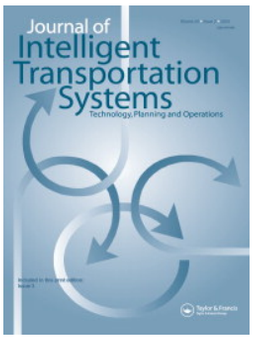

Taylor & Francis
Taylor & Francis Group

Journal of Intelligent Transportation Systems
Technology, Planning, and Operations

ISSN: 1547-2450 (Print) 1547-2442 (Online) Journal homepage: www.tandfonline.com/journals/gits20

Machine learning-based optimization framework for vehicle reidentification between detectors at signalized intersections

Pramesh Pudasaini, Henrick Haule & Yao-Jan Wu

To cite this article: Pramesh Pudasaini, Henrick Haule & Yao-Jan Wu (04 Jul 2025): Machine learning-based optimization framework for vehicle reidentification between detectors at signalized intersections, Journal of Intelligent Transportation Systems, DOI: 10.1080/15472450.2025.2526395

To link to this article: https://doi.org/10.1080/15472450.2025.2526395

JOURNAL OF INTELLIGENT TRANSPORTATION SYSTEMS
https://doi.org/10.1080/15472450.2025.2526395

Machine learning-based optimization framework for vehicle reidentification between detectors at signalized intersections

Pramesh Pudasaini[a], Henrick Haule[b], and Yao-Jan Wu[a]

[a]Department of Civil and Architectural Engineering and Mechanics, The University of Arizona, Tucson, Arizona; [b]Department of Civil and Environmental Engineering, The University of Alabama in Huntsville, Huntsville, AL, USA

ABSTRACT

Signalized intersections are equipped with advance and stop-bar detectors that detect vehicles at discrete locations without linking or reidentifying them over the approach area. Accurate tracking and reidentification of vehicles between these detectors could provide valuable driver behavior data, especially during the safety-critical yellow onset periods. However, reidentifying vehicles using non-visual detection data is challenging and not well-explored, with existing analytical models relying on a priori-calibrated parameters. To this end, we propose a machine learning (ML)-based reidentification framework for accurately tracking vehicles over the advance and stop bar loop detectors. The framework comprises two major components: advanced ML and deep learning (DL) models for accurately predicting the travel time between detectors and a novel optimization model that utilizes these predicted travel times and actuation events for reidentifying vehicles. Tests carried out on a major intersection approach in Phoenix, Arizona, showed that the optimization framework based on neural oblivious decision ensemble (NODE) reidentified vehicles even at congested conditions with 94.5% precision and 92.1% recall, outperforming state-of-the-art analytical, conventional ML, and comparable DL models. The low false alarm rate and high recall of this reidentification framework open avenues for obtaining valuable driver behavior data at the yellow onset to analyze stop/go behavior, dilemma zone entry/exit, red light running, and crossing conflicts at signalized intersections.

ARTICLE HISTORY
Received 9 July 2024
Revised 20 June 2025
Accepted 24 June 2025

KEYWORDS
high-resolution event data; loop detector; machine learning; signalized intersection; vehicle reidentification

Introduction

Vehicle reidentification is a core component of intelligent transportation systems (ITS), facilitating automatic vehicle tracking and identification to enhance traffic management and safety (Wang et al., 2019; Yi et al., 2025). Conventional vehicle reidentification primarily focuses on traffic surveillance and relies on images captured from multiple cameras with non-overlapping views (Khan & Ullah, 2019; Zapletal & Herout, 2016; Zhang et al., 2023). Practical applications in this domain of vehicle reidentification include live traffic monitoring, suspicious vehicle tracking, travel time prediction, intelligent parking, vehicle counting, and automatic toll collection (Gazzah et al., 2017; Hu et al., 2024; Khan & Ullah, 2019; Qian et al., 2024; Wang et al., 2019; Xiong et al., 2021; Zhang et al., 2022). In contrast to such reidentification based on visual data for traffic surveillance purposes, this study addresses the unique challenge of reidentifying vehicles using non-visual detection data, with a focus on obtaining valuable driver behavior data at signalized intersection approaches.

Signalized intersections in the United States are predominantly monitored using wired inductive loops or video-based sensors installed at two key locations: the advance detector, situated a certain distance upstream from the intersection stop line, and the stop bar detector, positioned right at the intersection stop line (Chandler et al., 2013; Urbanik et al., 2015). In this typical installation layout, vehicles approaching an intersection are detected at discrete locations, but the absence of a linking mechanism between detectors prevents accurate tracking of individual vehicles across the approach area. This limitation hinders efforts to capture valuable driver behavior data, particularly in safety-critical scenarios such as decision-making during the yellow onset (Pudasaini, Haule, et al., 2025). Collecting such behavioral insights is currently challenging due the limited availability of crowdsourced

CONTACT Pramesh Pudasaini pramesh@arizona.edu Department of Civil and Architectural Engineering and Mechanics, The University of Arizona, 1209 E 2nd St, Tucson, AZ 85721, USA

trajectory data (Kandiboina et al., 2024; Pudasaini, Pathivada, et al., 2025) and the reliance on labor-intensive, manual extraction of driver behavior data from field-recorded video footage (Abdelhalim et al., 2021; Do et al., 2023; Li & Wei, 2013; Rahman et al., 2021). Moreover, reidentification is particularly challenging with non-visual data from widely deployed inductive loop detectors, which lack the richness of visual tracking methods to capture detailed vehicle features. Unlike traffic surveillance cameras, conventional loop detectors provide only discrete actuation events, making it difficult to link vehicles across detectors without advanced predictive models capable of handling the inherent variability in driver behavior and traffic conditions. Therefore, a data-driven approach to reidentify vehicles between the advance and stop bar detectors could provide an automated and scalable alternative to obtain driver responses and driving behavior data at an intersection approach.

Several studies have highlighted the safety implications of reidentifying vehicles during the yellow onset when drivers make stop/run decisions (Chen et al., 2017; Ding et al., 2016; Liu et al., 2017; Lu et al., 2015; Ren et al., 2016; Wu et al., 2013). Despite the realized safety implications, the uniformity in detector installation layout, and the abundance of high-resolution event data, methodological advancements in accurately tracking and reidentifying vehicles across detectors at an intersection approach remain limited. Recent advances in traffic surveillance-focused vehicle reidentification have observed a huge shift from traditional machine learning (ML) to deep learning (DL)-based architectures (Amiri et al., 2024; Ning et al., 2025; Qian et al., 2024; Wang et al., 2019; Yi et al., 2025). In contrast, existing studies on reidentifying vehicles using non-visual detection data are not only limited but also constrained to analytical reidentification techniques. Multiple studies have proposed reidentification methodologies using inductive signatures from loop detectors but primarily for estimating travel times along corridors (Jeng et al., 2010; Lin & Tong, 2011; Wang et al., 2014). However, to the best of the authors' knowledge, only a handful of studies have addressed vehicle reidentification using non-visual loop detector data. Moreover, these methods were formulated as an analytical problem to match actuation events between detectors (Chen et al., 2017; Ding et al., 2016; Liu et al., 2017; Lu et al., 2015; Pudasaini et al., 2024; Ren et al., 2016; Wu et al., 2013). Given the importance of understanding drivers' behavior at the yellow onset, these studies were primarily focused on analyzing yellow/red light running, stop/go behavior, dilemma zone, or crossing conflicts.

Wu et al. (2013) reidentified detection events at the stop bar and advance detectors based on travel time estimates between the detectors, where speed was computed relying on an a priori calibrated effective vehicle length. A pair of events was considered a "match" if the observed travel time for the pair fell within the range of estimated travel time ± 2 s, accounting for the variances in effective vehicle lengths and deceleration rates. Lu et al. (2015) proposed a similar matching algorithm to reidentify vehicles between the stop bar and downstream entrance detectors. For each candidate reidentification pair, the authors introduced a match strength function: the reciprocal of the difference in observed and expected travel time. The reidentification pair with the highest match strength was considered in case of multiple matches. Ren et al. (2016) and Ding et al. (2016) implemented a window-searching method to reidentify vehicles between advance and stop-bar detectors. For each advance detection, this method deployed a "time window"–based on minimum and maximum travel time between detectors–within which a detection at the stop bar location can be identified. The minimum travel time is estimated assuming a maximum acceleration rate, whereas the maximum is computed assuming the vehicle fully stops at the stop bar. Reidentification between detectors was conducted based on the highest match strength in Ren et al. (2016) and the lowest percentage error between estimated and observed travel time in Ding et al. (2016). Similarly, Liu et al. (2017) proposed a matching algorithm between the stop bar and downstream entrance detectors for estimating crossing conflicts at signalized intersections, adopting an approach similar to Wu et al. (2013). Chen et al. (2017) implemented an analytical model similar to Ding et al. (2016) to reidentify vehicles between advance and stop bar detectors for estimating red-light running frequency. Recently, Pudasaini et al. (2024) proposed a rule-based algorithm for analytically matching actuation events between advance and stop-bar detectors using predefined, intersection-specific ideal travel times for stopping and running.

Despite minor differences in searching and classifying vehicle reidentification pairs, a common theme in all aforesaid studies was confining reidentification to an analytical matching task using kinematic models of motion. As a result, these analytical reidentification models suffer from two major drawbacks. First, their travel time estimation is based on either predefined

parameters or computing speed over each detector using an a priori known effective vehicle length. Such reliance on constant, predefined parameters and the requirement of the parameters' pre-calibration for a specific site limit the transferability of these analytical methods to new intersections, detector configurations, and traffic dynamics. Second, besides Pudasaini et al. (2024), the accuracy of analytically matching events between detectors was not evaluated or reported in any study. In summary, despite the enormous potential to yield valuable driver behavior information, developing a robust vehicle reidentification framework based on non-visual detection data remains a significant research gap in the existing literature.

This study formulates a novel ML-based optimization framework for reidentifying vehicles between the advance and stop bar detectors using non-visual detection data. The framework focuses on generating accurate datasets of driver behavior and decision-making through vehicles reidentified during the safety-critical yellow onset periods to aid studies on traffic operations and safety. The proposed reidentification framework comprises two major components: prediction of travel time between detectors and vehicle reidentification based on actuation events over detectors. We test six state-of-the-art ML and DL models for accurately predicting the travel time between detectors and formulate an optimization framework that utilizes these predicted travel times for reidentifying vehicles. Furthermore, we compare and discuss reidentification results from the proposed framework with three existing analytical methods. The major contributions of this study are threefold. First, it proposes a highly accurate and robust data-driven framework for vehicle reidentification using non-visual detection data, addressing a critical gap in the existing literature. Second, it provides a scalable and automated alternative to labor-intensive manual methods for collecting valuable driver behavior data, particularly during safety-critical yellow onset periods. Third, it advances the state-of-the-art in ITS by demonstrating the potential of high-resolution event data in tracking vehicles for improving intersection safety and traffic management.

Data collection and processing

Study intersection

The westbound approach of Indian School Rd and 19th Ave in Phoenix, Arizona, was selected as the site for testing the proposed framework for vehicle reidentification. Indian School Rd is one of the busiest arterials in Phoenix, providing a direct connection between the I-17 highway and the USA 51 highway. In the vicinity of 19th Ave, Indian School Rd had an annual average daily traffic of approximately 45,000 vehicles in 2023. Timestamped traffic data from the intersection's signal controllers–including actuations on these detectors–is archived in real-time by the City of Phoenix Street Transportation Department (Pudasaini et al., 2024). Besides being equipped with lane-by-lane loop detectors at the advance and stop bar locations, video recordings from a traffic camera were available for specific periods to validate the signal changes and detector actuations, making this intersection an ideal test site for vehicle reidentification.

Figure 1 presents the intersection approach with the layout and configuration for lanes and detectors. The approach has lane-by-lane loop detectors on three through lanes at the advance and stop bar locations. Advance detectors are 5 ft long, whereas stop bar detectors are 40 ft long. The distance between the stop line and the advance detector is 300 ft. The speed limit of the approach is 35 mph.

High-resolution event data collection and processing

High-resolution events in the form of timestamped logs from the signal controller and detectors were collected from TransSuite, a centralized traffic management system archiving real-time data from the City of Phoenix intersections (Pudasaini et al., 2024). Timestamped events comprise signal phase changes, detector actuations, and communication attempts at a resolution of 0.1 s. We divided the available high-resolution event datasets into two groups. The first dataset, collected for 15.5 h from three weekdays (6/12/2022, 6/14/2022, 3/27/2023), also had concurrent ground-truth video recordings archived in the TransSuite system. The availability of both event datasets and concurrent videos meant this dataset could be used to generate "ground-truth reidentification (ReId) pairs" and evaluate the accuracy of reidentified vehicles. The second dataset, collected for 14 days (1st and 3rd weeks) in January 2023, did not have concurrent video recordings available. Owing to the large sample of high-resolution events, this dataset was used to generate "inferred ReId pairs" for training ML/DL models. We discuss the details of data processing in the following section. The Methodology section delves into generating inferred and ground-truth ReId pairs for model training and testing.

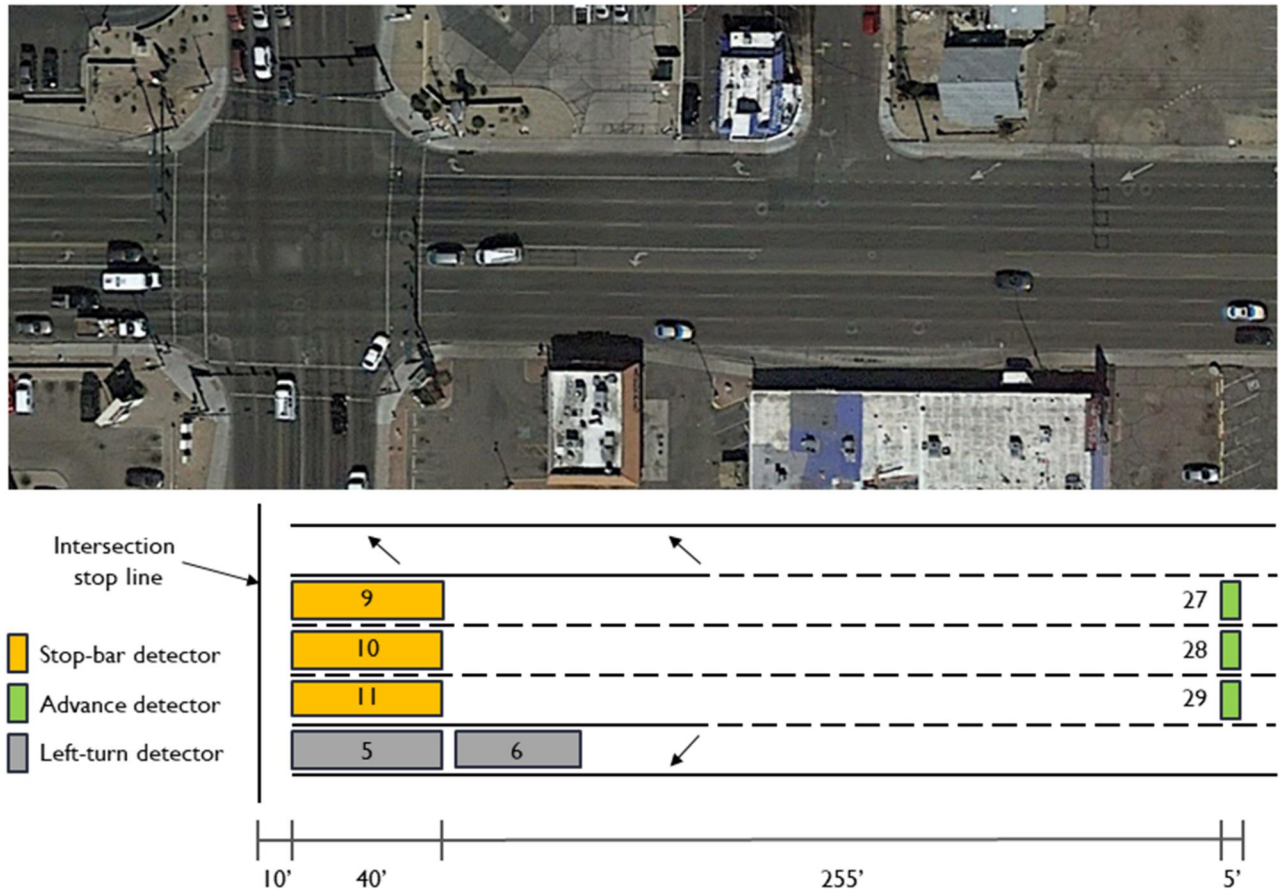

Figure 1. Study intersection with detector layout and configuration.

Figure 2 presents the overall data processing in three steps: pre-processing raw high-resolution data, processing signal phase change and detector actuation events, and filtering actuation events at the yellow onset. The raw data were first segregated into individual hours for data pre-processing. An assessment of communication loss and data continuity plots for signal phase changes and detector actuations provided data quality checks for further processing. Then, the two input datasets–signal phase change and detector actuation events–were loaded, and different parameters of interest were computed.

Table 1 lists the indices, sets, and parameters used throughout this study. We assume each cycle c starts on the yellow indication for processing events. The timestamps of yellow (T_c^y), red (T_c^r), and green (T_c^g) indications for each cycle are obtained from the signal phase change dataset. After assigning relevant indices, both input datasets were merged and split by lane to compute actuation-related parameters. Let the index $k : k \in K$ denote any actuation in general for data processing. Indices $i : i \in I, I \subseteq K$ and $j : j \in J, J \subseteq K$ represent respective actuations at the advance and stop bar detectors. Let T_k^o and T_k^f be the timestamps of actuations "on" and "off" for each detector. Detector occupancy time (τ_k^D), headway (τ_k^H), and gap (τ_k^G) as shown in Equations (1–3), respectively, were computed for each actuation and detector.

Next, the signal change during actuation ($SCA_k : SCA_k \in \{YY, YR, RR, RG, GG, GY\}$) is an important parameter denoting a combination of changes in signal indication when a particular actuation is turned "on" and "off" (Pudasaini et al., 2023). For instance, $SCA_i \in \{GG\}$ indicates that the advance detector actuation started on green and ended on green; $SCA_j \in \{RG\}$ indicates that the stop bar detector actuation started on red and ended on green. Together, SCA_k and τ_k^D yield information regarding whether a vehicle crossed or stopped over a detector. For example, $SCA_k \in \{GG\}$ usually have a low τ_k^D at both advance and stop bar detectors, implying that such actuations crossed the detector. The most useful case is $SCA_j \in \{RG\}$, which usually has a high τ_j^D, meaning such stop bar actuations correspond to vehicles actuating the detector on red, stopping before the stop line, and leaving the detector on green.

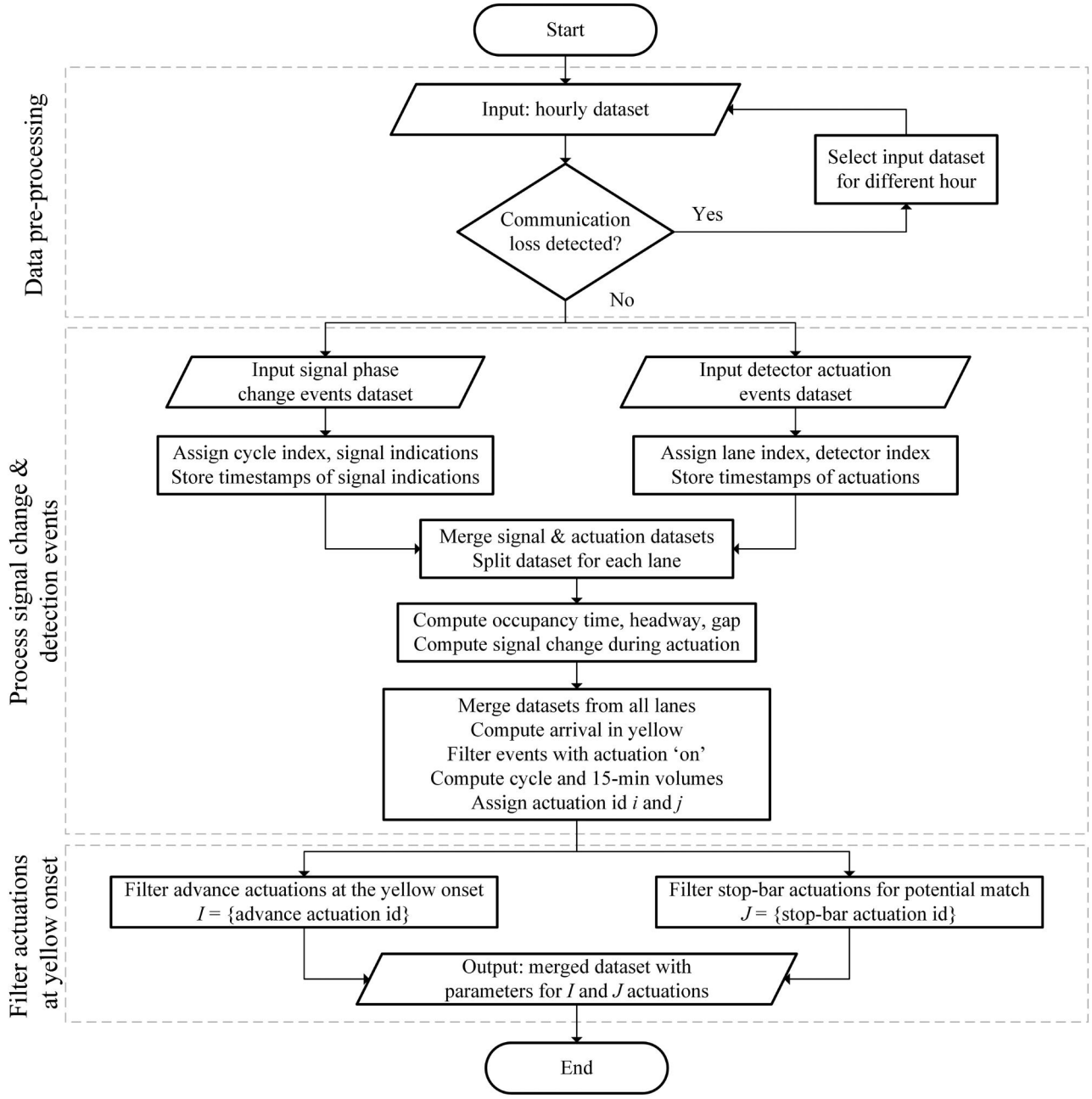

Figure 2. Data processing flowchart.

Interested readers are referred to Pudasaini et al. (2023) for a detailed discussion and empirical analysis of SCA_k with respect to τ_k^D, τ_k^H, and τ_k^G parameters.

Once actuation-related parameters are computed on each lane, the datasets are merged again for all through lanes. Let AIY_k, computed using Equation (4), denote the arrival time in yellow for actuation k. $AIY_k = 3$, for instance, indicates that the vehicle arrived at a detector 3 s after the signal turned yellow; similarly, $AIY_k = -2.5$ indicates the vehicle's arrival at the detector 2.5 s before the signal turned yellow. Arrival volume is also computed for each cycle and 15-min period. In the final step of processing events, the dataset is filtered for all events with actuation "on," and each actuation is assigned an actuation index.

$$\tau_k^D = T_k^f - T_k^o \ \forall k \tag{1}$$

$$\tau_k^H = T_{k+1}^o - T_k^o \ \forall k \tag{2}$$

$$\tau_k^G = T_{k+1}^o - T_k^f \ \forall k \tag{3}$$

$$AIY_k = \begin{cases} T_k^o - T_c^y, SCA_k \in \{YY, YR, RR, RG\} \\ T_k^o - T_{c+1}^y, SCA_k \in \{GG, GY\} \end{cases} \forall k \tag{4}$$

Upon processing all signal phase change and actuation events, we filter only those actuations which are

Table 1. Notations used in this study.

Notation	Description
Indices	
k	Any detector actuation
i, j	Detector actuation at advance and stop bar detectors
c	Cycle
o, f	Detector "on" and "off" events
y, r, g	Yellow, red, and green indication events
Sets	
SCA_k	Signal change during actuation k; $SCA \in \{YY, YR, RR, RG, GG, GY\}$
Parameters	
T_c^y, T_c^r, T_c^g	Timestamp of yellow, red, and green indications for cycle c
T_k^o, T_k^f	Timestamp of actuation "on" and "off" for vehicle k
τ_k^D	Occupancy time for actuation k (sec)
τ_k^H	Time headway between detector actuation k and $k+1$ (sec)
τ_k^G	Time gap between detector actuation k and $k+1$ (sec)
AIY_k	Arrival in yellow of actuation k at a detector (sec)
$t_{\min}, t_{\max}$	Minimum and maximum travel time to constrain the search space for vehicle traveling between rear ends of advance and stop bar detector (sec)
$t'_{\min}, t'_{\max}$	Minimum and maximum travel time to constrain the search space for vehicle traveling between extreme ends of advance and stop bar detector (sec)
ΔS_{ij}	Distance between extreme ends of detectors (ft)
v_{adv}	Velocity at advance detector (ft/sec)
$a_{\max}$	Maximum acceleration (ft/sec^2)
x_{ij}, y_{ij}	Candidate reidentification pair (i, j) obtained from manual inference and video observation
E_{ij}	Error associated with candidate reidentification pair (i, j)
L_{ij}	Lane change for candidate reidentification pair (i, j)
t_{ij}	Actual travel time for reidentification pair (i, j) (sec)
t_i^{pred}	Predicted travel time for advance actuation i (sec)
y_{ij}^{pred}, y_{ij}^{ground}	Predicted and ground truth reidentification pair (i, j)

susceptible to the yellow onset while approaching the intersection. At the advance location, we filter actuations based on two factors: a) yellow indication interval of 3.6 sec and b) approximate ideal travel time from the advance detector to the intersection stop line of 6.0 sec, i.e., traveling 310 ft at 35 mph. Hence, retaining advance actuations with $(AIY_i \leq 3.6, \forall SCA_i \in \{\text{YY, YR, RR, RG}\}) \cup (AIY_i \geq -6, \forall SCA_i \in \{\text{GG, GY}\})$ captures all vehicle arrivals at the yellow onset from the intersection stop line to a point 3.6 s beyond the advance location. Similarly, at the stop bar location, we first filter out actuations with $SCA_j \in \{\text{GG}\}$ as these vehicles crossing the stop bar on green are not influenced by the yellow onset. Then, we retain the stop bar actuations with $(AIY_j \leq 12, \forall SCA_j \in \{\text{YY, YR, RR, RG}\}) \cup (AIY_j \geq -1.5, \forall SCA_j \in \{\text{GY}\})$. $AIY_j \leq 12, \forall SCA_j \in \{\text{YY, YR, RR, RG}\}$ ensures potential matches for vehicles that decelerate downstream of the advance location and decide to stop before the intersection stop line. On the other hand, $AIY_j \geq -1.5, \forall SCA_j \in \{\text{GY}\}$ is for retaining stop bar actuations that face the yellow onset between the intersection stop line and the rear end of the stop bar detector. The final output of the overall data processing was a merged dataset containing parameters and indices for sets of filtered actuations at the advance and stop bar locations.

Ground-truth data collection

The video recordings available for 15.5 h were carefully reviewed to obtain a ground-truth dataset of ReId pairs, which will later be used to validate the performance and robustness of the proposed vehicle reidentification framework. Two research specialists were trained to reidentify vehicles between the advance and stop bar detectors at the yellow onset. From video observation, the focus was first identifying vehicles arriving at the advance detector with $-6 \leq AIY_i \leq 3.6$, i.e., up to 3.6 s into the yellow and 6 s before the indication turned yellow. The corresponding actuation indices at the advance and stop bar detectors were noted for each identified vehicle that made a through movement to the stop bar. Lane changes on the through lanes were also considered while recording these pairs of actuation indices. As the final output, we obtained 580 ground-truth ReId pairs for vehicles reidentified between advance and stop bar detectors. Only 15 ReId pairs out of the 580, corresponding to 2.6% of the total ground-truth pairs, involved lane changes while approaching the intersection at the yellow onset.

Methodology

The major contribution of this study is formulating a novel ML-based optimization framework for vehicle

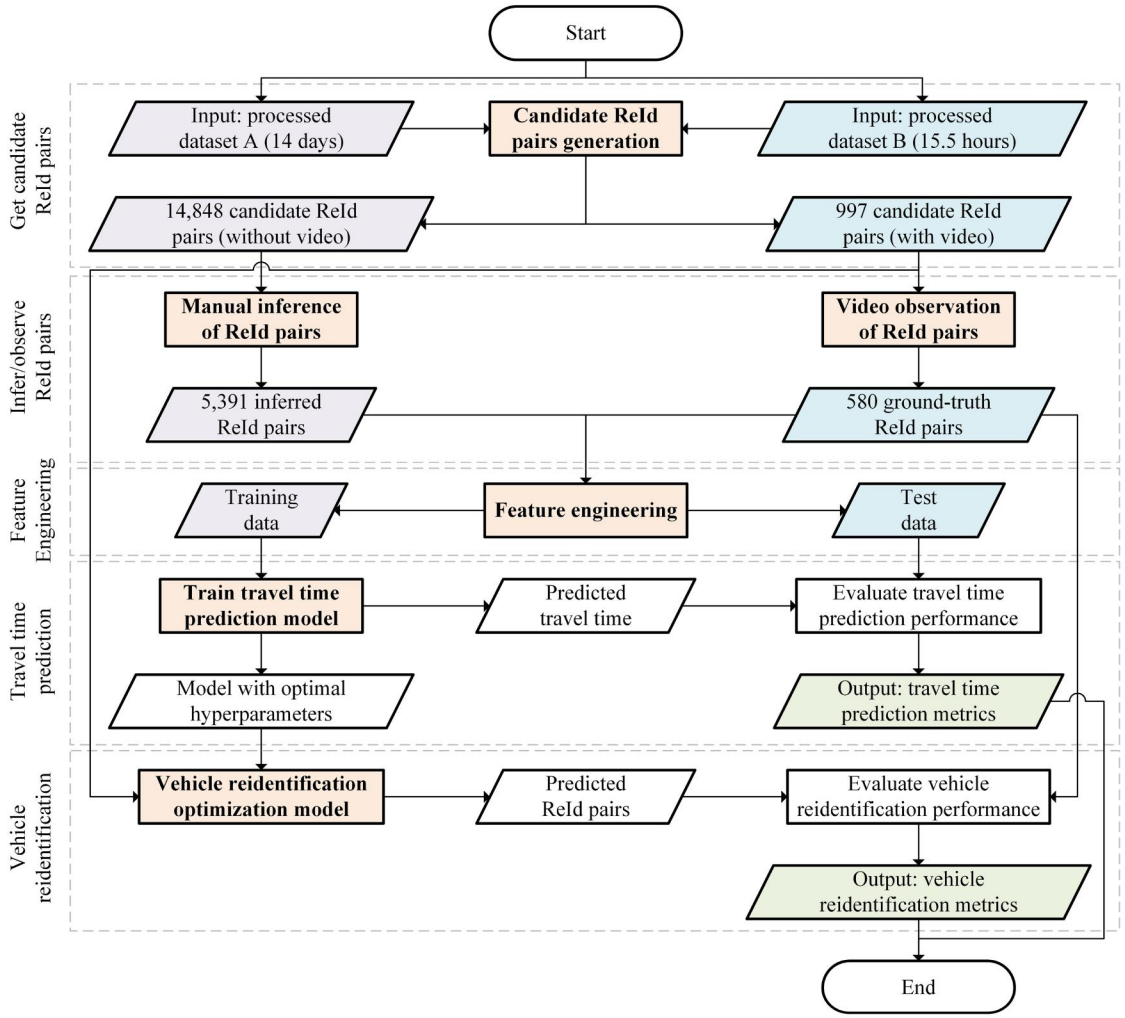

Figure 3. Proposed vehicle reidentification framework.

reidentification between detectors at an intersection approach using non-visual detection data. Figure 3 presents a high-level architecture of the proposed framework, which encompasses five steps: (1) generating sets of candidate ReId pairs, (2) generating inferred and ground-truth ReId pairs, (3) feature engineering, (4) training/testing ML models for travel time prediction, (and 5) reidentifying vehicles using an optimization model. The following sections discuss these steps in detail.

Generating candidate reidentification pairs

The data processing procedure yielded two sets I and J of advance and stop bar detector actuations at the yellow onset. For vehicle reidentification between these two sets, we first seek to generate a set of candidate ReId pairs between I and J by constraining the search space for any advance actuation i. This constraining procedure, similar to Pudasaini et al. (2024), has two subtle differences from the "window-searching" method adopted by Lu et al. (2015) and Chen et al. (2017). The proposed framework estimates a search space and implements it for "all" i based on the minimum and maximum values of travel time ($t_{\min}$ and $t_{\max}$) between the "rear" ends of advance and stop bar detectors. The window searching method, on the other hand, implemented a time window for "each" i based on the minimum and maximum values of travel time ($t'_{\min}$ and $t'_{\max}$) between the "extreme" ends, i.e., front end of stop bar and rear end of advance detectors. The focus here is on rear ends because the proposed vehicle reidentification methodology, unlike prior analytical methods, does not rely on stop bar detector information other than the timestamp at which the stop bar detector was actuated.

In the window-searching method, $t'_{\min}$ and $t'_{\max}$ were estimated using Equations (5) and (6), where ΔS_{ij} is the distance between the extreme ends of detectors, v_{adv} is the velocity at the advance detector, and $a_{\max}$ is the maximum acceleration rate of 6 ft/s^2

as suggested by Long (2000). Assume a vehicle approaching the intersection at the yellow onset at a high velocity (say 50 mph) and crossing the intersection stop line; this results in a $t'_{\min}$ of 3.1 s. Similarly, consider a vehicle approaching the intersection at the yellow onset at a low velocity (say 30 mph) and stopping at the stop bar; this yields a $t'_{\max}$ of 13.6 sec. Since $t_{\min} < t'_{\min}$ and $t_{\max} < t'_{\max}$, we approximate 2.5 and 13 s as the values of $t_{\min}$ and $t_{\max}$ between the rear ends of detectors to constrain the search space for each i. These limits made a good approximation in this study because empirical observation of ground-truth ReId pairs yielded similar limiting travel time values of 2.9 and 11.4 sec between the rear ends of detectors.

$$t'_{\min} = \frac{2\Delta S_{ij}}{v_{\text{adv}} + \sqrt{{v_{\text{adv}}}^2 + 2a_{\max}\Delta S_{ij}}} \tag{5}$$

$$t'_{\max} = \frac{2\Delta S_{ij}}{v_{\text{adv}}} \tag{6}$$

Next, for each advance actuation i, we look for a stop bar actuation j on the same lane within the $t_{\min}$, $t_{\max}$ travel time limits. We discard potential lane changes because we observed only 2.6% of vehicles changing lanes during the yellow onset. As a potential ReId pair, i can have multiple candidates in J. This procedure was implemented using a Python script, which resulted in a set of $x_{ij} : x_{ij} \in X_{ij}$, denoting a candidate ReId pair between i and j. As the final output, 14,848 pairs of x_{ij} were generated from the processed high-resolution dataset of 14 days, while the ground-truth dataset yielded 997 candidate pairs.

Generating inferred reidentification pairs dataset

The proposed vehicle reidentification framework relies on accurate travel time prediction between the advance detector and the rear end of the stop bar detector using information from only the advance location. Such travel time prediction problem necessitates a large training dataset, for which we manually examine each x_{ij} from the 14,848 candidate ReId pairs. Figure 4 explains this manual inference procedure with four examples, where each observation represents a detector actuation color-coded with the SCA parameter. The x-axis in this figure represents the timestamp (minute and second) of detector actuation "on" at the 17th hour on 18 January 2023; the y-axis represents a combination of advance and corresponding stop bar detectors on a particular lane.

Figure 4(a) presents a simple example of a manual reidentification when only a few vehicles approach the intersection at the yellow onset. For illustration, this figure has the actuations labeled by an ID. Recall each advance actuation i has a set of candidate ReId pair x_{ij} on the same lane within the $t_{\min}$, $t_{\max}$ travel time limit. For advance actuation index 50 on the right through lane, for instance, the candidate ReId set would be {50–60, 50–62}, indicating two potential candidates at the stop bar for reidentification. The 50–60 and 50–62 ReId pairs correspond to travel

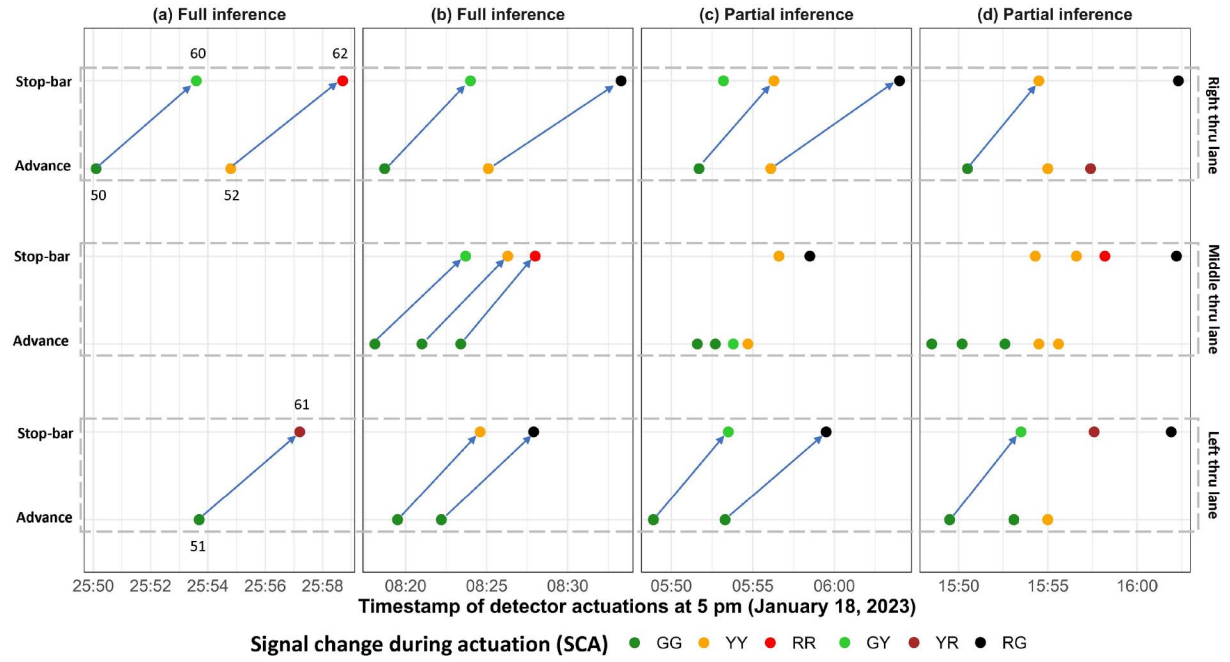

Figure 4. Manual inference of reidentification pairs.

times of 3.5 and 8.6 s, respectively. Based on the SCA of the stop bar actuations, ID 50 crosses the intersection with either 60 or 62 as the ReId pair. As a travel time of 3.5 sec is logical to traverse a distance of 260 ft for ID 50, its ReId pair is identified as ID 60. Next, ID 52 has only one candidate ID at the stop bar, so 52–62 is another ReId pair. 51–61, the only candidate pair on the left through lane, is inferred as a ReId pair. Like the example discussed above, manual inference yields 7 ReId pairs for a more complex case of full inference, as shown in Figure 4(b).

Figure 4(c,d) presents examples of manual reidentification when the number of vehicles approaching the intersection at the yellow onset is high. In such cases, easy manual inference of all candidate ReId pairs might not be possible. In Figure 4(c), for instance, the intuition of SCA and associated ideal travel time yields two ReId pairs each on the right through and left through lanes. However, the inference on the middle lane is complicated as there are four advance actuations with only two stop bar actuations for potential reidentification. Here, any two advance actuations might have undergone a lane change to either the right-turn lane or left-turn lane or might have entered the intermediate driveway. Due to such ambiguity, no inference of a ReId pair is considered on the middle through lane. The case is similar in Figure 4(d), where only two ReId pairs can be intuitively identified, while the reidentification of other candidate pairs is ambiguous.

The two research specialists manually reviewed plots similar to Figure 4 for manual inference of ReId pairs from the generated candidate sets. Only the ReId pairs that can be inferred with the intuition of SCA and ideal travel time were noted as an "inferred ReId pair"; the rest were ignored as the goal of this manual inference was creating a dataset sufficient for training travel time prediction between the rear ends of detectors. This laborious process resulted in reviewing 321 h of processed high-resolution event dataset plots, yielding 5,391 inferred ReId pairs with corresponding travel times. Note that the advance actuation parameters computed during the data processing step were carried over for each ReId pair.

Feature engineering

Based on the data processing carried out earlier, several features for travel time prediction are associated with a ReId pair. The 15-min and cycle volumes are representative of traffic volume or level of congestion at the intersection approach. Occupancy time, following headway, leading headway, and following gap are parameters computed from the detector actuation events dataset; these parameters consider the influence of following and leading vehicles while predicting travel time. The AIY, also computed during data processing, impacts travel time as a vehicle's arrival before the yellow at the advance location indicates a higher likelihood of crossing the intersection; in contrast, those arriving after the yellow tend to stop before the intersection stop line. Lane was categorized as a categorical variable with one-hot encoding to consider the effect of lane position on travel time. Similarly, the SCA parameter was also one-hot encoded as it directly influences travel time to the stop bar. The final feature for travel time prediction was the car-following effect. Brackstone and McDonald (1999) identified a commonly used threshold of 1.5 s for time headway to define car-following. Based on this limit, we categorized car-following at the advance detector as a binary variable for each ReId pair.

Table 2 summarizes the 13 features generated for 5,391 inferred ReId pairs and 997 ground-truth ReId pairs for training and testing different ML-based models for travel time prediction. The target variable is the travel time between the rear ends of detectors for each ReId pair.

Table 2. Summary of features for travel time prediction.

Category	Feature names	Feature description	Feature type
Detector actuation	volume_15	Arrival volume at advance location during 15-min interval	Count
	volume_cycle	Arrival volume at advance location during a cycle	Count
	car_follow	Car-following behavior at advance detector (1 = yes, 0 = no)	Binary
	occ_time	Occupancy time over advance detector	Continuous
	headway_foll	Headway between target and leading vehicle at advance detector	Continuous
	headway_lead	Headway between target and following vehicle at advance detector	Continuous
	gap_foll	Gap between target and leading vehicle at advance detector	Continuous
Signal phase change and detector actuation	AIY	Arrival time in yellow at advance detector	Continuous
	is_SCA_GY	Signal change during actuation = GY? (1 = yes, 0 = no)	Binary
	is_SCA_YY	Signal change during actuation = YY? (1 = yes, 0 = no)	Binary
	is_SCA_YR	Signal change during actuation = YR? (1 = yes, 0 = no)	Binary
Lane position	is_lane_R	Lane position = right? (1 = yes, 0 = no)	Binary
	is_lane_M	Lane position = middle? (1 = yes, 0 = no)	Binary

ML and DL models for travel time prediction

We employed six ML/DL models for travel time prediction such that the predicted travel time can be used to reidentify vehicles between the advance and stop bar detectors. Model selection was based on the tabular data structure and the absence of spatial and temporal dependencies among generated features. These models spanning conventional ML approaches, tree-based ensembles, and DL architectures are, for brevity, briefly introduced below.

Support Vector Regression (SVR)

SVR is a conventional supervised learning algorithm that extends the principles of support vector machines (SVM) to regression tasks. It maps input features into a high-dimensional space using a kernel function and finds a hyperplane that minimizes prediction error within an ϵ-insensitive loss function. Given a dataset $\{(x_i, y_i)\}_{i=1}^{n}$, the optimization problem is formulated as:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\left(\xi_i + \xi_i^*\right) \tag{7}$$

subject to:

$$y_i - \langle w, \emptyset(x_i)\rangle - b \leq \epsilon + \xi_i \tag{8}$$

$$\langle w, \emptyset(x_i)\rangle + b - y_i \leq \epsilon + \xi_i^* \tag{9}$$

$$\xi_i, \xi_i^* \geq 0 \tag{10}$$

where C is a regularization parameter, $\emptyset(x)$ is a kernel function, and ξ_i, ξ_i^* are slack variables.

Random Forest (RF)

RF is an ensemble learning method that constructs multiple decision trees during training and outputs the mean prediction of the individual trees. Each tree is trained on a bootstrap sample of the training data, and at each node, splits are made using a random subset of features. The prediction for an input x is given by:

$$\hat{y} = \frac{1}{T}\sum_{t=1}^{T} f_t(x) \tag{11}$$

where T is the total number of trees, and $f_t(x)$ is the prediction of the t-th tree. RF mitigates overfitting by averaging predictions and introducing feature randomness.

XGBoost

XGBoost is a scalable and efficient implementation of gradient-boosted decision trees (Chen & Guestrin, 2016). It optimizes a regularized objective function that includes both the loss function and a penalty term for model complexity. The model is built in a sequential manner, where each new tree f_t minimizes the following objective:

$$L = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i\right) + \sum_{t=1}^{T} \Omega(f_t) \tag{12}$$

where $l\left(y_i, \hat{y}_i\right)$ is the loss function, and $\Omega(f_t)$ is a regularization term penalizing model complexity. XGBoost uses shrinking, efficient split finding, and column sampling to improve predictive performance.

Fully Connected Neural Network (FCNN)

FCNN is a type of artificial neural network (ANN) where each neuron in a layer is connected to every neuron in the subsequent layer. It consists of multiple layers: an input layer, one or more hidden layers, and an output layer. The network learns a mapping $f: \mathbb{R}^d \rightarrow \mathbb{R}$ through a series of nonlinear transformations. Each neuron applies an activation function σ to a weighted sum of inputs, yielding the following computation at each layer:

$$h^{(l)} = \sigma\left(W^{(l)}h^{(l-1)} + b^{(l)}\right) \tag{13}$$

where $h^{(l)}$ represents activations at layer l, $W^{(l)}$ and $b^{(l)}$ are the learnable weights and biases of the l-th layer, and σ is a nonlinear activation function. The network is trained using backpropagation and gradient descent, minimizing a loss function such as mean squared error for regression.

TabNet

TabNet is a state-of-the-art DL architecture specifically designed for tabular data (Arik & Pfister, 2021). Unlike traditional neural networks that transform all input features simultaneously, TabNet incorporates sequential attention mechanisms to dynamically select and process the most relevant features at each decision step, making the model both interpretable and effective. The TabNet architecture consists of multiple steps, where each step i performs feature selection and processing as follows.

$$\hat{y}_i = \sum_{j=1}^{N} f_j(x) \cdot a_{ij} \tag{14}$$

where $f_j(x)$ represents feature transformations, and a_{ij} are attention weights that determine the importance of feature j at step i. The final prediction is the sum of outputs from all steps. TabNet applies sparse feature selection and also includes a feature reuse

mechanism, allowing previously selected features to be revisited in later steps.

Neural Oblivious Decision Ensemble (NODE)

NODE is a differentiable deep-learning-based alternative to decision trees, combining soft feature selection with ensemble learning (Popov et al., 2019). Instead of traditional hard splits in decision trees, NODE applies learnable split functions to determine the best decision boundary. It uses oblivious decision trees as its building blocks, where all nodes at the same depth split on the same feature. This structure allows the model to be trained end-to-end using gradient descent while preserving the hierarchical feature interactions. The prediction for an input x is given by:

$$f(x) = \sum_{t=1}^{T} a_t \cdot g_t(x) \tag{15}$$

where T is the number of decision units, a_t are learnable weights, and $g_t(x)$ represents soft decision functions or output of the t-th oblivious decision tree. NODE is particularly effective for tabular data, as it balances the interpretability of tree-based models with the flexibility and scalability of DL.

Optimization model for vehicle reidentification

Recall that 997 candidate ReId pairs were generated after processing the high-resolution event dataset with ground-truth video recordings. Let y_{ij} denote each such candidate pair, the validity of which has to be assessed *via* the proposed vehicle reidentification framework. To this end, we propose an optimization model that minimizes the error E_{ij} associated with the pair of advance actuation i and stop bar actuation j. An ideal optimization model would yield as output 580 y_{ij} pairs which tally with the ground-truth ReId pairs. The parameters, decision variables, objective function, and constraints of the optimization model are formulated as follows:

Parameters

$$L_{ij} = \begin{cases} 1, \textit{if candidate ReId pair } (i,j) \textit{ belongs to the same lane} \\ 0, \textit{otherwise} \end{cases} \tag{16}$$

Decision variables

$$y_{ij} = \begin{cases} 1, \textit{if candidate ReId pair } (i,j) \textit{ is selected} \\ 0, \textit{otherwise} \end{cases} \tag{17}$$

Objective function

$$\min Z = \sum_i \sum_j y_{ij} E_{ij} \tag{18}$$

Constraints

$$t_{\min} \le t_{ij} \le t_{\max} \quad \forall (i,j) \tag{19}$$

$$E_{ij} = \left| t_{ij} - t_i^{pred} \right| \quad \forall (i,j) \tag{20}$$

$$\textstyle\sum_j y_{ij} \le 1 \quad \forall i \tag{21}$$

$$\textstyle\sum_i y_{ij} \le 1 \quad \forall j \tag{22}$$

$$y_{ij} \in \{1, 0\} \quad \forall (i,j) \tag{23}$$

$$L_{ij} \in \{1, 0\} \quad \forall (i,j) \tag{24}$$

$$y_{ij} = L_{ij} \quad \forall (i,j) \tag{25}$$

Equations (16) and (17) define the lane change parameter and decision variable associated with a candidate ReId pair i and j. The objective function, defined per Equation (18), aims to minimize the total error, considering only the selected ReId pairs. Equation (19) ensures that the predicted pairs satisfy the search space constraints. Let t_{ij} be the actual travel time between pairs i and j, whereas t_i^{pred} be the travel time predicted for i by the ML/DL model. Their absolute difference, as defined in Equation (20), yields the error E_{ij} between pairs i and j. Equations (21) and (22) ensure that each advance actuation is associated with at most one stop bar actuation and vice-versa. Equations (23) and (24) define the binary nature of the decision variable y_{ij} and the lane change parameter L_{ij}. Equation (25) ensures that the predicted ReId pairs are on the same lane.

Model hyperparameter tuning and performance evaluation

The inferred and test datasets with 5,391 and 580 respective ReId pairs were processed for feature extraction. Each model was trained on the inferred dataset for travel time prediction using a 5-fold cross-validation to avoid reliance on a single train-validation split. Hyperparameter tuning was performed in conjunction with the cross-validation process to identify the optimal model configuration. The inferred dataset was divided into five folds, where four folds were used for training a model, while the remaining fold was held out for validation during each iteration. A grid search was employed to systematically evaluate different hyperparameter configurations predefined for a model. Thus,

during each fold of cross-validation, the model was trained for every hyperparameter combination on the training folds, with performance evaluated separately on the held-out or the validation fold. The performance of each hyperparameter combination was then averaged over all validation folds, and the combination with the best average validation score yielded the optimal model configuration. Finally, the model was retrained on the entire inferred dataset using these optimal hyperparameters. The final model was then used to make travel time predictions on the test set, which remained completely unseen during hyperparameter tuning and model training. This overall procedure ensured that the model's performance on unseen test data was not influenced by the hyperparameter tuning or model training processes, thereby providing an unbiased evaluation of the model's predictive performance.

The travel time predicted by an ML/DL model is assessed using root mean squared error (RMSE), as represented in Equation (26). For performance evaluation of vehicle reidentification, let y_{ij}^{ground} represent the 580 ground-truth ReId pairs video-verified by research specialists, whereas y_{ij}^{pred} represent the ReId pairs predicted by the optimization model. We first define the metrics true positive (TP), false positive (FP), and false negative (FN) as follows: TP is the number of common ReId pairs in y_{ij}^{pred} and y_{ij}^{ground}; FP is the number of ReId pairs in y_{ij}^{pred} not present in y_{ij}^{ground}; FN is the number of ReId pairs in y_{ij}^{ground} not present in y_{ij}^{pred}. Based on these metrics, precision and recall are defined by Equations (27) and (28), respectively. Finally, Equation (29) defines the F1 score as a single number summary and the harmonic mean of precision and recall to conduct a cross-model comparison of the six ML/DL models for reidentification.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}\left(t_{ij} - t_i^{\text{pred}}\right)^2}{n}} \tag{26}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{27}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{28}$$

$$F1 = \frac{2{*}\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{29}$$

Results and discussions

Reidentification performance across models

Table 3 summarizes the results for travel time prediction and vehicle reidentification across the six selected ML/DL models: SVR, RF, XGBoost, FCNN, TabNet, and NODE.

Table 3. Model results for vehicle reidentification.

Models	$\text{RMSE}^{[V]}$	$\text{RMSE}^{[T]}$	TP	FP	FN	Precision	Recall	F1 score
SVR	0.9633	1.0460	529	**30**	51	**0.9463**	0.9121	0.9289
RF	0.8674	0.9801	530	37	50	0.9347	0.9138	0.9241
XGBoost	**0.8495**	0.9522	532	32	48	0.9433	0.9172	0.9301
FCNN	0.8789	0.9901	532	31	48	0.9449	0.9172	0.9309
TabNet	0.8675	0.9814	529	31	51	0.9446	0.9121	0.9281
NODE	0.8556	**0.9397**	**534**	31	**46**	0.9451	**0.9207**	**0.9328**

$\text{RMSE}^{[V]}$ = root mean squared error averaged over validation folds; $\text{RMSE}^{[T]}$ = root mean squared error on the test set.
Bold indicates the best result obtained for each metric.

While the reidentification performance was comparable across models, NODE yielded the best F1 score of 0.9328, with 94.5% precision and 92.1% recall. Although some individual metrics, such as validation RMSE and FPs, were marginally better for other models, NODE consistently performed well across all performance metrics, demonstrating the most balanced and robust performance overall. This balanced performance demonstrates NODE's suitability for vehicle reidentification, where it is critical to minimize both FPs and FNs. Similarly, FCNN showed competitive performance, with an F1 score of 0.9309, making it a close alternative to NODE. These results reinforce the effectiveness and general robustness of ML/DL models in capturing the variability of travel time for vehicle reidentification.

Analysis of travel time prediction

Table 3 illustrates NODE's superior performance for travel time prediction, with an average RMSE of 0.8556 s on the validation folds and 0.9397 s on the test set. The differences between validation and test RMSE values across models provide insights into their generalization capabilities. While XGBoost achieved the lowest RMSE on the validation set (0.8495 s), its test RMSE increased to 0.9522 s, indicating a drop in generalization performance. Similarly, RF, TabNet, and FCNN performed well on the validation folds but suffered a large performance drop on the test set. In contrast, NODE exhibited a relatively small increase in RMSE from validation to test, reinforcing its ability to maintain consistent performance across different data distributions.

We further explore why NODE, compared to other models, yielded better results for travel time prediction and vehicle reidentification. Figure 5 illustrates the relationship between ground truth and predicted travel time across all models, with each observation representing a ground truth ReId pair. The green circles represent the predicted TP reidentifications,

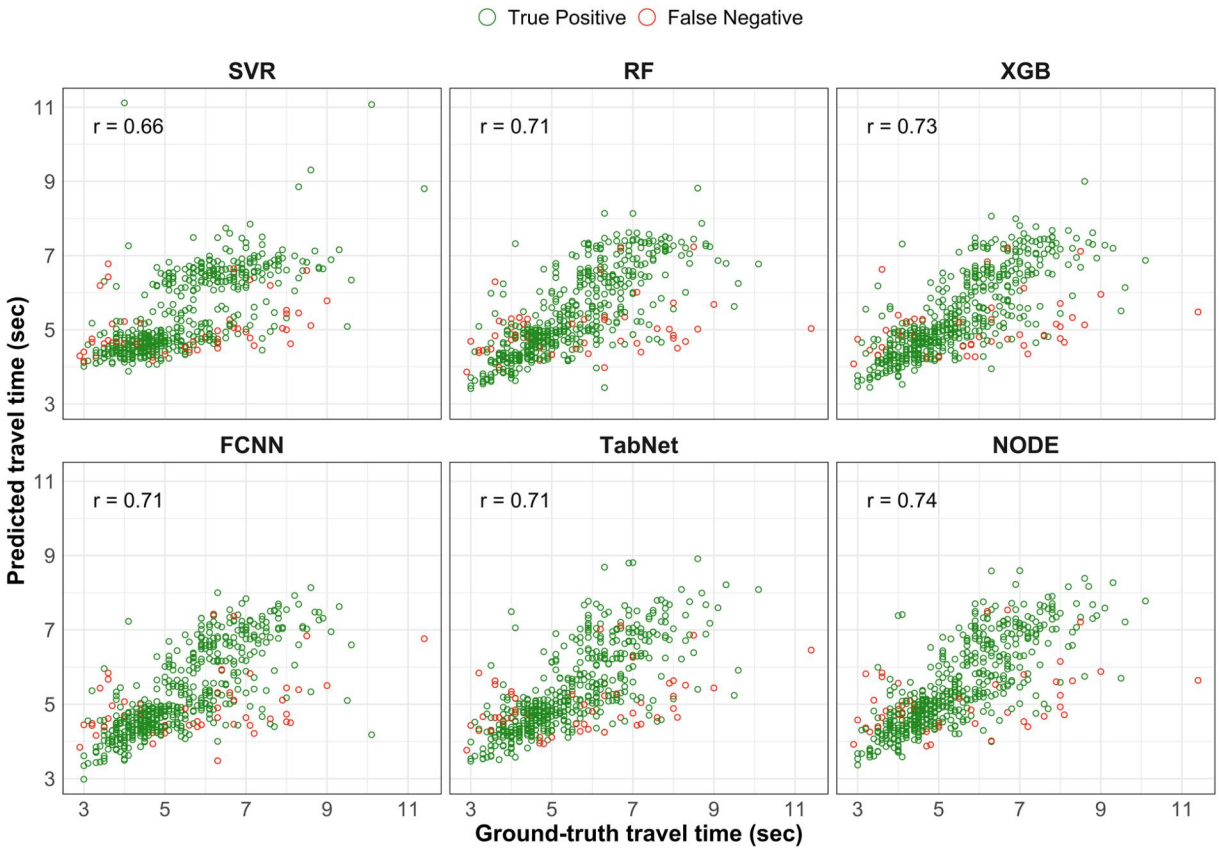

Figure 5. Comparison of ground-truth and predicted travel time across models.

while the red circles indicate FNs the model failed to reidentify. The correlation coefficient (r) indicates the strength of the linear association between the ground truth and predicted travel times.

NODE ($r = 0.74$) and XGBoost ($r = 0.73$) produced more consistent and reliable travel time estimates, which align with their lower RMSEs on the test set. The higher correlation and lower RMSEs in these models suggest that their errors are not only smaller in magnitude but also more uniformly distributed, evident from the tighter clustering of observations along the diagonal. Conversely, SVR exhibited the lowest correlation ($r = 0.66$) and highest RMSE on the test set due to greater dispersion of points along the diagonal, specifically at higher travel times. Figure 5 further reveals that FNs (red points) are more dispersed in lower-performing models–particularly in SVR, RF, and FCNN–contributing to their increased RMSE values. Overall, Figure 5 and Table 3 demonstrate NODE's ability to more accurately predict travel time on the unseen test set, highlighting its robustness over comparable attention-based and ensemble-based learning models to capture the required variability in travel time for vehicle reidentification.

Optimal hyperparameters and computational time

Table 4 presents the grid of hyperparameters with the total combinations tested for each ML/DL model. Given its larger number of hyperparameters to tune, XGBoost had the highest number of hyperparameter combinations tested, followed by TabNet, which required fine-tuning both transformer and decision step parameters. The model-specific optimal hyperparameter combinations yielding the best reidentification performance are in bold. For brevity, we discuss only the hyperparameters tuned for NODE, the best-performing model for vehicle reidentification.

Given the training dataset size and complexity, the optimal NODE model was configured with 16 trees, suggesting sufficient complexity to capture underlying patterns in travel time prediction while avoiding unnecessary redundancy in learned decision structures. A tree depth of 2 indicates a simpler, more interpretable model that preserves generalization capability without overfitting. For updating the model's parameters during training, the learning rate of 0.1 enabled stable yet efficient convergence and prevented slow optimization. Concurrently, a batch size of 32 balanced computational efficiency and model stability,

Table 4. Model hyperparameters.

Model	Hyperparameters	Values	Combinations
Support vector regression (SVR)	kernel	["**linear**", "rbf"]	24
	C	[**0.1**, 1, 10]	
	epsilon	[0.01, 0.1, 0.2, **0.5**]	
Random forest (RF)	n_estimators	[50, 100, **200**, 500]	144
	max_depth	[5, **10**, 15, 20]	
	min_samples_split	[2, 5, **10**]	
	min_samples_leaf	[1, **2**, 4]	
XGBoost (XGB)	n_estimators	[100, **200**, 300]	3888
	learning_rate	[0.01, **0.1**, 0.2]	
	max_depth	[2, **3**, 5, 7]	
	min_child_weight	[3, 5, **7**]	
	gamma	[0, 0.1, 0.2, **0.5**]	
	reg_alpha	[0, 0.1, **0.5**]	
	reg_lambda	[0, **0.1**, 0.5]	
Fully connected neural network (FCNN)	units	[**(64, 32)**, (128, 64), (64, 32, 16)]	27
	dropout	[**0.1**, 0.2, 0.3]	
	learning_rate	[**0.001**, 0.01, 0.1]	
TabNet	n_d	[8, 16, 24, **32**]	864
	n_a	[8, 16, **24**, 32]	
	n_steps	[**3**, 5, 7]	
	gamma	[1.0, 1.5, **2.0**]	
	lambda_sparse	[0.00001, **0.001**, 0.1]	
	mask_type	["**sparsemax**", "entmax"]	
Neural Oblivious Decision Ensemble (NODE)	num_trees	[**16**, 32, 64]	108
	tree_depth	[**2**, 3, 5]	
	learning_rate	[0.001, **0.01**]	
	batch_size	[**32**, 64]	
	Dropout	[0.0, **0.1**, 0.2]	

Bold indicates optimal hyperparameter combination.

Table 5. Computational time across models.

Model	Prediction (ms)	Reidentification (ms)
SVR	28.4	14.3
RF	21.5	14.4
XGBoost	2.2	14.5
FCNN	65.2	14.5
TabNet	18.5	14.4
NODE	3.9	14.5

Each model's computational time for prediction and reidentification is averaged over three runs.

minimizing gradient noise without excessively smoothing updates to the model's weights. Lastly, regularization using a dropout rate randomly deactivated 10% of neurons during each training step, preventing heavy reliance on specific neurons and resulting in more generalized feature representations.

Table 5 summarizes the computational performance of the proposed framework. After completing offline model training, the runtime for travel time prediction and subsequent reidentification was measured for each optimal model across three trials. Computational time for travel time prediction for the full test set (580 samples) ranged from 2.2 ms (XGBoost) to 65.2 ms (FCNN), with NODE requiring just 3.9 ms while achieving the best reidentification performance. The reidentification step, executed for each actuation at the advance detector, required approximately 14.4 ms consistently across all models. Model training time was not included in this analysis, since training typically occurs offline and varies substantially due to differences in hyperparameter search space–for instance, 3,888 combinations for XGBoost *vs.* 108 for NODE *vs.* 24 for SVR.

These findings–high reidentification accuracy and low computational cost–demonstrate that the proposed framework is robust and scalable for real-time processing and deployment, even at signalized intersections with high volume. Computational time assessments were performed using Python 3.10 on a high-performance workstation equipped with an AMD Ryzen 7 5800X CPU, 32 GB RAM, and dual NVIDIA RTX A4500 GPUs. However, the measured computation times for travel time prediction and vehicle reidentification tasks were consistently low (typically less than 20 ms per actuation), indicating that the framework's operation is lightweight and not dependent on GPU acceleration. This suggests the proposed framework can be readily deployed on standard computing infrastructure without requiring specialized hardware.

Performance comparison with state-of-the-art analytical models

The NODE-based optimization framework, yielding the best reidentification performance with low computational cost, was compared with three existing analytical methods for reidentifying vehicles between detectors. Methods proposed by Lu et al. (2015), Chen

Table 6. Reidentification performance comparison with state-of-the-art analytical methods.

Reidentification methods	TP	FP	FN	Precision	Recall	F1 score
Lu et al. (2015)*	469	55	111	0.8950	0.8086	0.8496
Chen et al. (2017)*	431	45	149	0.9055	0.7431	0.8163
Pudasaini et al. (2024)*	513	40	67	0.9277	0.8845	0.9056
NODE-based framework	**534**	**31**	**46**	**0.9451**	**0.9207**	**0.9328**

*Analytical method calibrated for best reidentification.
Bold indicates the best result obtained for each metric.

et al. (2017), and Pudasaini et al. (2024) were selected for performance comparison as these are among the most recent or prominent analytical methods for vehicle reidentification.

Pudasaini et al. (2024) relied on a priori knowledge of ideal travel times for stopping and running between detectors; sensitivity analysis carried out in the author's study estimated these times as 6.6 and 4.6 s, respectively. In Lu et al. (2015) and Chen et al. (2017), velocity was computed at the detector locations based on an a priori known effective vehicle length (assumed 25 ft in respective studies). Since an effective vehicle length was not calibrated a priori for the intersection approach in this study, we tested both methods by conducting a sensitivity analysis of vehicle lengths varying from 16 ft to 24 ft. Vehicle lengths of 19 ft in Chen et al. (2017) and 20 ft in Lu et al. (2015) yielded the best reidentification performance. The reidentification results from these calibrated analytical methods and the best-performing NODE model from this study are summarized in Table 6. All reidentification methods were evaluated using the same test dataset comprising 580 ground-truth vehicle ReId samples, ensuring a consistent and fair performance comparison.

NODE's F1 score of 0.9328, with 94.5% precision and 92.1% recall, demonstrated superior performance over the existing analytical methods for vehicle reidentification. Note that the comparatively high precision from the calibrated analytical methods is attributed to fewer samples of TPs. Moreover, a significantly high number of FNs contributed to a low recall, indicating that the analytical methods miss a significant proportion of the actual ground-truth ReId pairs. We hypothesize that constraining the search space is one of the factors influencing recall in the analytical methods. Chen et al. (2017), for instance, constrains the search space conservatively for each actuation at the advance detector based on the minimum and maximum values of travel time between detectors. Similarly, Lu et al. (2015) utilize the velocity measurement at advance and entrance detectors but does not constrain the search space to reidentify actuation pairs. The NODE-based optimization framework, in contrast, applies a consistently constrained search space across all advance detector actuations, allowing for a more accurate reidentification while maintaining high precision.

Another critical limitation of the analytical methods is their dependence on a priori knowledge of key parameters, restricting their transferability across different intersections. Lu et al. (2015) and Chen et al. (2017) rely on an assumed effective vehicle length, necessitating its manual calibration for each intersection. An incorrect assumption of this parameter can heavily skew reidentification accuracy, as velocity estimation at detectors in these methods is directly influenced by vehicle length. Similarly, Pudasaini et al. (2024) depend on predefined ideal travel times for stopping and running vehicles, requiring site-specific calibration for reliable reidentification. In contrast, NODE's ability to learn from data without requiring manual calibration of vehicle length or travel times makes the proposed ML-based optimization framework a more adaptable and scalable solution for vehicle reidentification across different intersection configurations.

Limitations and practical implications

While this study focused on reidentification at a single intersection approach, enhancing the scalability of the proposed framework across various intersection types and traffic conditions remains an important avenue for future research. Predicting travel time using advanced ML/DL models eliminates the need for site-specific calibration, suggesting that this framework is indeed transferable to different traffic environments, detector configurations, and signal control systems. However, additional validation using a large sample across multiple locations would strengthen its robustness under varying congestion levels, lane configurations, and vehicle mix. Furthermore, despite its high accuracy, the framework could face challenges in scenarios with extreme congestion or frequent lane changes, where the assumption of minimal lane changes during the yellow onset may not hold. Future work could address these limitations and also explore the integration of additional data sources, such as GPS and vehicle-to-infrastructure (V2I) communication, to further enhance reidentification accuracy, particularly in complex traffic scenarios involving lane changes, queue spillback, and multimodal interactions.

Despite the aforementioned limitations, this study demonstrated the effectiveness of integrating an optimization framework with ML/DL-based travel time prediction for reidentifying vehicles at a signalized

intersection using high-resolution event data. The NODE-based framework outperformed state-of-the-art analytical, conventional ML, and comparable DL models in accurately tracking vehicles over the advance and stop bar loop detectors that generate non-visual detection data. Unlike traditional analytical methods that rely on pre-calibrated parameters such as effective vehicle length or predefined travel time assumptions (Chen et al., 2017; Ding et al., 2016; Liu et al., 2017; Lu et al., 2015; Pudasaini et al., 2024; Ren et al., 2016; Wu et al., 2013), our framework adapts to varying traffic conditions without manual tuning, making it more scalable across different intersection and detector configurations. More importantly, the proposed framework's low false alarm rate and high recall ensure minimal misclassification while maximizing true reidentification pairs. These findings emphasize the importance of leveraging advanced learning models to replace conventional analytical methods that often struggle with adaptability and generalization for reidentifying vehicles.

Beyond the methodological advancements, this study offers practical contributions for both real-time and retrospective applications in assessing intersection safety and monitoring traffic operations. First, this study demonstrates that highly accurate vehicle reidentification can be achieved using high-resolution event data from widely deployed non-visual technologies like loop detectors, without requiring video footage or site-specific calibration. This practicality enables transportation agencies to reliably track vehicles at an intersection approach using data they already collect. The framework supports real-time applications, including identifying vehicles at risk of entering during yellow or red phases and informing dynamic signal control strategies, such as extending green time to reduce the likelihood of red light running or applying all-red extensions for late entries. Moreover, the framework facilitates large-scale retrospective safety analysis by generating vehicle-level behavioral data that can be used to evaluate dilemma zone risks and assess safety countermeasures with greater accuracy. Given that vehicle actuations typically occur every second on busy arterials, the proposed framework's low computational cost offers sufficient processing speed and scalability for real-time deployment. More importantly, its seamless compatibility with existing detector infrastructure and high-resolution data formats ensures practical, cost-effective integration into current ITS environments for live intersection monitoring and data-informed decision support.

Conclusions

Most research on vehicle reidentification in the existing literature focuses on traffic surveillance using images captured from multiple cameras with non-overlapping views. In contrast to such reidentification, this study addressed vehicle reidentification using non-visual detection data, focusing on obtaining driver behavior data by tracking vehicles at signalized intersection approaches. In the conventional layout of advance and stop bar detectors, vehicles approaching the intersection are detected at discrete locations but not reidentified over the approach area. Despite the uniformity in detector installation layout and the abundance of high-resolution event data across many transportation agencies, methodological advancements are limited in reidentifying vehicles at an intersection approach. Existing analytical attempts at reidentifying vehicles using such data relied on a priori calibrated parameters, limiting their applicability and transferability to new intersections, detector configurations, and traffic dynamics. This study formulated a novel ML-based optimization framework for reidentifying vehicles between the advance and stop bar detectors using non-visual detection data.

The proposed framework comprised two major components: prediction of travel time between detectors and vehicle reidentification based on actuations over these detectors. The travel time prediction was tested with six advanced ML and DL models: SVR, RF, XGBoost, FCNN, TabNet, and NODE. A novel optimization model then utilized these predicted travel times for reidentifying vehicles approaching an intersection. The reidentification framework was tested on a major intersection approach in Phoenix, Arizona. The NODE-based optimization framework yielded the best travel time prediction and vehicle reidentification results, with an F1 score of 0.9328, 94.5% precision, and 92.1% recall. The low false alarm and high recall rates of the proposed framework further demonstrated superiority over three calibrated analytical reidentification methods. The major contribution of this study is proposing a highly accurate and robust data-driven framework for vehicle reidentification using non-visual detection data, addressing a critical gap in the existing literature. Also, the study provides a scalable and automated alternative to labor-intensive manual methods for collecting valuable driver behavior data, particularly during safety-critical yellow onset periods.

Our future work will focus on applying the proposed vehicle reidentification framework for analyzing dilemma zone boundaries and drivers' acceleration/deceleration behaviors while approaching an

intersection. While this study demonstrated the effectiveness of integrating an optimization framework with ML/DL-based travel time prediction for non-visual vehicle reidentification, enhancing the framework's scalability across various intersection types and traffic conditions remains an important avenue for future research. Despite its high accuracy, the framework could face challenges in scenarios with extreme congestion or frequent lane changes. Integrating additional data sources, such as GPS and V2I communication, into the proposed framework could enhance the reidentification accuracy for real-time ITS applications, particularly in complex traffic scenarios involving lane changes, queue spillback, and multimodal interactions.

Acknowledgments

The authors thank the City of Phoenix for partial funding and data support in this study. Thanks to Simon Ramos, Robert Kyser, and Eric Hernandez for facilitating access to VPN, high-resolution event data, and video recordings. Thanks to Armstrong Aboah for his insights and feedback on the methodology proposed in this study. Special thanks to Cristina Valencia Reyes, Cynthia Navarro, and Will Reuter for their assistance in manually verifying detection matches from videos and inferring match pairs.

Author contributions

Pramesh Pudasaini: Conceptualization, Methodology, Data curation, Formal analysis, Investigation, Software, Validation, Visualization, Writing – original draft. Henrick Haule: Conceptualization, Investigation, Supervision, Writing – original draft, Writing – review & editing. Yao-Jan Wu: Conceptualization, Funding acquisition, Project administration, Supervision, Resources, Writing – review & editing.

Disclosure statement

None.

Funding

The authors thank the City of Phoenix for partial funding and data support in this study.

ORCID

Pramesh Pudasaini http://orcid.org/0000-0002-4238-1097
Henrick Haule http://orcid.org/0000-0003-0455-4134
Yao-Jan Wu http://orcid.org/0000-0002-0456-7915

References

Abdelhalim, A., Abbas, M., Kotha, B. B., & Wicks, A. (2021, September 19–22). *A framework for real-time traffic trajectory tracking, speed estimation, and driver behavior calibration at urban intersections using virtual traffic lanes* [Paper presentation]. 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, United States. https://doi.org/10.1109/ITSC48978.2021.9564525

Amiri, A., Kaya, A., & Keceli, A. S. (2024). A comprehensive survey on deep-learning-based vehicle re-identification: Models, data sets and challenges. *arXiv Preprint arXiv:2401.10643*.

Arik, S. Ö., Pfister, T. (2021). Tabnet: Attentive interpretable tabular learning. *Proceedings of theConference on Artificial Intelligence*. AAAI.

Brackstone, M., & McDonald, M. (1999). Car-following: A historical review. *Transportation Research Part F: Traffic Psychology and Behaviour*, *2*(4), 181–196. https://doi.org/10.1016/S1369-8478(00)00005-X

Chandler, B. E., Myers, M., Atkinson, J. E., Bryer, T., Retting, R., Smithline, J., Trim, J., Wojtkiewicz, P., Thomas, G. B., Venglar, S. P., Sunkari, S. R., Malone, B. J., & Izadpanah, P. (2013). *Signalized intersections informational guide* (No. FHWA-SA-13-027). United States: Federal Highway Administration. Office of Safety.

Chen, T., & Guestrin, C. (2016, August 13–17). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, San Francisco, California, USA*. New York, NY, United States: Association for Computing Machinery.

Chen, P., Yu, G., Wu, X., Ren, Y., & Li, Y. (2017). Estimation of red-light running frequency using high-resolution traffic and signal data. *Accident; Analysis and Prevention*, *102*, 235–247. https://doi.org/10.1016/j.aap.2017.03.010

Ding, C., Wu, X., Yu, G., & Wang, Y. (2016). A gradient boosting logit model to investigate driver's stop-or-run behavior at signalized intersections using high-resolution traffic data. *Transportation Research Part C: Emerging Technologies*, *72*, 225–238. https://doi.org/10.1016/j.trc.2016.09.016

Do, W., Saunier, N., & Miranda-Moreno, L. (2023). An empirical analysis of the effect of pedestrian signal countdown timer on driver behavior at signalized intersections. *Accident; Analysis and Prevention*, *180*, 106906. https://doi.org/10.1016/j.aap.2022.106906

Gazzah, S., Essoukri, N., Amara, B. (2017). Vehicle re-identification in Camera networks: A review and new perspectives. *Proceedings of the ACIT'2017 The International Arab Conference on Information Technology, Yassmine Hammamet, Tunisia*.

Hu, W., Zhan, H., Shivakumara, P., Pal, U., & Lu, Y. (2024). TANet: Text region attention learning for vehicle re-identification. *Engineering Applications of Artificial Intelligence*, *133*, 108448. https://doi.org/10.1016/j.engappai.2024.108448

Jeng, S. T., Tok, Y. C. A., & Ritchie, S. G. (2010). Freeway corridor performance measurement based on vehicle reidentification. *IEEE Transactions on Intelligent Transportation Systems*, *11*(3), 639–646. https://doi.org/10.1109/TITS.2010.2049105

Kandiboina, R., Knickerbocker, S., Bhagat, S., Hawkins, N., & Sharma, A. (2024). Exploring the efficacy of large-scale

connected vehicle data in real-time traffic applications. *Transportation Research Record: Journal of the Transportation Research Board*, *2678*(5), 651–665. https://doi.org/10.1177/03611981231191512

Khan, S. D., & Ullah, H. (2019). A survey of advances in vision-based vehicle re-identification. *Computer Vision and Image Understanding*, *182*, 50–63. https://doi.org/10.1016/j.cviu.2019.03.001

Li, Z., & Wei, H. (2013). Modeling dynamics of dilemma zones by formulating dynamical contributing factors with video-observed trajectory data. *Procedia - Social and Behavioral Sciences*, *80*, 880–900. https://doi.org/10.1016/j.sbspro.2013.05.048

Lin, W. H., & Tong, D. (2011). Vehicle re-identification with dynamic time windows for vehicle passage time estimation. *IEEE Transactions on Intelligent Transportation Systems*, *12*(4), 1057–1063. https://doi.org/10.1109/TITS.2011.2140318

Liu, H. X., Davis, G. A., Shen, S., Di, X., & Chatterjee, I. (2017). *Estimation of crossing conflict at signalized intersection using high-resolution traffic data* (Report number: MnDOT 2017-08). Minnesota Department of Transportation.

Long, G. (2000). Acceleration characteristics of starting vehicles. *Transportation Research Record: Journal of the Transportation Research Board*, *1737*(1), 58–70. https://doi.org/10.3141/1737-08

Lu, G., Wang, Y., Wu, X., & Liu, H. X. (2015). Analysis of yellow-light running at signalized intersections using high-resolution traffic data. *Transportation Research Part A: Policy and Practice*, *73*, 39–52. https://doi.org/10.1016/j.tra.2015.01.001

Ning, X., Qian, J., Dong, L., Yan, D., & Wang, C. (2025). Vehicle re-identification based on suppressing contaminated features. *Journal of Intelligent Transportation Systems*, 1–15. https://doi.org/10.1080/15472450.2025.2489657

Popov, S., Morozov, S., & Babenko, A. (2019). Neural oblivious decision ensembles for deep learning on tabular data. *arXiv Preprint arXiv:1909.06312*.

Pudasaini, P., Haule, H., & Wu, Y.-J. (2024). Empirical analysis of dilemma zone using high-resolution event data. *Transportmetrica B: Transport Dynamics*, *12*(1), 2379376. https://doi.org/10.1080/21680566.2024.2379376

Pudasaini, P., Haule, H., & Wu, Y.-J. (2025). Modeling dilemma zone at urban signalized intersections using crowdsourced trajectory data. *Accident; Analysis and Prevention*, *219*, 108070. https://doi.org/10.1016/j.aap.2025.108070

Pudasaini, P., Karimpour, A., & Wu, Y.-J. (2023). Real-time queue length estimation for signalized intersections using single-channel advance detector data. *Transportation Research Record: Journal of the Transportation Research Board*, *2677*(7), 144–156. https://doi.org/10.1177/03611981221151066

Pudasaini, P., Pathivada, B. K., & Wu, Y. J. (2025). Modeling influence area for urban signalized intersections using crowdsourced trajectory data. *Transportation Research Record*. https://doi.org/10.1177/03611981251334625

Qian, Y., Barthélemy, J., Du, B., & Shen, J. (2024). Paying attention to vehicles: A systematic review on transformer-based vehicle re-identification. *ACM transactions on multimedia computing communications and applications*. ACM.

Rahman, M., Kang, M.-W., & Biswas, P. (2021). Predicting time-varying, speed-varying dilemma zones using machine learning and continuous vehicle tracking. *Transportation Research Part C: Emerging Technologies*, *130*, 103310. https://doi.org/10.1016/j.trc.2021.103310

Ren, Y., Wang, Y., Wu, X., Yu, G., & Ding, C. (2016). Influential factors of red-light running at signalized intersection and prediction using a rare events logistic regression model. *Accident; Analysis and Prevention*, *95*(Pt A), 266–273. https://doi.org/10.1016/j.aap.2016.07.017

Urbanik, T., Tanaka, A., Lozner, B., Lindstrom, E., Lee, K., Quayle, S., Beaird, S., Tsoi, S., Ryus, P., & Gettman, D. (2015). *Signal timing manual* (Vol. 1). Transportation Research Board.

Wang, H., Hou, J., & Chen, N. (2019). A survey of vehicle re-identification based on deep learning. *IEEE Access*, *7*, 172443–172469. https://doi.org/10.1109/ACCESS.2019.2956172

Wang, J., Indra-Payoong, N., Sumalee, A., & Panwai, S. (2014). Vehicle reidentification with self-adaptive time windows for real-time travel time estimation. *IEEE Transactions on Intelligent Transportation Systems*, *15*(2), 540–552. https://doi.org/10.1109/TITS.2013.2282163

Wu, X., Vall, N. D., Liu, H. X., Cheng, W., & Jia, X. (2013). Analysis of drivers' stop-or-run behavior at signalized intersections with high-resolution traffic and signal event data. *Transportation Research Record: Journal of the Transportation Research Board*, *2365*(1), 99–108. https://doi.org/10.3141/2365-13

Xiong, Z., Li, M., Ma, Y., & Wu, X. (2021). Vehicle re-identification with image processing and car-following model using multiple surveillance cameras from urban arterials. *IEEE Transactions on Intelligent Transportation Systems*, *22*(12), 7619–7630. https://doi.org/10.1109/TITS.2020.3006047

Yi, X., Wang, Q., Liu, Q., Rui, Y., & Ran, B. (2025). Advances in vehicle re-identification techniques: A survey. *Neurocomputing*, *614*, 128745. https://doi.org/10.1016/j.neucom.2024.128745

Zapletal, D., Herout, A. (2016). Vehicle re-identification for automatic video traffic surveillance. *Proceedings of theConference on Computer Vision and Pattern Recognition Workshops*. IEEE.

Zhang, C., Chen, B. Y., Lam, W. H., Ho, H., Shi, X., Yang, X., Ma, W., Wong, S., & Chow, A. H. (2022). Vehicle re-identification for lane-level travel time estimations on congested urban road networks using video images. *IEEE Transactions on Intelligent Transportation Systems*, *23*(8), 12877–12893. https://doi.org/10.1109/TITS.2021.3118206

Zhang, C., Ho, H., Lam, W. H., Ma, W., Wong, S. C., & Chow, A. H. (2023). Lane-based estimation of travel time distributions by vehicle type via vehicle re-identification using low-resolution video images. *Journal of Intelligent Transportation Systems*, *27*(3), 364–383. https://doi.org/10.1080/15472450.2022.2027767