Advances in Intelligent Traffic and Transportation Systems
M. Shafik (Ed.)

© 2023 The authors and IOS Press.
This article is published online with Open Access by IOS Press and distributed under the terms of the Creative Commons Attribution Non-Commercial License 4.0 (CC BY-NC 4.0).
doi:10.3233/ATDE230009

Distributed Vision-Based Passenger Flow Monitoring System for Light Rail Networks

Trung HOANG, Donato Di PAOLA[1] and Anthony OHAZULIKE
Autonomy and Circularity Laboratory (ACL), Hitachi Europe Research & Development, France

Abstract. In this paper, we present an innovative approach to passenger flow monitoring for light rail transportation networks. We propose a distributed system based on two main concepts. On each vehicle in the transportation network, a set of sensors is used to count people at a given place. On a cloud-based server, a data synchronization and storage system aggregates the data sent from all vehicles and provides a global view of the transportation network. The contribution, with respect to the state of the art, of our approach is twofold. First, the proposed distributed architecture is able to reduce the system global cost via its flexibility and ease of deployment, since the main part of the system is onboard each vehicle and not fixed at stations or track sections. Second, the novel vision-based passenger counting approach guarantees high levels of reliability in the estimation of the number of people in a given area, and the ability to provide real-time data on the global transportation network. Experimental results demonstrate the validity and the advantages of the proposed approach, paving way to future uses of the system as the base of additional network optimization modules for the global light rail transportation.

Keywords. passenger counting, distributed systems, computer vision, railway transportation

1. Introduction

Sustainability of public transportation systems is one of the core challenges in the development of modern cities. Among all the different forms of transportation modes, light rail transportation (such as urban trains, trams, metro, etc.) have played a central role in the transportation urban context. The main factor in the sustainability of a light rail transportation system lays in its efficiency. The global efficiency of a transportation system can be improved via the optimization of several parameters, both on a single vehicle and on global network operations at large. In this context, the application of modern optimization methodologies and artificial intelligence techniques has recently gained great attention [1]. For instance, schedule optimization [2], minimization of time delays [3], prediction of vehicles passenger load [4], can be obtained by leveraging optimization techniques based on the data related to the passenger flow in the transportation network. In fact, the monitoring of the passenger flow is the core element in the optimization of the global transportation network [5].

[1] Corresponding Author, Donato Di PAOLA, IOS Press, Autonomy and Circularity Laboratory (ACL), Hitachi Europe Research & Development, France; E-mail: donato.di-paola@hitachi-eu.com.

As a consequence, different new technologies have been recently exploited for the automation of the passenger flow monitoring. On one hand, localization capabilities of daily use devices, like smartphones, can be used to estimate the number of passengers at stations or in the vehicles. In this context, recent research investigated the use of Wi-Fi technologies to extract information on public transportation systems [6], and in particular to count people at different locations (e.g., at stations, inside and outside the vehicle, etc.) [7]. On another hand, the use of sensor technology is playing a important role for passenger flow monitoring. Different types of sensors are used to extract data to determine the number of passengers at specific locations. Cameras inside vehicles can be used to count passenger arrivals and departures in real time [8], 3D LIDARS mounted on the doorways of trains can be used to count boarding and alighting passengers via neural algorithms [9], and combination of different sensors, such as cameras, floor-based sensing, and infrared sensing, with Wi-Fi technologies can provide robust automatic passenger counting results under different operating conditions [10]. In this research area on passenger flow monitoring, the vision-based methodologies are emerging as the most promising. This is mainly due to the low cost of devices and the robustness of these methods. Recent results demonstrated the effectiveness of the utilization of existing onboard surveillance cameras to count passengers, inside the vehicle, in cluttered situations [11]. Others methods are used to determine the number of people at given locations, by using fixed cameras. In [12], a vision based solution to detect crowdedness in public transport stations is presented. Furthermore, the use of fixed surveillance cameras to extract passenger data, combined with the cost theory and automatic frequency control, is providing prediction about the passenger flow at stations [13].

In this context, the key contribution of this paper is a novel approach to the passenger flow monitoring. In contrast to most latest passenger flow monitoring systems, which rely on fixed on-premise infrastructures, we propose a distributed system based on onboard sensors, which are installed on the rail vehicle. Moreover, differently from aforementioned systems that use onboard sensors (like cameras or LIDARs) mounted inside the vehicle or on the doors, we use external cameras (most of the times already present on the vehicles for security or driver assistance tasks). Thanks to the external cameras we can observe the environment around the vehicle, whether it is in motion or stopped at stations. The advantage of our approach is twofold. First, the global cost is reduced, since the hardware is installed on each vehicle and not at each station or track section we want to monitor. Second, our approach is flexible and easy-to-deploy. Since the passenger flow monitoring system is moving with the vehicle, we can easily configure the system to add a new stop or a new location of interest to monitor (i.e. intersections, roundabouts, etc.), without requiring additional hardware or working time for a new sensor or hardware installation. Furthermore, the innovative system that collects and synchronizes data, from each vehicle to a cloud-based central system, provides a real-time view of the passenger flow on the global transportation network that can be accessed simultaneously from different passenger flow monitoring front-end interfaces, hence increasing the scalability.

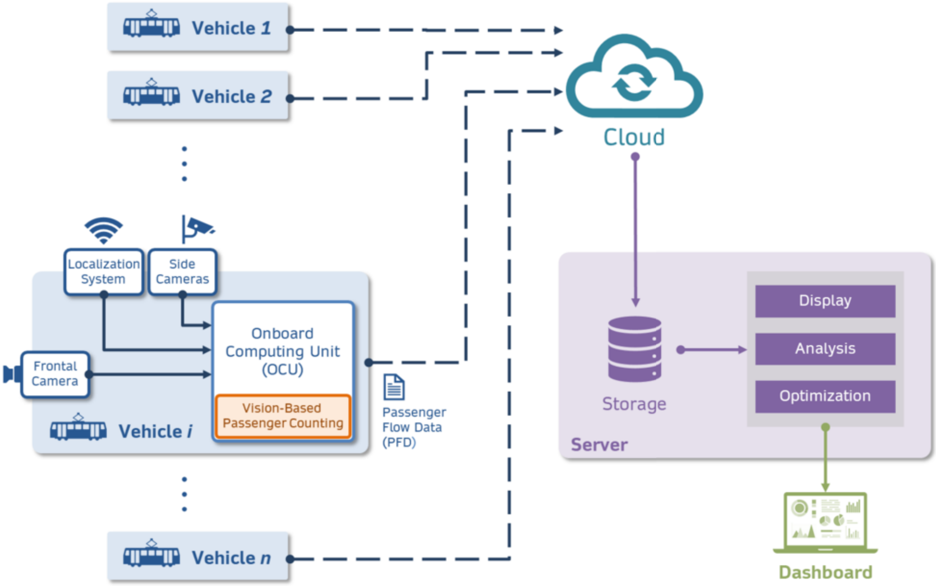


Figure 1. Diagram of the overall system architecture. On the left side, the network of vehicles, which communicates with the cloud-based system, is depicted. In particular, for the generic vechicle *i*, the details of the sensor set and the Onboard Computing Unit (OCU) are shown. The rigth side depicts the synchronization and storage system, which provides services to display, analyse and optimize the global Passenger Flow Data (PFD) messages obtained over time. All these services are accessible via a web-based Dashboard.

2. System Overview

The proposed distributed passenger flow monitoring system is composed of two main parts: the vision-based passenger counting system and the cloud-based data synchronization and storage system (see Figure 1).

The data acquisition and the passenger flow processing are done onboard the vehicle. In particular, the onboard system consists of a sensor set for data acquisition and an Onboard Computing Unit (OCU) for data processing and temporary storage. The onboard sensor set is composed of a set of monocular cameras, to acquire images of the scene around the vehicle, and a localization system that provides the accurate position of the vehicle over time. The latter system can be implemented by using a GNSS (Global Navigation Satellite System) unit in combination with inertial sensors, odometry data provided by the internal system of the vehicle, and vision-based motion estimation systems. The cameras composing the set of monocular cameras are calibrated (both intrinsically and extrinsically), thereby the mutual position of one camera with respect to the others and to the vehicle reference frame is known. In this way, detected objects (passengers) can be localized in the space around the vehicle. Moreover, a vehicle database, located in the persistent storage of the onboard processing unit, provides the key locations (e.g., at platform/station) where the passenger flow data must be acquired. Thus, when the onboard system is running, key locations stored in the vehicle database are used to trigger the passenger flow monitoring system. The result of this process, the Passenger Flow Data (PFD), namely the vehicle location, the passenger data (number of passengers in the area, at the station platform, getting on the vehicle, getting off the vehicle, etc.), and the acquisition time, are temporarily stored in the vehicle database. The detailed description of the people flow monitoring algorithm will be given later in this section.

The second part of the proposed system is the cloud-based data storage and synchronization system. This system manages the local data storage onboard each vehicle and the data synchronization to the cloud-based storage server. Each newly acquired PFD are first stored in the OCU, then they are sent via the synchronization system, via an Internet connection, to the cloud-based storage server. This two-phase mechanism avoids losing data being transferred from each vehicle to the cloud-based storage server. The PFDs are securely stored onboard each vehicle then, only if there is a valid connection and the transmission process is completed correctly, the passenger data are flagged as transmitted, otherwise the synchronization system will retry the transmission when the connection becomes available again. All the PRDs, coming from all the vehicles in the network, are stored in a database in the cloud-based storage system and updated on-line, as described above. Thus, the full view of passenger flow for the entire transportation network is provided by this part of the system. This view on the global transportation network can be accessed via a dashboard. In particular, the dashboard visualizes the current situation of the network via the on-line updates of passenger data, as well as the historical data with insights based on data analytic techniques. Furthermore, the cloud-based data storage and analysis system enables data-driven optimization tasks for the global transportation network and the city planning, such as timetable optimization, traffic lights synchronization, energy usage optimization, etc.

3. Vision-Based Passenger Counting

In this section the vision-based process, distributed across the vehicles in the transportation network, to count passengers is described. The onboard process related to the computation of the Passenger Flow Data (PFD) can be summarized as follows:

Step 1: The onboard localization system provides the vehicle global position over time. Thus, the vehicle position is compared, at a given rate (e.g. 20 Hz), to the collection of key locations (e.g. stations) stored in the vehicle database. If the vehicle is entering a location of interest then the vision-based passenger counting algorithm is triggered.

Step 2: The vision-based passenger counting algorithm receives the stream of images from the onboard cameras and it performs the counting of people by using a unique method described later in this section.

Step 3: The global position of the vehicle is then used to stop the passenger counting algorithm. If the vehicle reaches a specific location (e.g. the end of the stop platform, and other conditions are satisfied such as when the vehicle velocity approaches zero, the passenger counting algorithm is terminated).

Step 4: The result of the vision-based passenger counting algorithm is stored as Passenger Flow Data (PFD) in the local vehicle database for further synchronization with the global cloud-based system.

The core factor of the aforementioned process is the vision-based passenger counting algorithm mentioned in step 2. In fact, to obtain accurate and reliable information on the global flow of passenger in the transportation network, the performances of the counting system at each location are of essential importance. Thus, to design a good vision-based solution for passenger counting we considered the following assumptions:

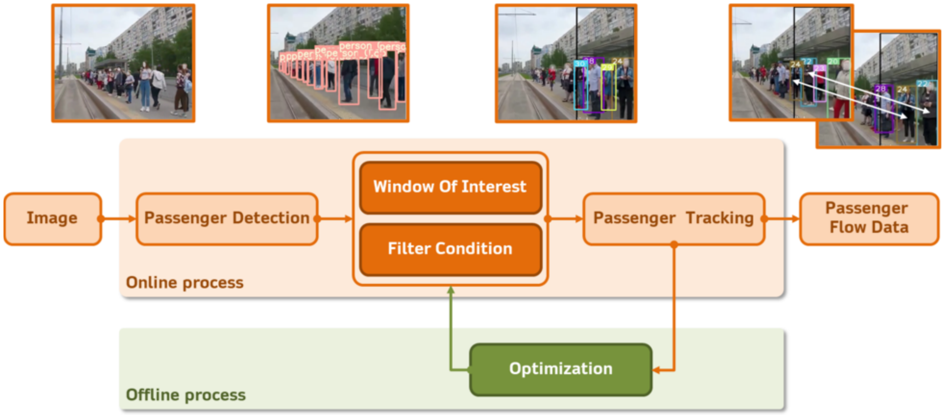


Figure 2. Overview of the detection and tracking pipeline. The Online process is depicted with its three steps, from the input image to the final resul as Passenger Flow Data (PFD). In correspodence of each step, on top of the diagram, the result is shown on an example image. Moreover, the Offline process is shown as the optimization step tacking the results of the Tracking Algorithm to better tune the Window of Interest (WOI) and the Filter Condition (FC).

- The speed of the vehicle approaching a station, is much higher than walking passenger's speed, though it still follows linear direction.
- A typical passenger will only appear in some frames then he/she will disappear as the vehicle passes by, and will not be re-identified afterward.
- There are certain areas of interest where we want to count passenger (e.g. the right-hand-side platform when the vehicle arrives at the station), and not others (e.g. passenger on the opposite direction, i.e. on the left-hand-side platform).
- The moment when we start and stop counting should be based on specific vehicle positions to increase the precision in the passenger counting process. As mentioned in the step 3, this can be done using vehicle localization system and other vehicle's information, such as the vehicle speed.
- There are passengers who approach the station later than the vehicle approaches the platform and others who leave the station when the vehicle is not stopped yet. Though these situations are not considered in this paper, further research can investigate how to detect and incorporate this information into counting passenger systems.

Following the aforementioned aspects, considered as design constraints, we developed the vision-based passenger counting algorithm, whose explicative diagram is showed in Figure 2.

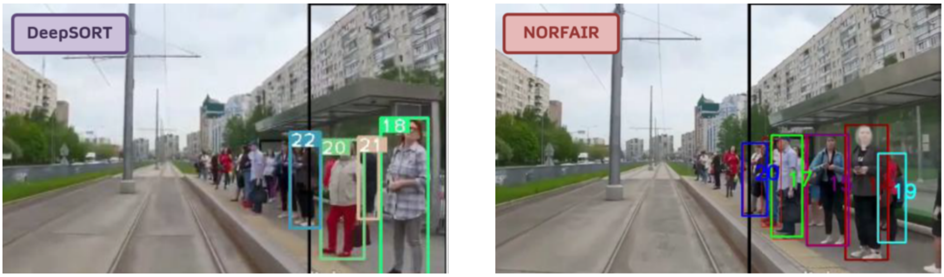
As already mentioned above, the algorithm is triggered when the train is approaching the station (or a location of interest). Thus, the image stream from the set of onboard cameras, pointing at the platform and at the people standing, is sent to the passenger counting module (e.g., 30 images per second, i.e. 30 Hz). Then, the first phase of the algorithm, *Passenger Detection*, is performed. In this phase, a generic detection algorithm is applied on each image, to identify people and give the position and size of each person in the image. The result is a set of bounding boxes (the pink rectangles around people in the second image from the left of Figure 2) identifying the area covered by each person on each image. In order to obtain a more robust detection, we introduced the concepts of

Figure 3. Example of intersections between detected bounding boxes and the Windows Of Interest (WOI). The WOI is highligheted as an black rectangle on the right side of the image. We can notice that the detection 22 is not completly included in the WOI area. Thus, the Intersection Over Union (IOU) concept will be used to determine if the detection 22 can be considered or not as a valid detection.

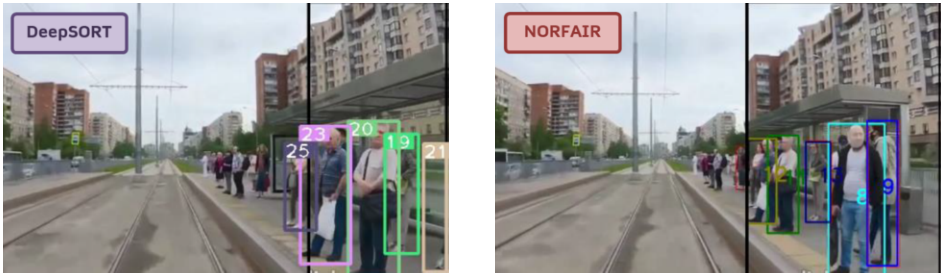
Window Of Interest (WOI) and *Filter Condition* (FC). Basically, on each image, the WOI and the FC filter out unwanted detected people based on segments covered by the WOI, the sizes of the bounding boxes, and the intersection between WOI and FC. If there is no such filtering, it often ends up with the algorithm detecting several bounding boxes, including small ones around people standing at the very end of the platform. This would result in wrong re-identification in subsequent steps of the tracking algorithm. By WOI and FC, we effectively limit the vision field of view of the frontal camera, removing the possibility to identify too early people standing at the end of the platform. Hence the algorithm will not consider these people early on when the train begins to approach the platform. These people however are naturally considered as the train moves towards the end of the platform once they satisfy the conditions of the filter.

The motivation behind the introduction of the WOI and the FC is based to the analysis of the key issues of common vision-based people detection and tracking systems. In general, the majority of people detection and tracking systems at stations (either inside or outside) assume that the camera is at fixed position. In our case, if we apply the common detection and tracking systems used for fixed cameras, the whole scene will be exposed to detection algorithm and new tracking identifiers will be assigned to detected people. This result is correct if people are closer to the camera, i.e. people on the platform closer to the vehicle. However, this result might not be desirable as the algorithm might only detected a small part of human body from afar, which cause the tracking algorithm to wrongly re-identify people. As a consequence, we propose the use of the WOI to focalize the detection only on the image area where passenger detection is more accurate, i.e. the detection confidence value is higher than a certain threshold, and the use of the FC, based on the concept of Intersection Over Union (IOU), to compare the area of each passenger bounding box with the area of the WOI (see Figure 3).

As a final phase, the set of detected bounding boxes, the result of the previous filtering stage, is sent as input of the *Passenger Tracking* algorithm, to assign unique identifiers and associate bounding boxes of detected people on each image over time. There-



(a) Station 1: The actual number of passengers is 42, our system using the Tracking Algorithm with DeepSORT predicted 40 people (left), while with the NORFAIR predicted 40 people (right).



(b) Station 2: The actual number of passengers is 28, our system using the Tracking Algorithm with DeepSORT predicted 28 people (left), while with the NORFAIR predicted 26 people (right).

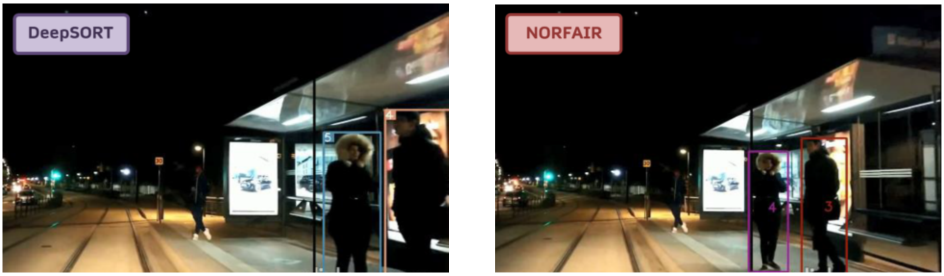
Figure 4. Comparison between our system using DeepSORT and NORFAIR as Passanger Tracking algorithms for the St.Petersburg tram dataset. Note that the Passenger Detection phase is perfromed using YOLO-v5 in both implementations shown here.

fore, we can count the total number of people standing on the platform by counting the unique identifiers among all image frames during the time the vision-based passenger counting algorithm has been executed.

As an offline step, we carried out an optimization process of the whole vision-based passenger counting algorithm. On one hand we optimized the hyperparameters within the machine learning models used in the detection and tracking phases (via a fine tuning process by controlling how many detected people they should keep, on how many frames, etc.). On the other hand, we also optimized the WOI and FC parameters. Since the most effective area of the WOI and the best FC in which only accurate bounding boxes are detected, are not known *a priori*, an empirical procedure has to be performed. We run various experiments with different datasets and, using heuristic grid search, we fine tune both WOI and FC parameters by comparing the algorithm results with the ground truth in several scenarios (various stations at different weather and lighting conditions, i.e. day-time, night-time, rain, snow, etc.).

4. Results

In this section we present experimental results, by using publicly available datasets and with simulated data, to demonstrate the feasibility and the advantages of the proposed



(a) Station 1 (Night): The actual number of passengers is 6, our system using the Tracking Algorithm with DeepSORT predicted 6 people (left), while with the NORFAIR predicted 5 people (right).

(b) Station 2 (Snow): The actual number of passengers is 7, our system using the Tracking Algorithm with DeepSORT predicted 8 people (left), while with the NORFAIR predicted 8 people (right).

Figure 5. Comparison between our system using DeepSORT and NORFAIR as Passanger Tracking algorithms for the Aubagne tram dataset. Note that the Passenger Detection phase is perfromed using YOLO-v5 in both implementations shown here.

system. In particular, first we discuss the results on the vision-based passenger counting system, then we analyze the result of the cloud-based data synchronization and storage system for the global light rail transportation network.

4.1. Vision-Based Passenger Counting System

As mentioned in the previous section, the phases of *Passenger Detection* and *Passenger Tracking* of the vision-based passenger counting algorithm can be implemented with generic detection and tracking algorithms. In the implementation described in this paper we chose the object detection algorithm YOLO-v5 [14] as Passenger Detection algorithm. The You Only Look Once (YOLO) family of algorithms (there are plenty of versions and variants) are object detection algorithms based on a neural network model to perform the supervised detection and classification of known objects. The advantage of YOLO algorithms lies in the model's small size and fast calculation speed. The methodology can directly output the position and class of the bounding box through the neural network. Moreover, YOLO has a strong generalization ability due to the learning phase of highly generalized features, then it can be successfully used with datasets similar to the one on which it has been trained. On the other hand, for the passenger tracking phase we compared two different algorithms. The rationale for this comparison was to find the best match between the tracking algorithm and the novel filtering phase introduced in our

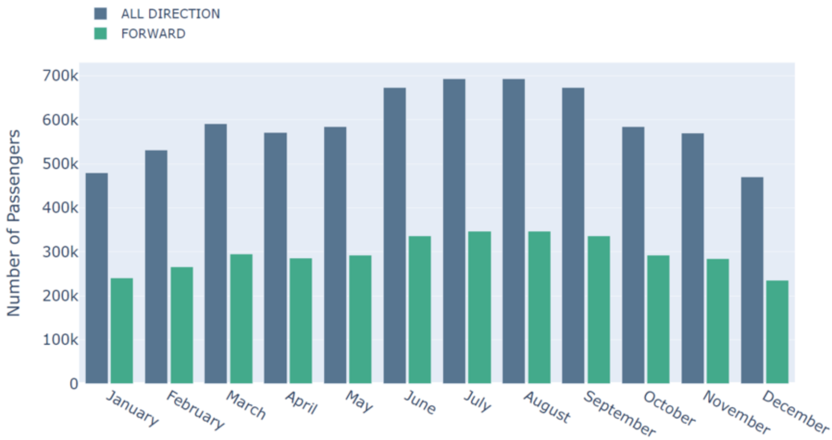
Table 1. Empirical Results from Testing Videos

Test data	Default Settings		Optimized Settings		Ground Truth
	DeepSORT 0.5	NORFAIR 0.5	DeepSORT 0.25	NORFAIR 0.35	
St. Petersburg 01	22	**40**	**<u>40</u>**	**40**	**42**
St. Petersburg 02	16	26	**<u>28</u>**	26	**28**
Aubagne (night) 01	**6**	5	**<u>6</u>**	5	**6**
Aubagne (snow) 02	10	**8**	**<u>8</u>**	**8**	**7**

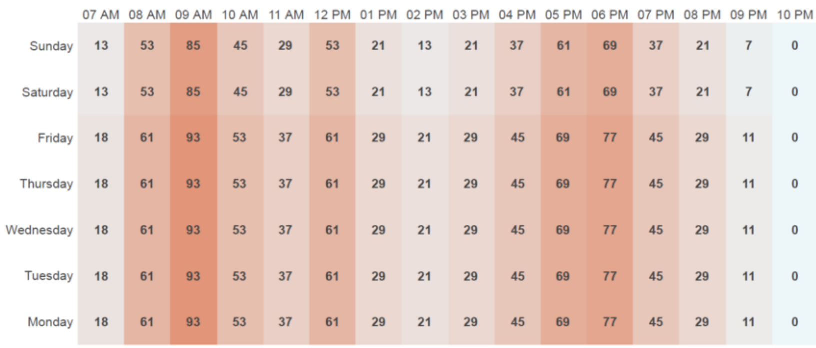
vision-based passenger counting algorithm. The two algorithms analyzed are the DeepSORT [15] algorithm and the NORFAIR [16] algorithm. The DeepSORT algorithm is designed to overcome the classical limitations of the object tracking algorithms by using a machine learning model. Usually, object tracking algorithms can assign an identifier and track an object by mapping bounding boxes of similar size and motion parameters across images frames. However, this process presents the re-identification limitation: if a previously detected object is hidden behind another object and then it reappears, it will probably be assigned to a new identifier if the projected trajectory is incorrect. DeepSORT solves this problem by using a machine learning model that compares similarity between people (via learned features from a trained model on a dateset of millions of people), thus reducing the issue of switching peoples' identifier. Similarly, the NORFAIR algorithm solves the problem of re-identification by estimating the future position of each point based on its past positions. It then tries to match these estimated positions with newly detected points provided by the detector. For this matching to occur, NORFAIR can rely on any distance function (in the present implementation we used the the Euclidean distance between tracked objects and detections). As mentioned above, we used both of the tracking algorithms after the filtering phase. Moreover, starting from default values for the WOI and FC parameters (e.g. 0.5 or 50% of the frame width as we are only concerned with the right hand side platform), we optimized the shape of the WOI and FC parameters for each of the two implementations. In this way we obtained a robust detection and tracking process for the whole vision-based passenger counting algorithm by experimenting with different lighting and weather conditions, i.e. day time, night time, and snowy weather.

We currently have no comparative and limited datasets to obtain the benchmark. Therefore we chose 4 videos of different settings with various circumstances based on weather, lighting and crowdedness. The result of some scenarios for the two different setups, one based on the DeepSORT and one on NORFAIR, are shown in Figure 4 and Figure 5. It is straightforward to note that globally the setup with DeepSORT performs better than the one with NORFAIR. This is especially valid on day-time and clear weather condition scenarios. Since this is the most common situation and the DeepSORT performances in more challenging scenarios (night and snow) are comparably with NORFAIR, we select the setup with DeepSORT as the final implementation of the proposed system.

On a more detailed results shown in Table 1, we have compared between two settings: default and optimized ones. In default settings, out-of-the-box parameters are assumed and the results are obtained directed from four testing videos. Note that the fraction beside algorithm name (e.g. DeepSORT 0.5) is the ratio of the window of interest against the total width of the frame. In the optimized settings, we have tuning hyperparameters for each algorithm with certain settings of the window of interests. The results highlighted in bold are closest counting to the ground truth. We can see clearly the



(a) Number of passengers over months for one year, for Line 1.



(b) Number of passengers over a selected period and at a given station, for Line 1.

Figure 6. View on aggregated passenger flow data from all the vehicles of the tram network over time.

improvement of optimized DeepSORT at window of interest's width equal 25% of the frame's total width.

4.2. Cloud-Based Data Synchronization and Storage System

In order to show the usefulness and the advantages of the cloud-based solution to synchronize and aggregate data from each vehicle in the transportation network, we developed a simulator for tramway networks. This choice is motivated by the lack of publicly available datasets, for light rail networks, to test the proposed system. In particular, no datasets with sensor data for each vehicle are available at the time of writing.

The simulator has been designed to demonstrate the functionalities of the synchronization, aggregation and storage part of the system. Thus, the vision-based passenger counting algorithm has not been taken into account in the simulator. As a consequence, the simulator assumes that each vehicle will produce Passenger Flow Data (PFD) at each station visited along the tram line. The structure of the simulator is the following:

- A set of functions simulate the operations of the set of vehicles in the network, by using a Monte Carlo Simulation approach.
- A centralized database receives the data produced by each function, i.e. the PFD, for all the vehicles and for all the lines over time.
- A web-based dashboard, connected to the cloud database, provides interactive data visualization updated in real-time.

The simulation results presented in this paper are based on the simulation of tram network of the city of Naples, Italy. The data obtained after the simulation, extracted from the cloud-based dashboard, are reported in Figure 6. The data in the pictures are the result of the simulation on a single line (the Line 1 of the Naples tram network), with 10 stations and with trams coming across each station every 5 minutes, in average, on both ways. In particular, the first graph, in Figure 6 (a), shows statistics on overall number of passenger over months, for both line directions or for a single one. Thanks to the dynamic dashboard, the duration of aggregation can be select for a period of interest (e.g. from 1 July to 30 August, 2021). From this graph we can see the changing in trend during the summer period than can be related to the typical vacation period for the majority of people in the area. The second graph, in Figure 6 (b), shows heat map of people counting over hours in every days of week for a single station up to all the stations on the line, as shown here. From this graph we can observe the upward trend during rush hours, around 9.00 AM and around 5.00-6.00 PM.

Finally, thanks to the capabilities provided by the cloud-based data synchronization and storage system, it is straightforward to not that the Passegner Flow Data (PFD) obtained by each vehicle can be further processed to obtain the input for optimization and recommendation modules, such as time-table optimization, network capacity and frequency optimization (to match the real request), analysis of connections between the light rail network and other transportation networks (e.g. bus network, vehicle sharing platforms, etc.).

5. Conclusion

In this paper, we presented a novel approach to passenger flow monitoring for light rail transportation networks. We proposed a distributed system based on onboard sensors, installed on the rail vehicle, and a cloud-based data synchronization and storage system that provides a global view of the transportation network. The novelty of our approach goes in two directions. On one direction, the proposed distributed architecture is able to reduce the system global cost, with respect to fixed passenger monitoring systems, via its flexibility, and to provide a real-time view of the passenger flow on the global transportation network. On the other direction, the novel vision-based passenger counting approach guarantees high levels of reliability in the core operation of the system, i.e. the estimation of the number of people in a given area. Experimental results demonstrated the validity and the advantages of the proposed system, opening the way to the possibility to use the system as base of additional optimization modules for the global light train transportation network. Particular edge cases, related to the difficulty of solving the occlusion problem from different point of views, however are still limitations of the proposed system. Besides, determining when to start and stop counting people needs to take into account passengers coming to the station after the tram has arrived. Future work can address this shortcomings by combining vision-based system with other passengers tracking systems (e.g. ticketing systems), for better accuracy of passenger flow monitoring.

References

[1] N. Kumar, A. Mishra. Role of Artificial Intelligence in Railways: An Overview. In: R. Kumar Phanden, K. Mathiyazhagan, Ravinder Kumar, J. P. Davim, editors. Advances in Industrial and Production Engineering. 2021.

[2] W. Li, H. Xu, G. Chen, J. Qian, and X. Wen. Research on Optimization of Intelligent Public Transportation Scheduling Based on Big Data. In: Proceedings of the 2020 9th International Conference on Computing and Pattern Recognition (ICCPR 2020). Association for Computing Machinery, New York, NY, USA, 467–471. 2020.

[3] P. Zhou, L. Chen, X. Dai, B. Li and T. Chai. Intelligent Prediction of Train Delay Changes and Propagation Using RVFLNs With Improved Transfer Learning and Ensemble Learning. IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 12, pp. 7432-7444, Dec. 2021.

[4] P. Wang, X. Chen, J. Chen, M. Hua and Z. Pu. A two-stage method for bus passenger load prediction using automatic passenger counting data. IET Intelligent Transport Systems, pp. 248-260, vol. 15. 2021

[5] I. Grgurević, K. Juršić and Vinko Rajič. Review of Automatic Passenger Counting Systems in Public Urban Transport. In: L. Knapčíková, D. Peraković, A. Behúnová, M. Periša, editors. 5th EAI International Conference on Management of Manufacturing Systems. 2022

[6] A. Bánhalmi, B. Vilmos, I. Megyeri, Z. Majó-Petri and J. A. Csirik. Extracting Information from Wi-Fi Traffic on Public Transport. International Journal of Transport Development and Integration 5(1):15-27, April 2021.

[7] I. Grgurević, K. Juršić and V. Rajič. Overview of Wi-Fi-Based Automatic Passenger Counting Solutions in Public Urban Transport. In: L. Knapčíková, D. Peraković, M. Periša, M. Balog, editors. Sustainable Management of Manufacturing Systems in Industry 4.0. EAI/Springer Innovations in Communication and Computing. Springer. 2022.

[8] J. Kniess, J. C. Rutke and W. A. C. Castañeda. An IoT Transport Architecture for Passenger Counting: A Real Implementation. 2021 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 613-617, 2021,.

[9] R. Seidel, N. Jahn, S. Seo, T. Goerttler and K. Obermayer. NAPC: A Neural Algorithm for Automated Passenger Counting in Public Transport on a Privacy-Friendly Dataset. IEEE Open Journal of Intelligent Transportation Systems, vol. 3, pp. 33-44, 2022.

[10] C. Mccarthy et al. A Field Study of Internet of Things-Based Solutions for Automatic Passenger Counting. IEEE Open Journal of Intelligent Transportation Systems, vol. 2, pp. 384-401, 2021.

[11] R. Sutopo, J.MY. Lim, V.M. Baskaran, et al. Appearance-based passenger counting in cluttered scenes with lateral movement compensation. Neural Comput & Applic 33, 9891–9912. 2021.

[12] R. A. Gheorghiu, V. Iordache and V. A. Stan. Computer vision application to determine crowdedness in public transport stations. 2021 13th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), pp. 1-4, 2021.

[13] J. Guo, Z. Xie, Q. Li, S. Zhan and J. Xu. Railway Passenger Flow Recognition Algorithm for Terminal Station Based on Cost Theory and Automatic Frequency Control. IEEE Access, vol. 8, pp. 26885-26892, 2020.

[14] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma. A Review of Yolo Algorithm Developments. In: Procedia Computer Science, vol. 199, pp. 1066–1073, Jan. 2022.

[15] N. Wojke, A. Bewley and D. Paulus. Simple online and realtime tracking with a deep association metric. In: 2017 IEEE International Conference on Image Processing (ICIP), pp. 3645-3649, 2017.

[16] J. Alori, A. Descoins, K. Yuhara, D. Facundo-Lezama, B. Ríos, F. Shafu, A. Castro, and D. Huh. tryolabs/norfair: v0.4.3 (v0.4.3). Zenodo. https://doi.org/10.5281/zenodo.6471349