Time_Since_Last_Departure_Stop 1',)	smoothed_cnt_mean	10.22	2.24	3.20	**0.93**	79.74
Stop 1		smoothed_cnt_mean	17.76	3.00	4.21	**0.87**	88.43
Stop 1	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 1')	cnt_mean	23.56	1.55	4.85	0.83	52.04
Stop 1	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 1')	cnt_mean	25.52	1.64	5.05	0.82	53.61
Stop 1	('Time_Since_Last_Departure_Stop 1',)	cnt_mean	25.80	2.02	5.08	0.82	61.67
Stop 1	('Time_Until_Departure_Ferry',)	cnt_mean	28.25	1.64	5.32	0.80	48.19
Stop 1	('Time_Until_Next_Departure_Stop 1',)	cnt_mean	31.16	2.23	5.58	0.78	64.37
Stop 1		cnt_mean	72.48	4.08	8.51	0.49	75.40

Stop 2 Results

Stop	Variables_Used	Bluetooth_Used	MSE	MAE	RMSE	**R2**	SMAPE
Stop 2	('Time_Until_Departure_Ferry', 'Drone_Exit_Gate_Stop 2')	smoothed_cnt_mean	0.10	0.21	0.31	**0.96**	85.62
Stop 2	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 2', 'Drone_Exit_Gate_Stop 2')	smoothed_cnt_mean	0.10	0.23	0.32	**0.95**	85.90
Stop 2	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 2', 'Drone_Exit_Gate_Stop 2')	smoothed_cnt_mean	0.10	0.23	0.32	**0.95**	85.84
Stop 2	('Time_Until_Departure_Ferry', 'Drone_Exit_Gate_Stop 2')	cnt_mean	0.11	0.23	0.34	**0.95**	86.69
Stop 2	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 2', 'Drone_Exit_Gate_Stop 2')	cnt_mean	0.12	0.24	0.34	**0.95**	86.78
Stop 2	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 2', 'Drone_Exit_Gate_Stop 2')	cnt_mean	0.12	0.24	0.35	**0.94**	86.53
Stop 2	('Time_Until_Departure_Ferry',)	smoothed_cnt_mean	0.22	0.25	0.47	**0.90**	40.28
Stop 2	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 2')	smoothed_cnt_mean	0.22	0.28	0.47	**0.90**	61.04
Stop 2	('Time_Until_Departure_Ferry',)	cnt_mean	0.24	0.30	0.49	**0.89**	54.38
Stop 2	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 2')	smoothed_cnt_mean	0.26	0.31	0.51	**0.88**	82.21
Stop 2	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 2')	cnt_mean	0.26	0.32	0.51	**0.88**	86.02
Stop 2	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 2')	cnt_mean	0.28	0.33	0.53	**0.87**	86.33
Stop 2	('Time_Since_Last_Departure_Stop 2', 'Drone_Exit_Gate_Stop 2')	cnt_mean	0.50	0.46	0.70	**0.77**	90.69
Stop 2	('Time_Until_Next_Departure_Stop 2', 'Drone_Exit_Gate_Stop 2')	cnt_mean	0.59	0.55	0.77	**0.73**	91.21
Stop 2	('Time_Since_Last_Departure_Stop 2', 'Drone_Exit_Gate_Stop 2')	smoothed_cnt_mean	0.66	0.49	0.81	**0.70**	89.77
Stop 2	('Time_Until_Next_Departure_Stop 2', 'Drone_Exit_Gate_Stop 2')	smoothed_cnt_mean	0.70	0.56	0.84	**0.68**	89.93

Stop	Variables_Used	Bluetooth_Used	MSE	MAE	RMSE	**R2**	SMAPE
Stop 2	('Drone_Exit_Gate_Stop 2',)	cnt_mean	0.84	0.54	0.92	**0.61**	90.93
Stop 2	('Drone_Exit_Gate_Stop 2',)	smoothed_cnt_mean	0.93	0.52	0.96	**0.57**	83.20
Stop 2	('Time_Since_Last_Departure_Stop 2',)	cnt_mean	1.01	0.72	1.00	**0.54**	95.19
Stop 2	('Time_Since_Last_Departure_Stop 2',)	smoothed_cnt_mean	1.13	0.70	1.06	**0.48**	94.53
Stop 2	('Time_Until_Next_Departure_Stop 2',)	cnt_mean	1.41	0.89	1.19	**0.36**	97.70
Stop 2	('Time_Until_Next_Departure_Stop 2',)	smoothed_cnt_mean	1.80	0.93	1.34	**0.17**	99.82
Stop 2		cnt_mean	2.65	1.24	1.63	**-0.21**	96.11
Stop 2		smoothed_cnt_mean	3.89	1.36	1.97	**-0.78**	104.62

Appendix B – Combination Results for Scenario 2 (XGBoost)

Stop 1 Results

Stop	Variables_Used	Bluetooth_Used	MSE	MAE	RMSE	**R2**	SMAPE
Stop 1	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 1')	smoothed_cnt_mean	7.03	1.65	2.65	**0.76**	126.06
Stop 1	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 1')	smoothed_cnt_mean	7.39	1.70	2.72	**0.75**	126.00
Stop 1	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 1')	cnt_mean	8.80	1.79	2.97	**0.70**	128.74
Stop 1	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 1')	cnt_mean	9.14	1.84	3.02	**0.69**	128.67
Stop 1	('Time_Until_Departure_Ferry',)	smoothed_cnt_mean	14.16	2.32	3.76	**0.51**	137.29
Stop 1	('Time_Since_Last_Departure_Stop 1',)	smoothed_cnt_mean	15.55	2.58	3.94	**0.47**	133.18
Stop 1	('Time_Until_Departure_Ferry',)	cnt_mean	15.55	2.51	3.94	**0.47**	137.87
Stop 1	('Time_Until_Next_Departure_Stop 1',)	smoothed_cnt_mean	16.20	2.60	4.02	**0.44**	131.92
Stop 1		smoothed_cnt_mean	17.54	2.56	4.19	**0.40**	129.54
Stop 1	('Time_Since_Last_Departure_Stop 1',)	cnt_mean	19.70	2.81	4.44	**0.32**	136.46
Stop 1	('Time_Until_Next_Departure_Stop 1',)	cnt_mean	20.42	2.80	4.52	**0.30**	133.77
Stop 1		cnt_mean	20.52	2.76	4.53	**0.29**	130.14

Stop 2 Results

Stop	Variables_Used	Target_Used	MSE	MAE	RMSE	**R2**	SMAPE
Stop 2	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 2')	cnt_mean	39.16	3.49	6.26	**0.85**	113.18
Stop 2	('Time_Until_Departure_Ferry', 'Time_Since_Last_Departure_Stop 2')	smoothed_cnt_mean	48.88	3.78	6.99	**0.81**	112.60
Stop 2	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 2')	cnt_mean	49.04	3.91	7.00	**0.81**	115.65
Stop 2	('Time_Until_Departure_Ferry', 'Time_Until_Next_Departure_Stop 2')	smoothed_cnt_mean	58.61	4.09	7.66	**0.78**	113.90
Stop 2	('Time_Until_Departure_Ferry',)	cnt_mean	83.60	5.34	9.14	**0.68**	128.24
Stop 2	('Time_Until_Next_Departure_Stop 2',)	cnt_mean	90.19	5.63	9.50	**0.66**	128.76
Stop 2	('Time_Since_Last_Departure_Stop 2',)	cnt_mean	92.10	5.79	9.60	**0.65**	127.89
Stop 2	('Time_Since_Last_Departure_Stop 2',)	smoothed_cnt_mean	102.25	6.24	10.11	**0.61**	127.69
Stop 2		cnt_mean	104.58	5.81	10.23	**0.60**	115.04
Stop 2	('Time_Until_Departure_Ferry',)	smoothed_cnt_mean	105.01	5.86	10.25	**0.60**	127.22
Stop 2	('Time_Until_Next_Departure_Stop 2',)	smoothed_cnt_mean	108.12	5.97	10.40	**0.59**	124.03
Stop 2		smoothed_cnt_mean	117.07	5.93	10.82	**0.56**	115.48