

CCTV-Based Passenger Tracking and Arrival Time Prediction for Subway Door Safety

Yanjing Jin

Department of Electrical and
Computer Engineering
Sungkyunkwan University
Suwon, Republic of Korea
jinyanjing@skku.edu

Heejae Kim

Department of Electrical and
Computer Engineering
Sungkyunkwan University
Suwon, Republic of Korea
heejae5334@skku.edu

Hee Jo

Department of Electrical and
Computer Engineering
Sungkyunkwan University
Suwon, Republic of Korea
ossam@skku.edu

Jo Woon Chong*

School of Electrical and
Electronic Engineering
Sungkyunkwan University
Suwon, Republic of Korea
jwchong@skku.edu

Abstract—This study proposes a CCTV-based pipeline for passenger detection, passenger tracking, and arrival time prediction at the train door to support subway door safety. Previous studies mainly estimated passenger locations and movement states from individual frames, relying on general distributions of measured passenger speed rather than measuring individual real-time speeds or trajectories to determine the remaining time required to reach the door. To address this limitation, a YOLO26n-based detector estimates passenger bounding boxes and movement classes, and descending passengers are subsequently tracked using Constraint-Guided SORT. Trajectory features extracted from the stair area are then used to predict the remaining time until door arrival. The detector achieved an overall mAP@0.5 of 0.932. For six selected passengers, the tracking method achieved an AUC of 0.796 ± 0.067 in the success plot and a Precision@20 px of 0.948 ± 0.066 . Using 44 passenger samples, ElasticNet showed the best prediction performance, with an MAE of 0.644 ± 0.021 s, an RMSE of 0.739 ± 0.027 s, and an R^2 of 0.399 ± 0.044 . These results indicate the potential of CCTV-based tracking trajectories for passenger arrival time estimation.

Index Terms—Subway Door Safety, Passenger Detection, Passenger Tracking, Trajectory Analysis, Arrival Time Prediction

I. INTRODUCTION

Door entrapment accidents on subway platforms affect passenger safety, train delays, and operational efficiency. According to the Seoul Metropolitan Government, 2,485 passengers were injured in safety accidents on Seoul subway lines between 2019 and August 2023, of which 747 cases, or 30.1%, were caused by door entrapment, making it the most common accident category [1]. Passengers descending the stairs and attempting to board shortly before door closure may face a particularly high risk. Similar accidents involving passengers caught in or dragged by train doors have also been reported in other railway systems [2].

Existing approaches to improving door safety include obstacle detection around train doors and safety management at the platform–train interface [3], [4]. CCTV-based surveillance and hazard detection have also been investigated for monitoring safety-critical environments, including railway platforms [5], [6]. However, many existing measures primarily detect passengers or obstacles after they have reached the door area or

entered a risk zone, limiting their ability to assess passengers who are still approaching from the stairs.

Recent studies have used CCTV-based deep learning to detect subway passengers and classify their movement states. Jo et al. classified passengers as ascending, descending, or passing using YOLOv7-tiny, achieving an mAP@0.5 of 91.83% and an inference time of 0.009 s/frame [7]. Their study demonstrated that passenger location and movement state can be estimated simultaneously from a single CCTV frame.

However, single-frame detection cannot capture the continuous trajectory or movement speed of the same passenger, making it difficult to determine when the passenger will reach the subway door. Door safety requires assessing not only whether a passenger is descending the stairs but also whether the passenger can reach the door before closure. Therefore, tracking is required to associate the same passenger across consecutive frames and generate a continuous trajectory [8]. This study uses these trajectories to predict the remaining time until door arrival.

In the field of pedestrian trajectory prediction, many studies have predicted future movement paths using information such as surrounding pedestrians, scene context, and destination intention [9], [10]. However, general trajectory prediction studies often focus on predicting future coordinates in complex open environments. In contrast, a subway platform has clear structural constraints, such as stairs, the platform area, and train doors, and passengers generally exhibit goal-directed movement patterns with the clear intention of boarding a train [11]. Therefore, this study focuses on predicting passenger arrival time rather than future coordinates. Because dwell time and door-closing timing are important operational factors [12], the predicted remaining time may support door-closing decisions.

The proposed pipeline consists of three stages: YOLO26n-based passenger detection, tracking of descending passengers, and remaining-time prediction using trajectory features.

The main contributions are as follows:

- 1) We extend single-frame passenger movement estimation through tracking-based trajectory analysis.
- 2) We investigate the feasibility of predicting the remaining time to the subway door from stair-area tracking trajectories

*Corresponding author: Jo Woon Chong (jwchong@skku.edu).

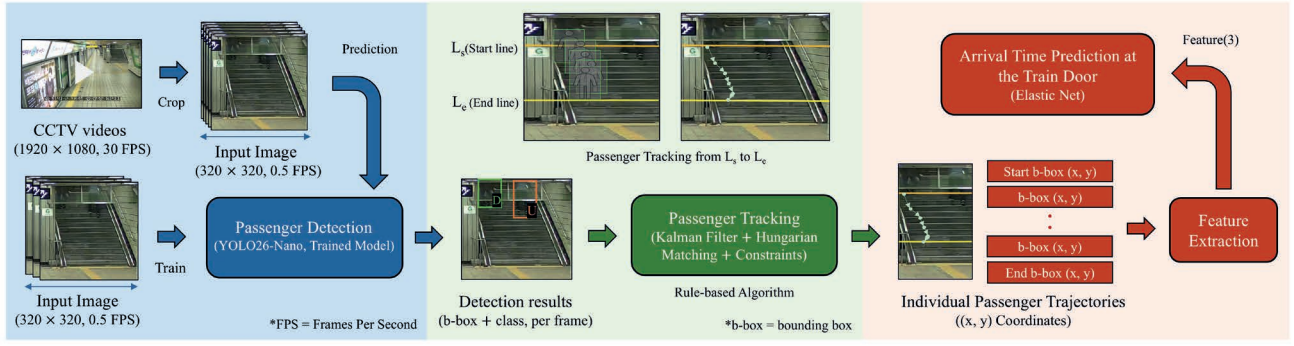


Fig. 1. Overall Pipeline of the Proposed System

to support subway door safety decisions.

3) We independently evaluate passenger detection, tracking, and arrival time prediction.

II. MATERIALS AND METHODS

A. Dataset Preparation

The three tasks of passenger detection, tracking, and arrival time prediction require different temporal resolutions and annotation types. To prevent data leakage between training and testing, the source CCTV videos were temporally divided into separate video segments before constructing the task-specific datasets. Independent test data were then constructed from designated held-out video segments, while the temporal resolution and annotations were adjusted according to the requirements of each task.

For training and evaluating the passenger detection model, we used the passenger detection dataset constructed in a previous study [7]. This dataset consists of image frames extracted every 2 seconds from 30 FPS CCTV videos. Each passenger was annotated with a bounding box and one of three movement classes: ascending, descending, or passing. A total of 4,783 images were used for model development, while 1,595 image frames from the temporally separated test portion were reserved for final evaluation. The 4,783 images were first divided into training and validation sets, and the validation set was used to determine the optimal number of training epochs. The final model was then retrained using the combined training and validation sets with the selected number of epochs. To prevent data leakage, the final model performance was evaluated only on the 1,595 images from the separated test video portion described in the Dataset Preparation subsection.

For passenger tracking, the test portion was processed at the original frame rate of 30 FPS to preserve continuous passenger trajectories. Six descending passengers were selected for tracking evaluation, and their frame-level bounding boxes were manually annotated by three experts as the tracking ground truth.

For arrival time prediction at the train door, passenger trajectories were extracted from the 30 FPS CCTV videos. A total of 44 passenger samples were used to construct the

prediction dataset, with the six passengers used for tracking evaluation assigned to the test set. For each passenger, the door arrival frame determined by three experts was compared with the last observed frame of the tracking trajectory, and the frame difference was converted into the remaining arrival time in seconds based on 30 FPS.

B. Overall Pipeline

As shown in Fig. 1, the proposed method consists of three stages: passenger detection, passenger tracking, and arrival time prediction at the train door. First, a region of interest (ROI) including the stairs and part of the platform is cropped from each CCTV frame. Then, the YOLO26n model is used to detect the bounding boxes and movement classes of passengers. Next, among the detection results, objects classified as descending are used as the input to the tracking module to generate a trajectory for each passenger. Finally, movement features are extracted from the generated trajectory, and the remaining time until the passenger reaches the subway door is predicted.

C. Passenger Detection

The passenger detection stage was used to identify descending passengers who may be at risk of approaching the subway door shortly before closure and to provide their bounding boxes as input to the subsequent tracking stage. For this purpose, we used the Ultralytics-based YOLO26n model [13]. The model was initialized with COCO-pretrained weights and fine-tuned to classify passengers into ascending, descending, and passing classes. Training and evaluation were conducted according to the data partitioning and training procedure described in the Dataset Preparation subsection. The input size was set to 320×320 , with a batch size of 32. Adam with an initial learning rate of 0.01 and a cosine learning rate schedule was used for optimization.

D. Passenger Tracking

In the passenger tracking stage, only detections classified as descending by YOLO26n were used as input. The proposed tracking method, referred to as Constraint-Guided SORT (CG-SORT), is based on the SORT framework and combines an 8-state Kalman filter with Hungarian matching [8]. In

addition, entry and exit zone constraints, appearance similarity, and physical motion constraints were incorporated to account for the characteristics of the subway stair environment. The entry and exit zone constraints restricted track initialization and termination to predefined regions. Appearance similarity compared passenger appearances using HSV color histograms. Physical motion constraints excluded implausible associations based on excessive upward movement, lateral movement, and frame-to-frame displacement.

The Kalman filter state was defined as $(c_x, c_y, w, h, v_{c_x}, v_{c_y}, v_w, v_h)$, where (c_x, c_y) denotes the geometric center of the passenger bounding box, and (w, h) denotes the bounding-box width and height. The variables v_{c_x} and v_{c_y} represent the frame-to-frame rates of change in the bounding-box center coordinates estimated by the Kalman filter, while v_w and v_h represent the estimated rates of change in the bounding-box width and height. These velocity components are internal state variables of the Kalman filter and do not directly represent the passenger's physical walking speed. The width and height components were included to accommodate temporal variations in the size and shape of the detected bounding box.

In each frame, the Kalman filter predicts the bounding boxes of existing tracks. The predicted tracks and new detections are then associated using Hungarian matching based on an association cost combining Intersection over Union (IoU), center distance, and HSV histogram similarity. A track-detection pair is assigned a lower association cost when the predicted and detected bounding boxes have greater overlap, a shorter distance between their geometric centers, and more similar HSV color histograms. Hungarian matching then determines the set of track-detection assignments that minimizes the total association cost.

Several rule-based constraints were applied during the association process. As shown in Fig. 1, L_s and L_e denote the start and end lines, respectively, and the region between them corresponds to the passenger tracking zone. First, a new track ID was created only when the center of an unmatched detection was located above L_s , which served as the entry boundary. This rule assumes that each target passenger first becomes visible near the top of the tracking region; passengers who initially appear in the middle of the tracking region due to occlusion were not considered in the current study. Second, candidate matches were rejected when the center displacement exceeded the predefined maximum distance or speed. Here, the center displacement was defined as the two-dimensional Euclidean pixel distance between the center of the Kalman-predicted bounding box and that of the candidate detection. The allowable distance increased by 8 pixels for each tracking update since the last match, but was limited to a maximum of 100 pixels. Third, candidate matches showing movement inconsistent with the expected downward motion were rejected, including excessive upward displacement from the predicted position and excessive horizontal displacement. Fourth, abrupt changes in bounding-box width or height were restricted. In addition, HSV color histogram similarity was

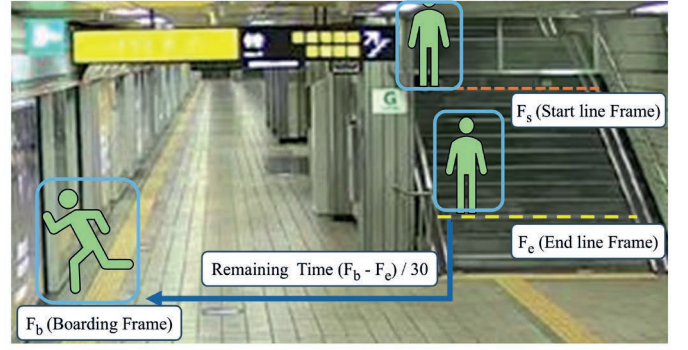


Fig. 2. Definition of remaining time for passenger door arrival prediction.

included in the matching cost. Finally, tracks were terminated and stored as completed trajectories when their bounding boxes passed L_e , which served as the exit boundary.

E. Arrival Time Prediction at the Train Door

The goal of arrival time prediction is to estimate the remaining time until a tracked passenger reaches the subway door. The start line frame (F_s) was defined as the frame in which the passenger's foot crossed the predefined start line and the downward movement could be clearly observed. The end line frame (F_e) was defined as the frame in which the passenger's foot crossed the third step from the bottom of the staircase. The boarding frame (F_b) was defined as the frame in which the passenger's entire body had passed through the subway doorway. It was determined by cross-checking annotations from three experts, as shown in Fig. 2. The ground-truth remaining time was calculated from the frame difference between F_e and F_b at 30 FPS. Passenger trajectories were stored as frame-level position sequences, and abrupt bounding-box changes were corrected using linear interpolation.

Three trajectory features were used for arrival time prediction: average speed, final x -position, and vertical travel distance. Average speed was calculated as the mean Euclidean displacement between consecutive trajectory points, final x -position represented the passenger's horizontal position at F_e , and Vertical travel distance was calculated as the absolute difference between the passenger's center y -coordinate at F_s and that at F_e . Among the 44 passenger samples, six passengers from the designated video segment described in the Dataset Preparation subsection were reserved as an independent test set, while the remaining 38 samples were used for model training with 5-fold cross-validation. Support Vector Regres-

TABLE I
PASSENGER DETECTION PERFORMANCE

Class	Precision	Recall	AP@0.5	AP@0.5:0.95
Ascending	0.925	0.875	0.927	0.686
Descending	0.921	0.924	0.943	0.731
Passing	0.946	0.901	0.925	0.760
Overall (mAP)	0.931	0.900	0.932	0.726

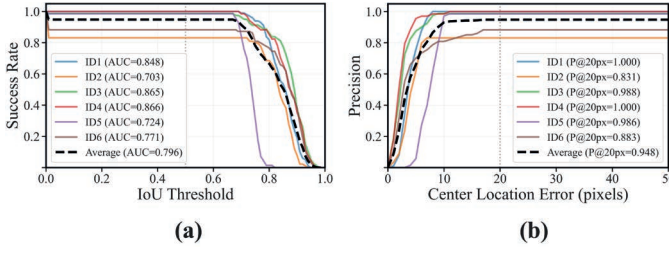


Fig. 3. Tracking evaluation results for six selected passenger tracks: (a) success plot and (b) precision plot.

sion, Random Forest, Gradient Boosting, Bayesian Ridge, and ElasticNet were evaluated to compare nonlinear, ensemble-based, and regularized linear regression approaches.

III. EXPERIMENTS AND RESULTS

A. Experimental Setup

All experiments in this study were conducted on a local PC equipped with an NVIDIA GeForce RTX 5090 GPU and an AMD Ryzen Threadripper PRO 7995WX 96-Core CPU. All stages were implemented in Python. Passenger detection was based on the Ultralytics YOLO framework [13], and tracking and arrival time prediction were implemented using custom Python code.

The arrival time prediction model was built using scikit-learn and evaluated using 5-fold cross-validation.

B. Passenger Detection Performance

The passenger detection performance of the YOLO26n model was evaluated using 1,595 test images and 3,615 passenger instances. The overall results were 0.931 precision, 0.900 recall, 0.932 mAP@0.5, and 0.726 mAP@0.5:0.95. In the inference stage, agnostic Non-Maximum Suppression (NMS) was applied to reduce duplicate detections of the same passenger with different classes.

As shown in Table I, the descending class achieved a recall of 0.924 and the highest AP@0.5 of 0.943 among the three classes. Its precision and mAP@0.5:0.95 were 0.921 and 0.731, respectively. Since the descending class is the main input for tracking and arrival time prediction in this study, these results indicate that the detection model can provide relatively stable inputs for the subsequent tracking stage.

C. Passenger Tracking Performance

Passenger tracking performance was evaluated using the six target passengers described in the Dataset Preparation subsection. The evaluation metrics included the AUC of the success plot, Precision, average IoU, and average Center Location Error (CLE). The success and precision plots followed the evaluation protocol commonly used in object tracking benchmarks [14]. The success plot represents the proportion of frames whose IoU exceeds a given threshold, and its AUC summarizes passenger tracking performance over all IoU thresholds. Precision was calculated as the proportion of frames in which the distance between the predicted and ground

TABLE II
PASSENGER TRACKING PERFORMANCE COMPARISON WITH BASELINE METHODS

Method	Success Plot AUC	Precision @20 px	Avg. IoU	Avg. CLE (px)
SORT	0.793 ± 0.068	0.944 ± 0.070	0.793 ± 0.068	3.896 ± 1.626
ByteTrack	0.783 ± 0.069	0.933 ± 0.076	0.783 ± 0.069	3.912 ± 1.626
CG-SORT	0.796 ± 0.067	0.948 ± 0.066	0.796 ± 0.067	3.897 ± 1.629

truth (GT) bounding-box centers was less than or equal to 20 pixels.

Fig. 3 presents the success and precision plots for the six passenger tracks obtained using CG-SORT. In the success plot, ID1, ID3, and ID4 achieved relatively high AUC values, whereas ID2 and ID6 showed lower performance. In the precision plot, ID1 and ID4 achieved a Precision@20 px of 1.000, while ID2 showed the lowest Precision@20 px. Overall, the main tracking errors were associated with initial detection failures, occlusion, and interference between nearby passengers.

Table II compares CG-SORT with SORT [8] and ByteTrack [15]. CG-SORT achieved the highest AUC of the success plot (0.796 ± 0.067), Precision@20 px (0.948 ± 0.066), and average IoU (0.796 ± 0.067). Its average CLE (3.897 ± 1.629 pixels) was nearly identical to that of SORT (3.896 ± 1.626 pixels), indicating slightly better overall tracking accuracy with comparable localization error.

D. Performance of Arrival Time Prediction at the Train Door

The arrival time prediction dataset consisted of 44 passenger samples whose tracking trajectories were matched with manually annotated door arrival frames. Six passengers used for tracking evaluation were reserved as an independent test set, while the remaining 38 were used for 5-fold model training. The five fold-trained models were then evaluated on the same six test passengers.

Table III reports the mean and standard deviation of the test performance obtained from the five fold-trained models. ElasticNet achieved the best overall performance, with a Mean Absolute Error (MAE) of 0.644 ± 0.021 s, a Root Mean Square Error (RMSE) of 0.739 ± 0.027 s, and an R^2 of 0.399 ± 0.044 .

MAE represents the average absolute prediction error in the original unit of the target variable, whereas RMSE is more sensitive to relatively large errors [16]. Therefore, the predicted remaining arrival time differed from the ground truth by approximately 0.64 s on average, while the higher RMSE indicates that larger errors occurred for some passengers. The

TABLE III
PERFORMANCE OF ARRIVAL TIME PREDICTION AT THE TRAIN DOOR

Model	MAE (s)	RMSE (s)	R^2
ElasticNet	0.644 ± 0.021	0.739 ± 0.027	0.399 ± 0.044
Bayesian Ridge	0.686 ± 0.021	0.792 ± 0.029	0.309 ± 0.050
Random Forest	0.745 ± 0.033	0.895 ± 0.038	0.117 ± 0.074
SVR	0.817 ± 0.051	0.938 ± 0.083	0.025 ± 0.179
Gradient Boosting	0.815 ± 0.095	0.994 ± 0.180	-0.117 ± 0.400

R^2 value indicates that ElasticNet explained approximately 39.9% of the variation in the remaining arrival time [17].

The results indicate that ElasticNet was the most suitable model under the current data and feature conditions. However, further validation using more passengers and diverse crowd conditions is required.

IV. DISCUSSION AND CONCLUSION

This study proposed a subway CCTV-based pipeline that detects passengers at risk of being caught in train doors, tracks their movements across consecutive frames, and predicts their arrival time at the train door. The results demonstrate the feasibility of integrating passenger detection, tracking, and arrival time prediction to support subway door safety.

The passenger detector achieved an overall mAP@0.5 of 0.932 and a descending-class AP@0.5 of 0.943. The proposed CG-SORT achieved an AUC of 0.796 ± 0.067 for the success plot, a Precision@20 px of 0.948 ± 0.066 , an average IoU of 0.796 ± 0.067 , and an average CLE of 3.897 ± 1.629 pixels. For arrival time prediction, ElasticNet achieved the best overall performance, with an MAE of 0.644 ± 0.021 s, an RMSE of 0.739 ± 0.027 s, and an R^2 of 0.399 ± 0.044 . These results indicate that CCTV-based tracking trajectories can provide useful information for estimating the remaining time until passengers reach the subway door.

In a practical door-closing decision system, the predicted remaining arrival time could be compared with the mechanical door-closing duration. If a passenger is predicted to reach the door within a threshold that includes an additional safety margin, the system could recommend delaying door closure. Otherwise, the normal door-closing sequence could proceed. A conservative threshold may reduce the risk of missing an approaching passenger, but it may also increase false alarms and train dwell time. Therefore, the proposed method could initially be used as a decision-support tool for train operators and later extended to broader operational use after sufficient validation.

The current study evaluated tracking using six target passengers and arrival time prediction using 44 passenger samples under a fixed CCTV viewpoint and predefined tracking regions. These conditions limit the generalizability of the findings, and further validation using larger datasets collected under more diverse environmental and crowd conditions is required.

Future work will expand the tracking and arrival-time datasets under diverse crowd conditions, improve tracking stability under occlusion, and validate the prediction results using actual door-closing times. Safety margins and decision thresholds should also be determined empirically, and false-alarm and missed-risk rates should be evaluated to further improve the reliability and practical applicability of the proposed system.

ACKNOWLEDGMENT

This work was supported by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and the

Ministry of Climate, Energy and Environment (MCEE) of the Republic of Korea (No. RS-2022-KP002701).

REFERENCES

- [1] C. Kim, "Seoul Subway Station with the Highest Number of Injured Passengers Is Sadang Station on Line 2," *Yonhap News Agency*, Sep. 2023, [Online]. Available: <https://www.yna.co.kr/view/AKR20230919139900001>. Accessed: Jul. 18, 2026.
- [2] Rail Accident Investigation Branch, "Passenger Trapped in Doors and Dragged at Wood Street Station, North-East London," Rail Accident Investigation Branch, Derby, U.K., Tech. Rep., 2022.
- [3] S. Kim, S. Ryou, Y. Lim, H. Moon, and Y. Jeong, "A Study of Optimizing Obstacle Detection System on Door for Improving Passenger's Safety and Operation Availability by Driverless EMU in Shinbundang Linez," in *Proceedings of Korean Society for Railway Conference*, Gangwon-do, Korea, 2013.
- [4] S. Seriani, T. Fujiyama, and G. De Ana Rodriguez, "Boarding and Alighting Matrix on Behaviour and Interaction at the Platform Train Interface," in *Proceedings of Rail Research UK Association (RRUKA) Conference*, London, U.K., 2016.
- [5] R. Kumar, D. S. Rajpurohit, M. Hanief, S. Sharma, T. Choudhury, and A. Sar, "Detection Of Criminal Activities/Criminal Through CCTV By Live Footage Analysis," in *Proceedings of OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 4.0*, Jun. 2024, p. 292.
- [6] A. A. Serna, X. Yu, and J. Saniie, "AI-Based Security Surveillance and Hazard Detection for Train Platform Safety," in *Proceedings of IEEE International Conference on Electro Information Technology (eIT)*, Eau Claire, WI, USA, 2024, pp. 185–190.
- [7] H. Jo, L. Lim, and J. W. Chong, "Real-Time Passenger Movement Estimation for Subway Door Safety Using Deep Learning," in *Proceedings of IEEE Transportation Electrification Conference and Expo, Asia-Pacific (ITEC Asia-Pacific)*, Singapore, 2025, pp. 1–4.
- [8] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upercroft, "Simple online and realtime tracking," in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, Phoenix, AZ, USA, 2016, pp. 3464–3468. [Online]. Available: <http://dx.doi.org/10.1109/ICIP.2016.7533003>
- [9] R. Wang, Z. Hu, X. Song, and W. Li, "Trajectory Distribution Aware Graph Convolutional Network for Trajectory Prediction Considering Spatio-Temporal Interactions and Scene Information," *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 8, pp. 4304–4316, 2024.
- [10] H. Jo and J. W. Chong, "Trajectory Modeling of Passengers on a Subway Platform with Stairs," in *Proceedings of UKC*, Garden Grove, CA, USA, Dec. 2021.
- [11] H. Jo, Y. Li, L. Lim, and J. W. Chong, "Passenger Trajectory Model and Passenger Movement Estimation System for Preventing Passenger Subway Door Accidents," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–16, 2026.
- [12] D. Li, W. Daamen, and R. M. Goverde, "Estimation of train dwell time at short stops based on track occupation event data: A study at a Dutch railway station," *Journal of Advanced Transportation*, vol. 50, no. 5, pp. 877–896, 2016.
- [13] G. Jocher, J. Qiu, M. Liu, S. Lyu, F. C. Akyon, and M. E. Kalfaoğlu, "Ultralytics YOLO26: Unified Real-Time End-to-End Vision Models," *arXiv:2606.03748*, 2026.
- [14] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proceedings of IEEE conference on computer vision and pattern recognition*, Portland, OR, USA, 2013, pp. 2411–2418.
- [15] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "Bytetrack: Multi-object tracking by associating every detection box," in *Proceedings of European conference on computer vision*, Tel Aviv, Israel, 2022, pp. 1–21.
- [16] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)—Arguments against avoiding RMSE in the literature," *Geoscientific model development*, vol. 7, no. 3, pp. 1247–1250, 2014.
- [17] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *Peerj computer science*, vol. 7, p. e623, 2021.