

Article

High-Resolution Traffic Sensing with Probe Autonomous Vehicles: A Data-Driven Approach

Wei Ma ¹  and Sean Qian ^{2,3,*} 

¹ Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hong Kong, China; wei.w.ma@polyu.edu.hk

² Department of Civil and Environmental Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA

³ H. John Heinz III Heinz College, Carnegie Mellon University, Pittsburgh, PA 15213, USA

* Correspondence: seanqian@cmu.edu

Abstract: Recent decades have witnessed the breakthrough of autonomous vehicles (AVs), and the sensing capabilities of AVs have been dramatically improved. Various sensors installed on AVs will be collecting massive data and perceiving the surrounding traffic continuously. In fact, a fleet of AVs can serve as floating (or probe) sensors, which can be utilized to infer traffic information while cruising around the roadway networks. Unlike conventional traffic sensing methods relying on fixed location sensors or moving sensors that acquire only the information of their carrying vehicle, this paper leverages data from AVs carrying sensors for not only the information of the AVs, but also the characteristics of the surrounding traffic. A high-resolution data-driven traffic sensing framework is proposed, which estimates the fundamental traffic state characteristics, namely, flow, density and speed in high spatio-temporal resolutions and of each lane on a general road, and it is developed under different levels of AV perception capabilities and for any AV market penetration rate. Experimental results show that the proposed method achieves high accuracy even with a low AV market penetration rate. This study would help policymakers and private sectors (e.g., Waymo) to understand the values of massive data collected by AVs in traffic operation and management.

Keywords: autonomous vehicle; LiDAR; camera; state estimation; traffic sensing; data-driven; traffic flow; NGSIM



Citation: Ma, W.; Qian, S. High-Resolution Traffic Sensing with Probe Autonomous Vehicles: A Data-Driven Approach. *Sensors* **2021**, *21*, 464. <https://doi.org/10.3390/s21020464>

Received: 18 December 2020

Accepted: 5 January 2021

Published: 11 January 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As the combination of a wide spectrum of cutting-edge technologies, autonomous vehicles (AVs) are destined to fundamentally change and reform the whole mobility system [1]. AVs have great potentials in improving safety and mobility [2–4], reducing fuel consumption and emission [5,6], and redefining civil infrastructure systems, such as road networks [7–9], parking spaces [10–12], and public transit systems [13,14]. Over the past two decades, many advanced driver assistance systems (ADAS) (e.g., lane keeping, adaptive cruise control) have been deployed in various types of production vehicles. Currently, both traditional car manufacturers and high-tech companies are competing to lead full autonomy technologies. For example, Waymo's AVs alone are driving 25,000 miles every day in 2018 [15], and there have been commercialized AVs operating in multiple cities by Uber [16].

Despite the rapid development of AVs technologies, there is still a long way to reach the full autonomy and to completely replace all conventional vehicles with AVs. We will witness a long period over which AVs and conventional vehicles co-exist on public roads. How to sense, model and manage the mixed transportation systems presents a great challenge to public agencies. To the best of our knowledge, most current studies view AVs as controllers and focus on modeling and managing the mixed traffic networks [17]. For example, novel system optimal (SO) and user equilibrium (UE) models are established to include AVs [18,19], coordinated intersections are proposed to improve the traffic

throughput [20–22], vehicle platooning strategies are developed to reduce highway congestion [23,24], and AVs can also complement conventional vehicles to solve last-mile problems [25,26]. However, there is a lack of studies in traffic sensing methods for the mixed traffic networks.

In this paper, we advocate the great potentials of AVs as moving observers in high-resolution traffic sensing. We note that traffic sensing with AVs in this paper is different from perception of AVs [27]. The perception of AV is the key to the safe and reliable AVs, and it refers to the ability of AVs in collecting information and extracting relevant knowledge from the environment through various sensors [28], while traffic sensing with AVs refers to estimating the traffic conditions, such as flow, density and speed using the information perceived by AVs [29]. To be precise, traffic sensing with AVs is built on top of the perception technologies on AV, and in this paper, we will discuss the impact of different perception technologies on traffic sensing.

In fact, a fleet of autonomous vehicles (AVs) can serve as floating (or probe) sensors, detecting and tracking the surrounding traffic conditions to infer traffic information when cruising around the roadway network. Enabling traffic sensing with AVs is cost effective. The AVs equipped with various sensors and data analytics capabilities may be costly. While costly, those sensors and data are used primarily to detect and track adjacent objects to enable safe AV driving in the first place. Therefore, there is no additional overhead cost of these data collections for traffic sensing, since it is a secondary use.

High-resolution traffic sensing is central to traffic management and public policies. For instance, local municipalities would need information regarding how public space (e.g., curbs) is being utilized to set up optimal parking duration limits; metropolitan planning agencies would need various types of traffic/passenger information, including travel speed, traffic density and traffic flow by vehicle classifications, as well as pedestrians and cyclists. In addition, non-emergent and emergent incidents are reported by citizens through the 911 system, respectively. Automated traffic sensing, both historical and in real time, can complement those systems to enhance their timeliness, accuracy and accessibility. In general, accurate and ubiquitous information of infrastructure and usage patterns in public space is currently missing.

By leveraging the rich data collected through AVs, we are able to detect and track various objects in transportation networks. The objects include, but are not limited to, moving vehicles by vehicle classifications, parked vehicles, pedestrians, cyclists, signage in public space. When all those objects in high spatio-temporal resolutions are being continuously tracked, those data can be translated to useful traffic information for public policies and decision making. The three key features of traffic sensing based on autonomous vehicles sensors are: inexpensive, ubiquitous and reliable. Those data are collected by automotive manufacturers for guiding autonomous driving in the first place, which promises great scalability in this approach. With minimum additional efforts, the same data can be effectively translated into information useful for the community. For instance, how much time in public space at a particular location is utilized by different classifications of vehicles and by what travel modes, respectively? Can we effectively evaluate the accessibility, mobility and safety of the mobility networks? The sensing coverage will become ubiquitous in the near future, provided with an increasing market share of autonomous vehicles. Data acquired from individual autonomous vehicles can be compared, validated, corrected, consolidated, generalized and anonymized to retrieve the most reliable and ubiquitous traffic information. In addition, this paper for traffic sensing also implies the future possibility of interventions for effective and timely traffic management. It enables real-time traffic monitoring, potentially safer traffic operation, faster emergency response, and smarter infrastructure management.

The rest of this paper focuses on a critical problem to estimate the fundamental traffic state variables, namely, flow, density and speed, in high resolution, to demonstrate the sensing power of AVs. In addition to traffic sensing, there are many aspects and data in community sensing that could be explored in the near future. For example, perception of

AVs can be used for monitoring urban forest health, air quality, street surface roughness and many other applications of municipal asset management [30–33].

Traffic state variables (e.g., flow, density and speed) play a key role in traffic operation and management. Over the past several decades, traffic state estimation (TSE) methods have been developed for not only stationary sensors (i.e., Eulerian data) but also moving observers (i.e., Lagrangian data) [34]. Stationary sensors, including loop detectors, cameras and radar, monitor the traffic conditions at a fixed location. Due to the high installation and maintenance cost, the stationary sensors are usually sparsely installed in the network, and hence the collected data are not sufficient for the practical traffic operation and management [35]. Data collected by moving observers (e.g., probe vehicles, ride-sourcing vehicles, unmanned aerial vehicles, mobile phones) have a better spatial coverage and hence it enables cost-effective TSE in large-scale networks [36]. Though the TSE method with moving observer dates back to 1954 [37], recent advances in communication and Internet of Things (IoT) technologies have catalyzed the development and deployment of various moving observers in the real world. Readers are referred to Seo et al. [29], Wang and Papageorgiou [38] for a comprehensive review of existing TSE models.

To highlight our contributions, we present studies that are closely related to this paper. The moving observers can be categorized into four types: originally defined moving observers, probe vehicles (PVs), unmanned aerial vehicles (UAVs) and AVs. Their characteristics and related TSE models are presented as follows:

- Originally defined moving observers. The moving observer method for TSE was originally proposed by Wardrop and Charlesworth [37]. The proposed method requires a probe vehicle to transverse along the road and count the number of slower vehicles overtaken by the probe vehicle and the number of faster vehicles which overtake the probe vehicle [39]. Though the setting of the originally defined moving observers is too ideal for practice, it enlightened us on the value of using Lagrangian data for TSE.
- PVs. The PVs refer to all the vehicles that can be geo-tracked, and it includes, but is not limited to, taxis, buses, trucks, connected vehicles, ride-sourcing vehicles [40]. The PV data have great advantages in estimating speed, while it hardly contains density/flow information. Studies have explored the sensing power of PVs [41]. PV data are usually used to complement stationary sensor data to enhance the traffic state estimation [42,43]. PVs with spacing measurement equipment can estimate traffic flow and speed simultaneously [44–47].
- UAVs. By flying over the roads and viewing from top-view perspectives, UAVs are able to monitor a segment of road or even the entire network [48–50]. UAVs have the advantage of better spatial coverage, while extra purchase of UAVs and the corresponding maintenance cost are required. Traffic sensing with UAV has been extensively studied in recent years, including vehicle identification algorithms [50–52], sensing frameworks [53,54], and UAV routing mechanisms [55,56].
- AVs. AVs can be viewed as probe vehicles equipped with more sensors and hence have better perception capabilities. Not only can the AV itself be geo-tracked, the vehicles surrounded by AVs can also be detected and tracked. AVs also share some similarities with UAVs because AVs can scan a continuous segment of road. We believe that AVs fall in between the PVs and UAVs, and hence existing TSE methods can hardly be applied to AVs. Furthermore, there are few studies on TSE with AVs. Moreover, Chen et al. [57] presents a cyber-physical system to model the traffic flow near AVs based on flow theory, while the TSE for the whole road is not studied. Recently, Uber ATG conduct an experiment to explore the possibility of TSE using AVs [58] but no rigorous quantitative analysis is provided.

The characteristics of TSE methods using different sensors are compared in Table 1. One can see that AVs have unique characteristics that are different from any other moving observers. Given the unique characteristics of AVs, there is a great need to study the AV-based TSE methods. However, as discussed above, this research area is still under-explored.

In view of this, we develop a data-driven framework that estimates high-resolution traffic state variables, namely flow, density and speed using the massive data collected by AVs. The framework clearly defines the task of TSE with AVs involved and considers different perception levels of AVs. A two-step TSE method is proposed under a low AV market penetration rate. While this paper focuses on the road level traffic state estimation, the proposed approach could be further extended to the network-wide TSE, which is left for future research. The main contributions of this paper are summarized as follows:

- We firstly raise and clearly define the problem of TSE with multi-source data collected by AVs.
- We discuss the functionality and role of various AV sensors in traffic state estimation. The sensing power of AVs is categorized into three levels.
- We rigorously formulate the AV-based TSE problem. A two-step framework that leverages the sensing power of AVs to estimate high-resolution traffic state variables is developed. The first step directly translates the information observed by AVs to spatio-temporal traffic states and the second step employs data-driven methods to estimate the traffic states that are not observed by AVs. The proposed estimation methods are data-driven and consistent with the traffic flow theory.
- The next generation simulation (NGSIM) data are adopted to examine the accuracy and robustness of the proposed framework. Experimental results are compelling, satisfactory and interpretable. Sensitivity analysis regarding AV penetration rate, sensor configuration, and perception accuracy will also be studied.

Table 1. Comparisons among TSE methods using different sensors.

	Stationary	Probe Vehicles (PVs)			UAVs
	Detectors	Conventional PVs	PVs with Spacing Measurement	AVs	
Installed Sensors	Loop detectors, cameras	GPS	GPS, LRR	GPS, LRR, LiDAR, cameras	GPS, cameras
Raw Data Collected	Vehicle counts over time	Trajectory of the PVs	Trajectory of the PVs and their first preceding vehicles	Trajectory of the AVs and their first preceding vehicles; locations (or trajectories) of all the surrounding vehicles	Birdviews of all vehicle locations
Is TSE possible?	Yes	No, only speed can be estimated	Yes	Yes	Yes
Cost-effective	No	Yes	Yes	Yes	No
Easy to Deploy	No	Yes	Yes	Yes	Yes
Required Market Penetration Rate for TSE	N/A	N/A	High	Low	Low
Literature	Wang and Papageorgiou [38], Thai and Bayen [59]	O’Keeffe et al. [41], Herring et al. [60], Yu et al. [61]	Wilby et al. [44], Seo et al. [45,46], Fountoulakis et al. [47]	This paper	Puri [48], Kanistras et al. [49], Ke et al. [50]

GPS: The Global Positioning System. LRR: Long Range Radar. LiDAR: Light Detection and Ranging.

Since TSE with AV data has not been explored in previous studies, this paper is probably the first attempt to rigorously tackle this problem. To this end, we first review the existing AV technologies that can contribute to traffic sensing, then we rigorously formulate the TSE problem. Finally, we propose and examine the solution framework. An overview of the paper structure is presented in Figure 1.

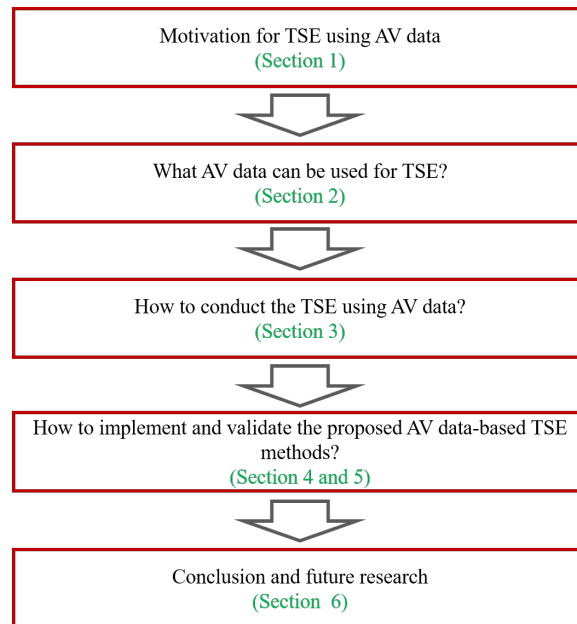


Figure 1. An overview of the paper structure.

Section 2 discusses the sensing power of AVs. Section 3 rigorously formulates the high-resolution TSE framework with AVs, followed by a discussion of the solution algorithms in Section 4. In Section 5, numerical experiments are conducted with NGSIM data to demonstrate the effectiveness of the proposed framework. Lastly, conclusions are drawn in Section 6.

2. Sensing Power of Autonomous Vehicles

In order to prepare for the rigorous formulation of the AV-based TSE framework, we first discuss different levels of AV perception capabilities and how they associate with traffic sensing in this section. We discuss various sensors installed on AVs and their relation to traffic sensing. Analogous to the automation level definitions from the Society of Automotive Engineers (SAE), we define three sensing levels of AVs. Lastly, we discuss a conceptual data center for processing the sensing data.

2.1. Sensors

In this section, we discuss different types of sensors used for AV perception and their potential usage for traffic sensing. Sensors for perception that are mounted on AVs include, but are not limited to, camera, stereo vision camera, LiDAR, radar and sonar [28].

A camera can detect shapes and colors, so it is widely used for object detection (e.g., signals, pedestrians, vehicles and lane marks). Due to its low cost, multiple cameras can be mounted on a single AV. Theoretically, studies have shown that camera data can be used for object detection, tracking and traffic sensing [62,63]. In practice, camera image does not contain depth (distance) information, the localization of vehicles is challenging when using a single camera. On the modern AV prototypes, cameras are usually fused with stereo vision camera system or LiDAR to perceive the surrounding environments. In particular, the shape and color information obtained from camera are essential for object tracking [64,65]. Stereo vision camera refers to a device with two or more cameras horizontally mounted. Stereo vision camera is able to obtain the depth information of each pixel from the slightly different images taken by its cameras.

Light detection and ranging (LiDAR) uses the pulsed laser beam to measure the distance to the detected object. LiDAR can also obtain the 3D shape of the detected object. The LiDAR used on AVs is typically 360°, and the detection range varies from 30 to 150 m, depending on makers, detection algorithms and weather conditions. Both LiDAR and stereo vision camera can be used for vehicle detection and 3D mapping. The system latency

(time delay for processing the retrieved data) of stereo vision camera is higher than LiDAR, though the price of stereo vision camera is much cheaper [27]. Theoretically, either the LiDAR or stereo vision camera can be used to build the full AV perception system, while currently, most AVs use LiDAR as the primary sensor.

There are two types of radar-mounted on AVs. The short-range radar (SRR) is typically used for blind spot detection, parking assist and collision warning. The range for SRR is around 20 m [66]. Similarly, sonar, with its limited detection range (3 to 5 m), is also frequently used for blind spot detection and parking assist. Neither of the two sensors are considered as appropriate sensors for traffic sensing. In contrast, the long-range radar (LRR), which is primarily used for adaptive cruise control, can be potentially used for traffic sensing. The range for LRR is around 150 m and it is dedicated to detecting the preceding vehicle in its current lane.

To conclude, Table 2 summarizes a list of sensors that can be potentially used for traffic sensing based on Van Brummelen et al. [27], Thakur [67].

Table 2. Sensors used for traffic sensing.

Sensors	Usage	Range
Camera	Surrounding vehicle detection/tracking, lane detection	20~60 m
Stereo vision camera	Surrounding vehicle detection/tracking, 3D mapping	20~60 m
LiDAR	Surrounding vehicle detection/tracking, 3D mapping	30~150 m
Long-range radar	Preceding vehicle detection	150 m

2.2. Levels of Perception

In this section, we discuss how to categorize the sensing power of AVs with sensors listed in Table 2. The Society of Automotive Engineers (SAE) proposed a six-level classification criteria for autonomous vehicles [68]. L1 AVs can conduct adaptive cruise control (ACC), which is fulfilled by the long-range radar. From the perspective of traffic sensing, the L1 AV can always detect the location and speed of its preceding vehicle in the same lane. From L2 to L5, AVs gradually take control from human drivers. To achieve that, AVs need to continuously monitor the surrounding traffic conditions. From the perspective of traffic sensing, L2–L5 vehicles can detect or track the vehicles in their surrounding areas. Here, we emphasize the difference between vehicle detection and vehicle tracking. Detection refers to the localization of a vehicle when it appears in the detection area of an AV, and tracking means that AV can keep track of a vehicle, as long as it is within the detection area. To be precise, the task of detection does not require one to “memorize” the detected vehicles in each time frame, while tracking requires the AV to keep track of the detected vehicles when they are within the detection range. Tracking is technically much more challenging than the detection. As of today, the detection technology is fairly mature, while the tracking technology is still not ready for real-world applications [69]. The reason for the difference is that the detection/tracking is conducted frame by frame on AVs. If the AV processes 30 frame per-second, tracking requires one to accurately detect all the vehicles in each frame and match them correspondingly, while detection does not require one to match the vehicles in different frames. The matching is challenging because vehicles often block each other at times, and this makes it difficult for machines to decide whether the detected vehicle is the same vehicle detected in previous frames. From the perspective of traffic sensing, detection only provides the locations of each vehicle, but tracking can provide additional speed information.

Analogous to the SAE’s automation level definitions, we define three levels of sensing power for AVs, as presented in Figure 2.

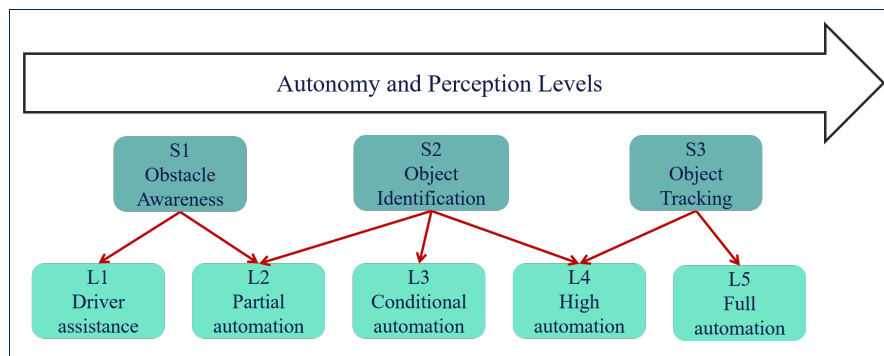


Figure 2. Overview of the perception levels.

The precise descriptions of the three perception levels are as follows.

- S_1 : The primary task for S_1 is to track the preceding vehicle within the same lane, and this is originally used for the ACC. However, the speed and location of the preceding vehicle are obtained for TSE.
- S_2 : In addition to S_1 , the primary task for S_2 is to detect and locate surrounding vehicles of an AV. Only vehicle counting at each time frame is obtained, and the speed information is not required in S_2 .
- S_3 : In addition to S_2 , the primary task for S_3 is to track every uniquely identified vehicle in the detection area, hence the location and speed of each vehicle are monitored over time by AVs in S_3 .

Based on the definition of AV perception levels, S_1 requires a LRR dedicated for preceding vehicles, S_2 requires a LiDAR/radar system, and S_3 requires a comprehensive fusion of camera, LiDAR, and radar. To be precise, Section 2.3 discusses how different sensors are combined to fulfill different levels of sensing power.

2.3. Detection Area of AVs

We now define the surrounding area (or detection area) of AVs, which is used throughout the paper. The detection area of AVs depends on the sensor configurations. Figure ?? presents two configurations of AV sensors. In the model of nuScenes, various sensors are mounted at different locations of an AV, while Waymo integrates most of the sensors on top of the vehicle. Depending on various sensor configurations on different AVs, the detection area of AVs can be different [70].

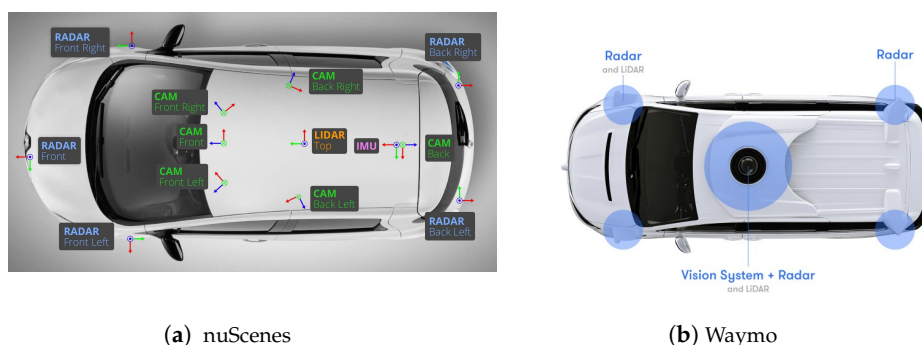


Figure 3. Two examples of sensor configurations, from nuScenes [71], Waymo Team [72].

In this paper, we adopt a simplified representation of the AV detection area, as presented in Figure 4.

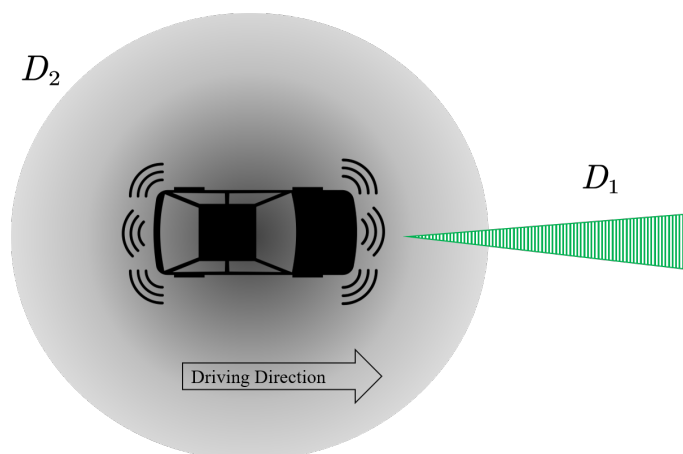


Figure 4. A simplified representation of AV detection area.

The detection area in Figure 4 consists of two components: D_1 and D_2 . D_1 is used for detecting the preceding vehicle and fulfilled by the LRR; D_2 is for detecting all the surrounding vehicles, which is supported by the combination of LiDAR and cameras. We assume only D_1 is active in S_1 , while both D_1 and D_2 are active in S_2 and S_3 , as presented in Table 3. Within the detection range, we assume that the AV can measure the distance between itself and surrounding vehicles with a zero mean distance error, and the impact of the error will be quantified in the numerical experiments.

Table 3. Summary of detection area and level of perceptions.

Sensing Power	Detection Area	Information Obtained
S_1	D_1	Speed/location of the preceding vehicle
S_2	D_1 and D_2	Speed/location of the preceding vehicle, location of surround vehicles
S_3	D_1 and D_2	Speed/location of the preceding vehicle and surrounding vehicles

2.4. Centralized Data Communication, Collection, and Processing

In this paper, we assume that there is a centralized data service (data center) that receives all the information sent by AVs, as presented in Figure 5. Due to the bandwidth and latency restrictions, AVs do not send all the raw data to the data center. Instead, they only send the location and speed of the surrounding vehicles if applicable. The main task for the data center is to aggregate the information and remove the redundant information when the same vehicle is detected multiple times by different AVs in S_2 and S_3 . This task can be done by checking and matching the location of the detected vehicles. For example, the vehicle with a green rectangle in Figure 5 is detected by two AVs, hence two duplicate data points are sent to the data center and the data center is able to identify and clean these duplicate data points. The localization accuracy is usually within the size of a standard vehicle, hence the accuracy for matching and cleaning is high [73]. In the numerical experiments, we will conduct a sensitivity analysis to evaluate the impact of different matching accuracies.

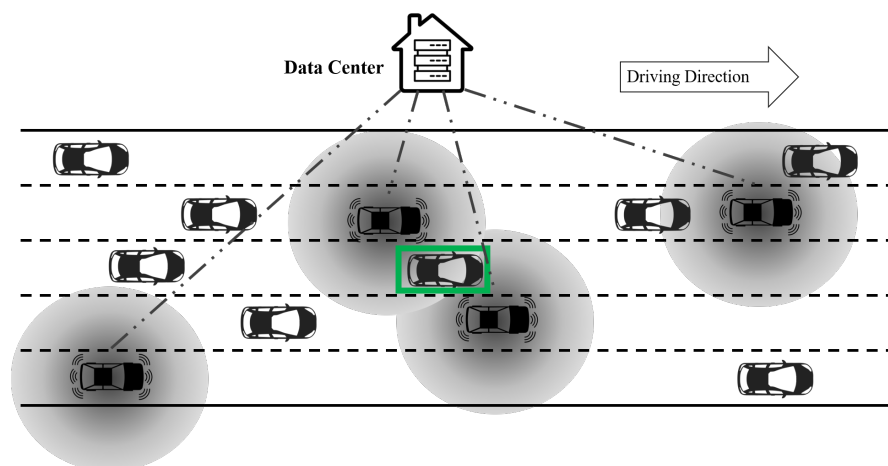


Figure 5. An illustration of the data center.

3. Formulation

Now, we are ready to rigorously formulate the traffic state estimation (TSE) framework with AVs. We first present the notations, and then the traffic states variables are defined. A two-step estimation method is proposed: the first step directly translates the information observed by AVs to spatio-temporal traffic states, and the second step employs data-driven methods to estimate the traffic states that are not observed by AVs.

3.1. Notations

All the notations will be introduced in context, and Table 4 provides a summary of the commonly used notations for reference.

3.2. Modeling Traffic States in Time-Space Region

We consider a highway with $|L|$ lanes, where $L = \{0, 1, \dots, |L| - 1\}$. The operator $|\cdot|$ is the counting measure for countable sets. For each lane $l \in L$, we denote X_l as the set of longitudinal locations on lane l . In this paper, we treat each lane as a one-dimensional line. Without loss of generality, we set the starting point of X_l to be 0, hence $X_l = [0, \mu(X_l)]$, where $\mu(X_l)$ is the length of lane l . Throughout the paper, we denote operator $\mu(\cdot)$ as the Lebesgue measure in either one or two dimensional Euclidean space, and it represents the length or area for one or two dimensional space. Note in this paper we assume the length of each lane is the same $\mu(X) = \mu(X_l), \forall l \in L$, while the proposed estimation method can be easily extended to accommodate different lane lengths. We further discretize the road X_l to $|S|$ equal distance road segments and each road segment is denoted by X_{ls} , where $s \in S$ is the index of the road segment and $S = \{0, 1, \dots, |S| - 1\}$. Hence, we have $X_{ls} = [s\Delta_X, (s + 1)\Delta_X]$ and $\Delta_X = \frac{\mu(X)}{|S|}$. The above road discretization is visualized in Figure 6.

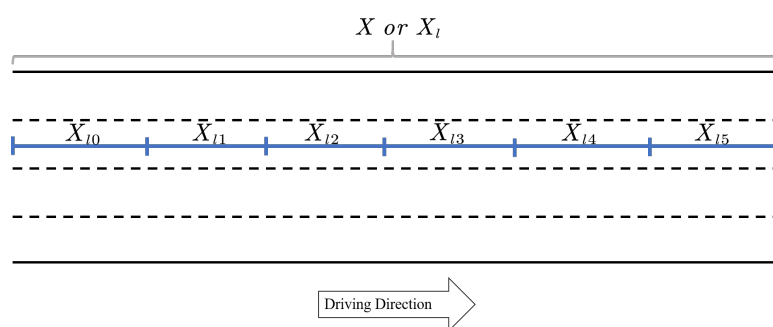


Figure 6. An illustration of the highway discretization.

Table 4. List of notations.

General Variables	
l	Index of a lane
L	The set of all lane indices l
T	The set of all time points in the study period
T_h	The set of all time points in time interval h
x	A longitudinal location along the road
X_l	The set of all longitudinal locations on lane l
$*$	Traffic state that is not directly observed by AVs
$\mu(\cdot)$	The Lebesgue measure for either one or two dimensional Euclidean space
$ \cdot $	The counting measure for the countable sets
Variables in a Time-Space Region	
h	Index of a time interval
H	The set of all indices t in the study period
s	Index of a longitudinal road segment
S	The set of all indices s
X_{ls}	The set of all longitudinal locations in road segment s and lane l
$A_l(h, s)$	A cell in time-space region for time interval h road segment s and lane l
$\bar{v}_l(h, s)$	Average speed for time interval h road segment s and lane l
$\bar{q}_l(h, s)$	Average traffic flow for time interval h road segment s and lane l
$\bar{k}_l(h, s)$	Average density for time interval h road segment s and lane l
$\alpha_l^i(h, s)$	The headway area of vehicle i in time-space region $A_l(h, s)$
Variables Related to Vehicles	
i	Index of a vehicle
I	The set of all vehicle indices i
$I_l(h, s)$	The set of all vehicles indices in time interval h road segment s and lane l
$v^i(t)$	Instantaneous speed of vehicle i at time t
$h^i(t)$	Instantaneous headway of vehicle i at time t
$x^i(t)$	Instantaneous longitudinal location of vehicle i at time t
$l^i(t)$	The lane in which vehicle i is located at time t
\underline{t}^i	The time point when the vehicle i enters the road
\bar{t}^i	The time point when the vehicle i exits the roads
d_{ls}^i	The distance traveled by vehicle i in road segment s on lane l
t_{lh}^i	The time spent in time interval h on lane l by vehicle i
Variables Related to Autonomous Vehicles	
j	Index of detection area
D_j	The detection area of an AV
I^A	The set of all autonomous vehicle indices
O_l^j	The set of time-space indices (h, s) such that X_{ls} is covered by the AV detection area D_j in time interval h
Variables Related to the Sensing Framework	
$\hat{k}_l(h, s)$	The directly observed density for time interval h road segment s and lane l
$\hat{v}_l(h, s)$	The directly observed speed for time interval h road segment s and lane l
$\hat{k}_l(h, s)$	The estimated density and speed for time interval h road segment s and lane l
$\hat{v}_l(h, s)$	The estimated speed for time interval h road segment s and lane l

We denote i as the index of a vehicle and I as the set of all vehicle indices. We further define the location $(x^i(t), l^i(t))$, speed $v^i(t)$, and space headway $h^i(t)$ at a time point $t \in T$, where $x^i(t)$ is the longitudinal location of vehicle i at time t , $l^i(t)$ is the lane in which vehicle i is located at time t , and T is the set of all time points in the study period. We assume that each vehicle i only enters the highway once. If a vehicle enters the highway multiple times, the vehicle at each entrance will be treated as a different vehicle. To obtain the traffic states, we construct the distance d_{ls}^i and time t_{lh}^i and headway area α_l^i from vehicle location $(x^i(t), l^i(t))$, speed $v^i(t)$, and headway $h^i(t)$ for a vehicle i based on Edie [74]. Throughout the paper, each vehicle is represented by a point, which is located at the center of each vehicle shape. When the vehicle is at the border of two lanes, function $l^i(t)$ will randomly assign the vehicle to either of the lanes.

Suppose \underline{t}^i denotes the time point when the vehicle enters the highway and \bar{t}^i denotes the time point when the vehicle exits the highway, we denote the distance traveled and time spent on lane l by vehicle i as d_{ls}^i and t_{lh}^i , respectively. Mathematically, d_{ls}^i and t_{lh}^i are presented in Equation (1).

$$\begin{aligned} d_{l_s}^i &= \mu(\{x^i(t) \in X_{l_s} | \underline{t}^i \leq t \leq \bar{t}^i, l^i(t) = l\}) \\ t_{l_h}^i &= \mu(\{t \in T_h | \underline{t}^i \leq t \leq \bar{t}^i, l^i(t) = l\}) \end{aligned} \quad (1)$$

We use the headway area α_l^i to represent the headway between vehicle i and its preceding vehicle on lane l in the time-space region, and it is represented by Equation (2).

$$\alpha_l^i = \{(t, x) | \underline{t}^i \leq t \leq \bar{t}^i, l^i(t) = l, x^i(t) \leq x \leq x^i(t) + h^i(t)\} \quad (2)$$

When we have the trajectories of all vehicles on the road, we can model the traffic states of each lane in a time-space region. Without loss of generality, we set the starting point of t to be zero, hence we have $T = [0, \mu(T)]$, where $\mu(T)$ is the length of the study period. We discretize the study period T to $|H|$ equal time intervals, where $H = \{0, 1, \dots, |H| - 1\}$. We denote T_h as the set of time points for interval h , where $h \in H$. Therefore, we have $T_h = [h\Delta_H, (h+1)\Delta_H]$, where $\Delta_H = \frac{\mu(T)}{|H|}$. In this paper, we use uniform discretization for X_l and T to simplify the formulation, while the proposed estimation methods work for the arbitrary discretization scheme.

We use $A_l(h, s)$ to denote a cell in the time-space region for road segment X_{l_s} and time period T_h , as presented in Equation (3).

$$\begin{aligned} A_l(h, s) &= T_h \otimes X_{l_s} \\ &= \text{Polygon}[(h\Delta_H, s\Delta_X), ((h+1)\Delta_H, s\Delta_X), \\ &\quad ((h+1)\Delta_H, (s+1)\Delta_X), (h\Delta_H, (s+1)\Delta_X)] \\ &= \{(t, x) | h\Delta_H \leq t \leq (h+1)\Delta_H, s\Delta_X \leq x \leq (s+1)\Delta_X\} \end{aligned}$$

We denote the headway area of vehicle i in cell $A_l(h, s)$ by $a_l^i(h, s)$, as presented in Equation (3). The headway area can be thought of as the multiplication of the space headway and time headway in the time-space region.

$$a_l^i(h, s) = A_l(h, s) \cap \alpha_l^i \quad (3)$$

Example 1 (Variable representation in time-space region). *In this example, we illustrate the variables defined in the time-space region. We consider a one-lane road and the lane index is 1. Furthermore, X_l is segmented into 6 road segments (X_{l_0}, \dots, X_{l_5}), and T is segmented into 10 time intervals (T_0, \dots, T_9), as presented in Figure 7. The cell $A_l(0, 4)$ is the intersection of X_{l_4} and T_0 , $A_l(8, 1)$ is the intersection of X_{l_1} and T_8 .*

Each green line in the time-space region represents the trajectory of a vehicle. In Figure 7, we highlight the first ($i = 0$), second ($i = 1$) and the 8th ($i = 7$) vehicle trajectory. The distance traveled by each vehicle is the same, hence $d_l^i = \mu(X_l) = \mu(X)$. We also highlight t_l^7 in Figure 7, which represents the time spent by each vehicle $i = 7$ on lane 1.

The headway area of vehicle $i = 1$, denoted by α_l^1 , is represented by the green shaded area. The red shaded area, which represents $a_l^1(1, 2)$, is the intersection of α_l^1 and $A_l(1, 2)$, based on Equation (3).

According to Seo et al. [46], Edie [74], we compute the traffic states variables, namely flow $\bar{q}_l(h, s)$, density $\bar{k}_l(h, s)$ and speed $\bar{v}_l(h, s)$, for each road segment X_{l_s} and time period T_h , as presented in Equation (4).

$$\begin{aligned} \bar{q}_l(h, s) &= \frac{\sum_{i \in I} d_{l_s}^i}{\sum_{i \in I} \mu(a_l^i(h, s))} \\ \bar{k}_l(h, s) &= \frac{\sum_{i \in I} t_{l_h}^i}{\sum_{i \in I} \mu(a_l^i(h, s))} \\ \bar{v}_l(h, s) &= \frac{\bar{q}_l(h, s)}{\bar{k}_l(h, s)} = \frac{\sum_{i \in I} d_{l_s}^i}{\sum_{i \in I} t_{l_h}^i} \end{aligned} \quad (4)$$

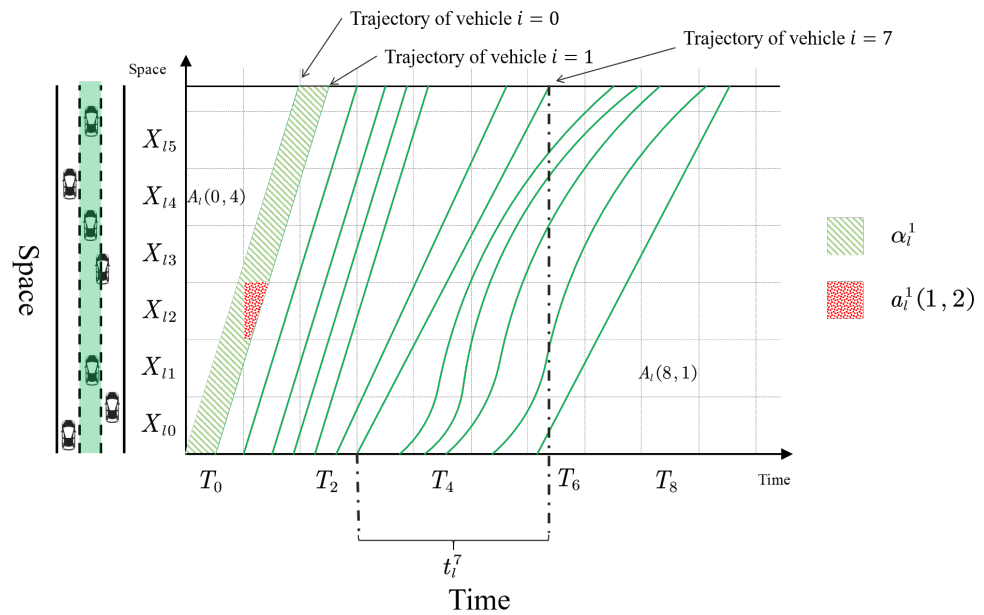


Figure 7. An example of variables in time-space region (the time-space region is associated with the green-colored lane, vehicles on the left subplot are for illustration purpose and do not exactly match with the trajectories on the left subplot).

We treat the traffic states, (e.g., flow $\bar{q}_l(h, s)$, density $\bar{k}_l(h, s)$ and speed $\bar{v}_l(h, s)$) estimated from full samples of vehicles I as ground truth and unknown. In the following sections, we will develop a data-driven framework to estimate the traffic states from the partially observed traffic information obtained from autonomous vehicles under different levels of perception power.

3.3. Challenges in the High-Resolution TSE with AVs

As summarized in Table 1, TSE with AVs is a unique problem that has not been explored. In this section, we highlight the difficulties of this unique problem, and further motivate the proposed traffic sensing framework in the following sections.

As the vehicle trajectories in high-resolution time-space region are complicated, the information collected by the AVs is also highly complicated and fragmented. To illustrate, we consider a two-lane road with three vehicles (one AV, and two conventional vehicles A and B), as shown in Figure 8. The trajectory of the AV is represented by the blue line, and the shaded area is the detection area of the AV. The detection area of the AV is represented by the shaded areas with blue and grey color. Even when the AV is on the lane 0, it can still detect the vehicles on lane 1, thanks to the characteristics of LiDAR and cameras. Hence, the blue area means that the AV is on the current lane, and the grey area means that the AV is on the other lane.

As shown in Figure 8, both the AV and vehicle A change lanes during the trip. Vehicle A changes lane at the middle of time interval 1, and the AV change lanes at the beginning of time interval 3. One can see that vehicle A can be detected by the AV in the cell (1, 2) on both lane 0 and lane 1, and vehicle B can be detected in the cell (3, 3) and (3, 4) on lane 1. For the cell (1, 3) on lane 1, it is not straightforward to see whether vehicle A is detected or not, hence rigorous mathematical formulations should be developed to determine which cells are observable and which cells should be estimated. In real-world congested roads, vehicles might change lanes frequently, and hence their trajectories can be very complicated. From the example, we can see that, due to the complicated vehicle trajectories on the roads, the information collected by the AVs is non-uniformly distributed in the time-space region, and most of the existing TSE methods cannot be applied to such data. Therefore, solving the TSE with AV data calls for a new estimation framework.

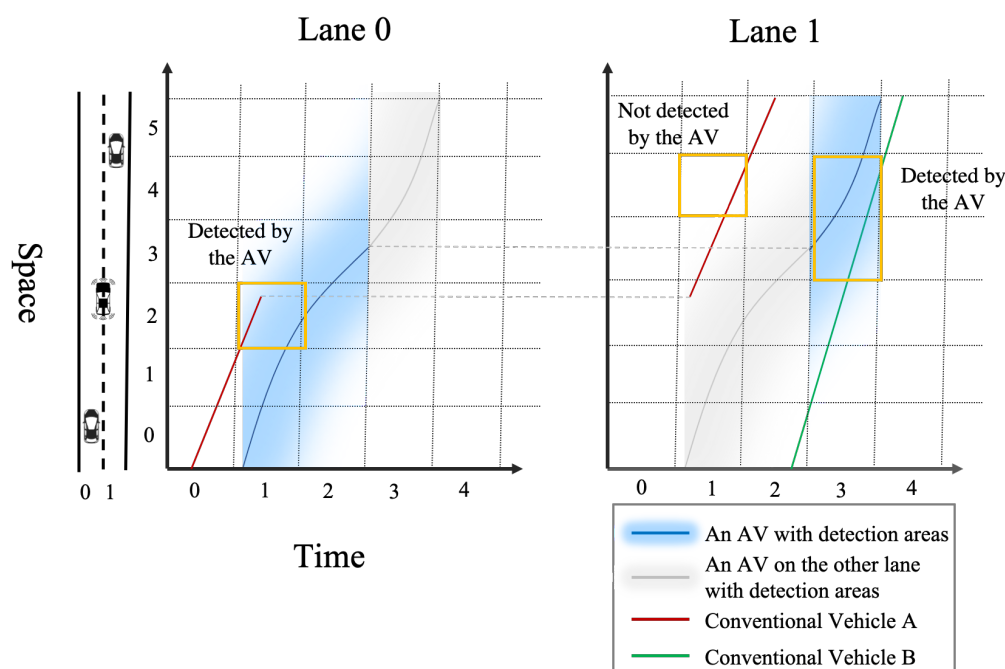


Figure 8. An illustrative example of the complicated and fragmented information collected by AVs (colored version available online).

3.4. Overview of the Traffic Sensing Framework

In this section, we present an overview of the traffic sensing framework with AVs. We assume a subset of vehicles are AVs, namely $I^A \subseteq I$, where I^A denotes the index set of all AVs. The goal for the traffic sensing framework is to estimate the density and speed of each cell in the time-space region, using the information observed by AVs. Once the speed and density are estimated accurately, the traffic flow can be obtained by the conservation law [75].

The TSE methods can be categorized into two types: model-driven methods and data-driven methods. The model-driven methods rely on physical models such as traffic flow theory, while the data-driven methods automatically learn the relationship between different variables. In the case of AV-based TSE, the observed information is fragmented and lacking in certain patterns, so it is challenging to establish the physical model. In contrast, the data-driven methods can be built easily, thanks to the massive data collected by the AVs. Therefore, this paper focuses on the data-driven approach.

The proposed framework consists of two major parts: direct observation and data-driven estimation, as presented in Figure 9.

In the direct observation step, density and speed are observed directly through AVs. Since AVs are moving observers [37], traffic states can only be observed partially for a set of time intervals and road segments (i.e. cells) in a time-space region. Section 3.5 will rigorously determine the set of cells that can be directly observed by AVs and compute the direct observations from information obtained by AVs. We will discuss the direct observation with different levels of sensing power. The second part aims at filling up the unobserved information with data-driven estimation methods. The functions Ψ_l and Φ_l are used to estimate the unobserved density and speed on lane l , respectively. Details will be presented in Section 3.6.

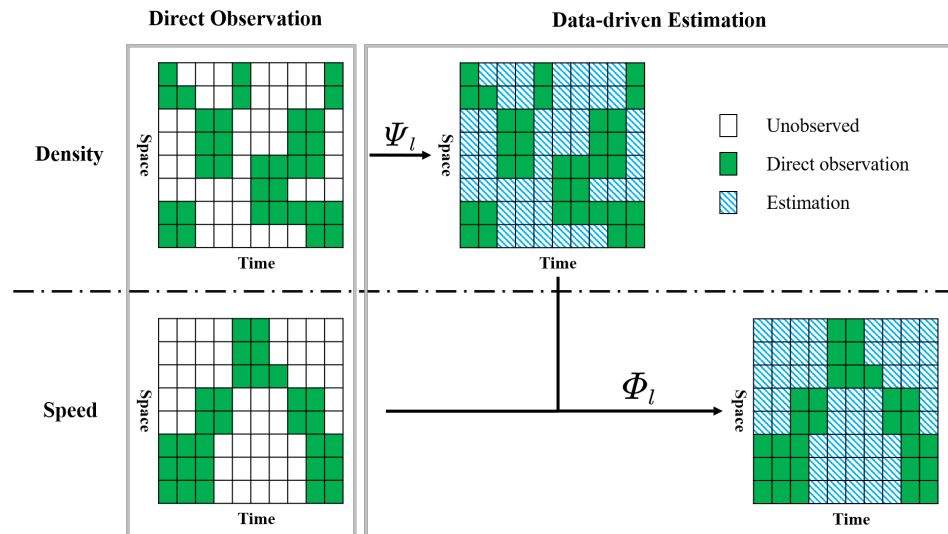


Figure 9. An overview of the traffic sensing framework for each one lane.

3.5. Direct Observation

In this section, we present to compute traffic states using the information that is directly observed by AVs under different levels of perception. We define $O_l^j, j \in \{1, 2\}$ as the set of time-space indices (h, s) of the cells that are observed by the AVs, and j is the index of the detection area. The set O_l^1 presents all (h, s) detected by D_1 , and O_l^2 present all (h, s) detected by D_2 . Details are presented in Appendix A. Now we formulate the traffic states that can be directly observed by AVs under different levels of perception. As a notation convention, we use $*$ to represent the information that cannot be directly observed by AVs, and \tilde{k}_l, \tilde{v}_l denote the directly observed density and speed, respectively.

3.5.1. S_1 : Tracking the Preceding Vehicle

In the perception level S_1 , an AV can only detect and track its preceding vehicle, and hence its detection area for density and speed is O_l^1 . The observed density and speed can be represented in Equation (5).

$$\tilde{k}_l(h, s) = \begin{cases} \frac{\sum_{i \in I^A} t_{ih}^i}{\sum_{i \in I^A} \mu(a_i^i(h, s))} & \text{if } (h, s) \in O_l^1 \\ * & \text{else} \end{cases} \quad (5)$$

$$\tilde{v}_l(h, s) = \begin{cases} \frac{\sum_{i \in I^A} d_{is}^i}{\sum_{i \in I^A} t_{ih}^i} & \text{if } (h, s) \in O_l^1 \\ * & \text{else} \end{cases}$$

Equation (5) is proven to be an accurate estimation of the traffic states [46]. We note that some AVs also have a LRR mounted to track the following vehicle behind the AVs, and this situation can be accommodated by replacing the set I^A with $I^A \cup \text{Following}(I^A)$ in Equation (5), where $\text{Following}(I^A)$ represents all the vehicles that follow AVs in I^A .

When the AV market penetration rate is low, O_l^1 only covers a small fraction of all cells in the time-space region, especially for multi-lane highways. In contrast, O_l^2 covers more cells than O_l^1 . Practically, it implies that the LiDAR and cameras are the major sensors for traffic sensing with AVs.

3.5.2. S_2 : Locating Surrounding Vehicles

In the perception level S_2 , both D_1 and D_2 are enabled by the LRR, LiDAR and cameras, while D_2 can only detect the location of surrounding vehicles. Hence, the density can be observed in both D_1 and D_2 , and the speed is only observed in D_1 . The estimation

method for D_1 cannot be used for D_2 , since the preceding vehicles of the detected vehicle might not be detected, hence $\alpha_l^i(h, s)$ cannot be estimated accurately. Instead, D_2 provides a snapshot of the traffic density at a certain time point, and we can compute the density of time interval h by taking the average of all snapshots, as presented in Equation (6).

$$\tilde{k}_l(h, s) = \begin{cases} \frac{\sum_{i \in A} t_{1h}^i}{\sum_{i \in A} \mu(a_l^i(h, s))} & \text{if } (h, s) \in O_l^1 \\ \frac{1}{\mu(T_h^o(l, s))} \int_{t \in T_h^o(l, s)} \frac{|I_l^o(t, s)|}{\Delta_X} dt & \text{else if } (h, s) \in O_l^2 \\ \star & \text{else} \end{cases} \quad (6)$$

$$\tilde{v}_l(h, s) = \begin{cases} \frac{\sum_{i \in A} d_{1s}^i}{\sum_{i \in A} t_{1h}^i} & \text{if } (h, s) \in O_l^1 \\ \star & \text{else} \end{cases}$$

where $T_h^o(l, s)$ represents the set of time indices when X_{ls} is covered by the D_2 in T_h , and $I_l^o(t, s)$ represents the set of vehicles detected by D_2 on X_{ls} at time t . Detailed formulations are presented in Appendix A.

3.5.3. S_3 : Tracking Surrounding Vehicles

In the perception level S_3 , both localization and tracking are enabled by the LRR, LiDAR and cameras. In addition to the information obtained by S_2 , speed information of surrounding vehicles in D_2 is also available. Similar to the density estimation, we first computed the instantaneous speed of a cell at a certain time point by taking the harmonic mean of all detected vehicles, and then the average speed of a cell is computed by taking the average of all time points, as presented by Equation (7).

$$\tilde{k}_l(h, s) = \begin{cases} \frac{\sum_{i \in A} t_{1h}^i}{\sum_{i \in A} \mu(a_l^i(h, s))} & \text{if } (h, s) \in O_l^1 \\ \frac{1}{\mu(T_h^o(l, s))} \int_{t \in T_h^o(l, s)} \frac{|I_l^o(t, s)|}{\Delta_X} dt & \text{else if } (h, s) \in O_l^2 \\ \star & \text{else} \end{cases} \quad (7)$$

$$\tilde{v}_l(h, s) = \begin{cases} \frac{\sum_{i \in A} d_{1s}^i}{\sum_{i \in A} t_{1h}^i} & \text{if } (h, s) \in O_l^1 \\ \frac{1}{\mu(T_h^o(l, s))} \int_{t \in T_h^o(l, s)} \text{hmean}(\{v^i(t) | x^i(t) \in X_{ls}, i \in I\}) dt & \text{else if } (h, s) \in O_l^2 \\ \star & \text{else} \end{cases}$$

where $\text{hmean}(\cdot)$ represents the harmonic mean. Though S_3 provides the most speed information, the directly observed density is the same for S_2 and S_3 . Overall, the sensing power of AV increases as more cells are directly observed from S_1 to S_3 . In the following section, we will present to fill the \star using data-driven methods.

3.6. Data-Driven Estimation Method

In this section, we propose a data-driven framework to estimate the unobserved density and speed in $\tilde{k}_l(h, s), \tilde{v}_l(h, s)$. To differentiate the density (speed) before and after the estimation, we use $\hat{k}_l(h, s)$ and $\hat{v}_l(h, s)$ to represent the estimated density and speed for time interval h road segment s and lane l , while $\tilde{k}_l(h, s), \tilde{v}_l(h, s)$ denote the density and speed before the estimation (i.e. after the direct observation). The method consists of two steps: (1) estimate the unobserved density $\hat{k}_l(h, s)$ given the observed density $\tilde{k}_l(h, s)$; (2) estimate the unobserved speed $\hat{v}_l(h, s)$ given that the density $\hat{k}_l(h, s)$ is fully known from estimation and speed $\tilde{v}_l(h, s)$ is partially known from direct observation.

We present the generalized form for estimating the unobserved density and speed in Equations (8) and (9), respectively.

$$\hat{k}_l(h, s) = \Psi_l(h, s, \{\tilde{k}_{l'}\}_{l'}) \quad (8)$$

$$\hat{v}_l(h, s) = \Phi_l(h, s, \{\tilde{v}_{l'}\}_{l'}, \{\hat{k}_{l'}\}_{l'}) \quad (9)$$

where Ψ_l is a generalized function that takes the observed density $\{\tilde{k}_l\}_l$ and time/space index h, s as input and outputs the estimated density. Φ_l is also a generalized function to estimate speed, while its inputs include the observed speed $\{\tilde{v}_l\}_l$, the estimated density $\{\hat{k}_l\}_l$, and the time/space index h, s . In this paper, we propose matrix completion-based methods for Ψ_l , and both matrix completion-based and regression-based methods for Φ_l . The details are presented in the following subsections.

3.6.1. Matrix Completion-Based Methods

The matrix completion-based model can be used to estimate either density or speed. We first assume that densities (speeds) in certain cells are directly observed by the AVs, as presented in Equation (10).

$$\begin{aligned}\hat{k}_l(h, s) &= \tilde{k}_l(h, s), \forall (h, s) \in O_l^k \\ \hat{v}_l(h, s) &= \tilde{v}_l(h, s), \forall (h, s) \in O_l^v\end{aligned}\quad (10)$$

where we denote O_l^k and O_l^v as the detection area for density and speed of lane l in time-space region with a certain sensing power. Precisely, for S_1 , $O_l^k = O_l^1, O_l^v = O_l^1$; for S_2 , $O_l^k = O_l^1 \cup O_l^2, O_l^v = O_l^1$; and for S_3 , $O_l^k = O_l^1 \cup O_l^2, O_l^v = O_l^1 \cup O_l^2$.

For each lane l , the estimated density \hat{k}_l (or speed \hat{v}_l) forms a matrix in the time-space region, and each row represents a road segment s , and each column represents a time interval h . Some entries ($(h, s) \notin O_l^k$ or $(h, s) \notin O_l^v$) in the density matrix (or speed matrix) are missing. To fill the missing entries, many standard matrix completion methods can be used. For example, the naive imputation (imputing with the average values across all time intervals or across all cells), k-nearest neighbor (k-NN) imputation [76], and the singular-value decomposition (SVD)-based SOFTIMPUTE algorithm [77]. In the numerical experiments, the above-mentioned methods will be benchmarked using real-world data.

3.6.2. Regression-Based Methods

The speed data can also be estimated by a regression-based model, given that the density $\hat{k}_l(h, s)$ is fully estimated. We train a regression model f_l to estimate the speed from densities for lane l , as presented in Equation (11).

$$\begin{aligned}\hat{v}_l(h, s) = f_l(\{\hat{k}_{l'}(h', s') : l - \delta_l \leq l' \leq l + \delta_l, h - \delta_h \leq h' \leq h, \\ s - \delta_s \leq s' \leq s + \delta_s, l' \in L, s \in S, h \in H\})\end{aligned}\quad (11)$$

where $\delta_l, \delta_h, \delta_s$ represent the number of nearby lanes, time intervals and road segments considered in the regression model. The intuition behind the regression model is that the speed of a cell can be inferred by the densities of its neighboring cells. The choice of Equation (11) is inspired by the traffic flow theory (e.g., fundamental diagrams and car-following models) as the interactions between vehicles result in the road volume/speed. A specific example of f_l is the fundamental diagram [78], which is formulated as $\hat{v}_l(h, s) = f_l(\hat{k}_l(h, s))$ by setting $\delta_l = \delta_h = \delta_s = 0$.

In this paper, we adopt a simplified function $f_l(\cdot)$ presented in Figure 10. Suppose we want to estimate the speed for cell 1; there are 12 neighboring cells (including cell 1) considered as inputs. The regression methods adopted in this paper are Lasso [79] and random forests [80]. We also map the cells to the physical roads in time t and $t - 1$, as presented in Figure 11. Figure 11 shows a three-lane road in time t and $t - 1$, and the numbers marked for each road segment exactly match the cell number in Figure 10.

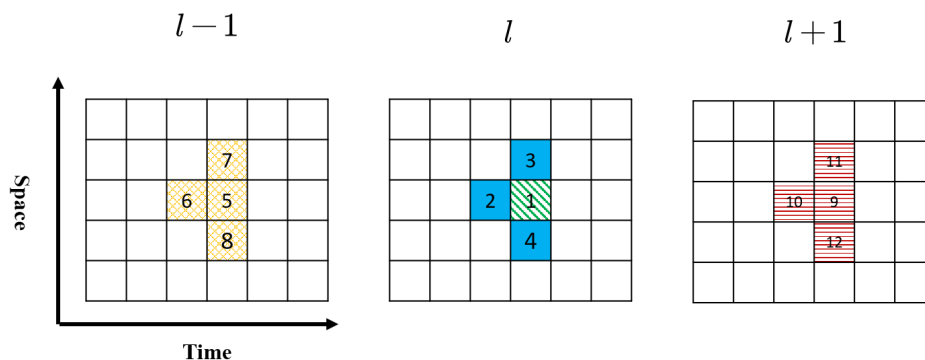


Figure 10. Cells in time-space region used for speed estimation.

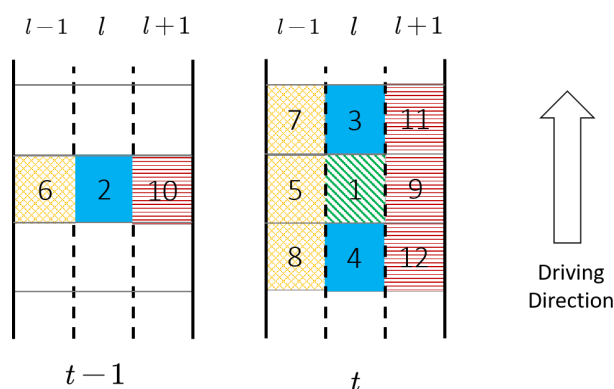


Figure 11. Cells in physical road used for speed estimation.

4. Solution Algorithms

In this section, we discuss some practical issues regarding the traffic sensing framework proposed in Section 3.

4.1. Computation of $a_i^j(h, s)$

To obtain the ground truth (Equation (4)) and the observed density (Equation (5)), $a_i^j(h, s)$, which denotes the headway area of vehicle i in cell $A_l(h, s)$ (Equation (3)), needs to be computed in the time-space region. Moreover, $a_i^j(h, s)$ is computed by intersecting $A_l(h, s)$ and α_i^j , and $A_l(h, s)$ can be represented by a rectangle in the time-space region. The headway area for vehicle i α_i^j is usually banded [46], which can be approximated by a polygon. Therefore, $a_i^j(h, s)$ can also be represented by a polygon, and the interaction of a rectangle (which is also a special polygon) and a polygon can be conducted efficiently [81].

4.2. Sampling Rate

As discussed in Section 2.4, AVs send messages to the data center periodically. Let the sampling rate denote the message sending frequency, and we assume that all AVs have the same sampling rate. When the sampling rate is high, the data center can obtain the density and speed information in high temporal resolution, hence the traffic sensing can be accurate. On the other hand, the sampling rate is limited by the bandwidth and latency of the message transmission network. In the numerical experiments, sensitivity analysis will be conducted to study the impact of sampling rate.

4.3. Cross-Validation

In the data-driven method presented in Section 3.6, the cross-validation is conducted for model selection in both matrix completion-based and regression-based methods.

In the matrix completion-based model, we use cross-validation to select the maximal rank in the SOFTIMPUTE and the number of nearest neighbors in the k-NN imputation [82]. To perform the cross-validation for the matrix completion, we randomly hide 10% of the matrix entries and run the imputation methods on the rest of entries. Then, we measure the imputation accuracy by comparing the imputed values and the actual values on the 10% hidden entries.

In the regression-based model, 5-fold cross-validation is performed to select the optimal parameter settings for different regression methods, such as the weight of regularization term in Lasso, number of base estimators in random forests.

5. Numerical Experiments

In this section, we conduct the numerical experiments with NGSIM data to examine the proposed TSE framework. All the experiments below are conducted on a desktop with Intel Core i7-6700K CPU @ 4.00GHz \times 8, 2133 MHz 2 \times 16GB RAM, 500GB SSD, and the programming language is Python 3.6.8.

5.1. Data and Experiment Setups

We use next generation simulation (NGSIM) data to validate the proposed framework. NGSIM data contain high-resolution vehicle trajectory data on different roads [83]. Our experiments are conducted on I-80, US-101 and Lankershim Boulevard, and the overviews of the three roads are presented in Figure 12. NGSIM data are collected using a digital video camera, and its temporal resolution is 100ms. Details of the three roads can be found in Alexiadis et al. [83], He [84].

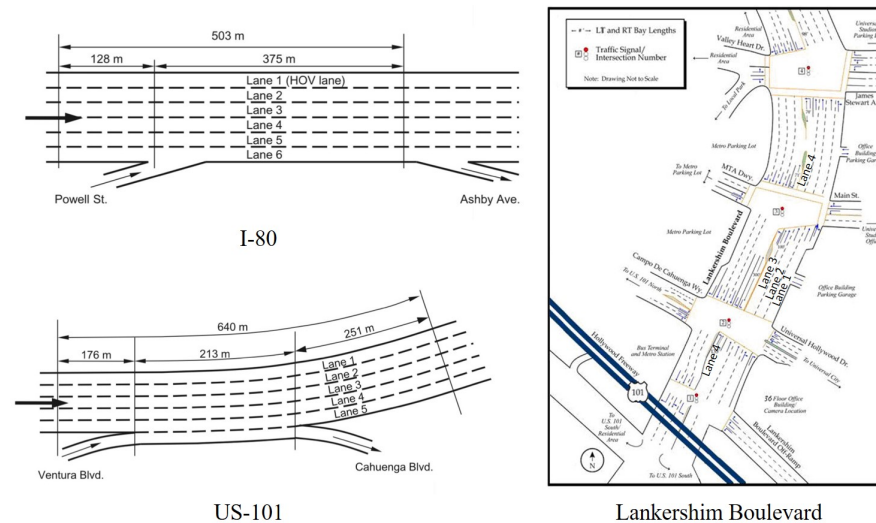


Figure 12. Overview of three networks (adapted from NGSIM website [83] and He [84], high-resolution figures are available at FHWA [85]).

We assume that a random set of vehicles are AVs and the AVs can perceive the surrounding traffic conditions. Given the limited information collected by AVs, we estimate the traffic states using the proposed framework. We further compare the estimation results with the ground truth computed from the full vehicle trajectory data. The normalized root mean squared error (NRMSE), symmetric mean absolute percentage error (SMAPE1, SMAPE2) will be used to examine the estimation accuracy, as presented by Equation (12). SMAPE2 is considered as a robust version of SMAPE1 [86]. All three measurements are unitless.

$$\begin{aligned}
\text{NRMSE}(z, \hat{z}) &= \sqrt{\frac{\sum_{v \in \mathcal{M}} (z_v - \hat{z}_v)^2}{\sum_{v \in \mathcal{M}} z_v^2}} \\
\text{SMAPE1}(z, \hat{z}) &= \frac{1}{|\mathcal{M}|} \sum_{v \in \mathcal{M}} \frac{|z_v - \hat{z}_v|}{z_v + \hat{z}_v} \\
\text{SMAPE2}(z, \hat{z}) &= \frac{\sum_{v \in \mathcal{M}} |z_v - \hat{z}_v|}{\sum_{v \in \mathcal{M}} (z_v + \hat{z}_v)}
\end{aligned} \tag{12}$$

where z is the true vector, \hat{z} is the estimated vector, v is the index of the vector, and \mathcal{M} is the set of indices in vector z and \hat{z} . When comparing two matrices, we flatten the matrices to vector and then conduct the comparison.

Here, we describe all the factors that affect the estimation results. The market penetration rate denotes the portion of AVs in the fleet. In the experiments, we assume that AVs are uniformly distributed in the fleet. The detection area D_1 is a ray fulfilled by LRR and D_2 is a circle fulfilled by LiDAR. We assume that LiDAR has a detection range (radius of the circle) and it might also oversee a vehicle with a certain probability (referred to as missing rate). The AVs can be at one level of perception, as discussed in Section 2. The sampling rate of data center can be different. In addition, different data-driven estimation methods are used to estimate the density and speed, as presented in Section 3.6. We define LR1 and LR2 as Lasso regressions, and RF1 and RF2 as random forests regressions. The number 1 means that only cells 1 to 4 are used as inputs, while the number 2 means that all 12 cells in Figure 10 are used as inputs. SI denotes the SOFTIMPUTE, KNN denotes the k-nearest neighbor imputation, and NI denotes the naive imputation by simply replacing missing entries with the mean of each column.

Baseline setting: the market penetration rate of AVs is 5%. The detection range of LRR is 150 m, and the detection range of LiDAR is 50 m with 5% missing rate. The level of perception is S_3 , and the speed is detected without any noise. The sampling rate of data center is 1 Hz. SI is used to estimate density and LR2 is used to estimate speed. We set $|H| = 90$ and $|S| = 60$.

5.2. Basic Results

We first run the proposed estimation method with the baseline setting. The estimation method takes around 7 minutes to estimate all three roads, and the most time consuming part is the information aggregation in the data center (discussed in Section 2.4) and the computation of Equation (7). The estimation accuracy is computed by averaging the NRMSE, SMAPE1 and SMAPE2 through all lanes, and the results are presented in Table 5. In addition to the unitless measures, we also include the mean absolute error (MAE) in the table.

Table 5. Estimation accuracy with basic setting (unit for NRMSE, SMAPE1, SMAPE2: %; unit for speed MAE: miles/hour; unit for density MAE: vehicles/miles).

Measures	Density				Speed			
	NRMSE	SMAPE1	SMAPE2	MAE	NRMSE	SMAPE1	SMAPE2	MAE
I-80	18.61	7.65	6.87	10.83	9.40	3.17	2.73	0.56
US-101	18.28	7.76	6.89	10.75	7.49	2.88	2.40	1.13
Lankershim	50.94	22.73	19.71	27.03	24.08	10.01	8.00	4.26

In general, the estimation method yields accurate estimation on highways (I-80 and US-101), while it underperforms on the complex arterial road (Lankershim Boulevard). Estimation accuracy of speed is always higher than that of density, which is because the density estimation requires every vehicle being sensed, while speed estimation only needs a small fraction of vehicles being sensed [87].

Estimation accuracy on different lanes. We then examine the performance of the proposed method on each lane separately, and the estimation accuracy of each lane is summarized in Table 6.

Table 6. Estimation accuracy on each lane with basic setting (unit: %).

Item	Lanes	I-80								
		1	2	3	4	5	6			
Density	NRMSE	32.08	15.24	18.08	15.46	15.51	15.30			
	SMAPE1	13.43	6.21	7.17	6.02	6.45	6.61			
	SMAPE2	12.40	5.57	6.39	5.35	5.71	5.75			
Speed	NRMSE	6.82	9.68	9.17	10.99	9.99	9.73			
	SMAPE1	1.93	3.71	3.17	3.77	3.26	3.18			
	SMAPE2	1.94	3.01	2.73	3.17	2.83	2.73			
Item	Lanes	US-101					Lankershim Blvd			
		1	2	3	4	5	1	2	3	4
Density	NRMSE	17.90	17.95	18.47	18.09	18.96	45.49	41.30	43.75	73.19
	SMAPE1	7.56	7.62	7.60	7.75	8.27	20.92	21.00	21.37	27.63
	SMAPE2	6.75	6.79	6.83	6.79	7.28	17.53	16.59	16.95	27.73
Speed	NRMSE	8.76	7.85	6.86	7.18	6.76	23.94	21.33	25.26	25.78
	SMAPE1	3.62	3.05	2.65	2.56	2.50	10.22	8.77	10.97	10.08
	SMAPE2	2.87	2.48	2.22	2.19	2.21	7.71	6.77	8.49	9.02

One can read from Table 6 that the proposed method performs similarly on most lanes. One interesting observation is that the proposed method performs well on Lane 6 on I-80, Lane 5 on US-101 and Lane 1 on Lankershim Blvd, and those lanes are merged with ramps. This implies that the proposed method has the potential to work well on merging intersections.

The proposed method performs differently on lanes that are near the edge of roads. For example, the proposed method yields the worst density estimation and the best speed estimation on lane 1 of I-80, which is an HOV lane. The vehicle headway is relatively large on the HOV lane, hence estimating density is more challenging given limited detection range of LiDAR. In contrast, speed on HOV lane is relatively stable, making the speed estimation easy. In addition, the estimation accuracy of the Lane 4 of Lankershim Blvd is low, as a result of the physical discontinuity of the lane.

One noteworthy point is that the estimation accuracy also depends on the traffic conditions. For example, traffic conditions on the HOV lane of I-80 tend to be free-flow and low-density, so the estimation accuracy is different from other lanes which tend to be dense and congested.

To visually inspect the estimation accuracy, we plot the true and estimated density and speed in time-space region for Lane 2 and Lane 4 of all three roads in Figures 13 and 14. It can be seen that the estimated density and speed resemble the ground truth; even the congestion is discontinuous in the time-space region (see Lankershim Blvd in Figure 13). Again, the Lane 4 of Lankershim Blvd is physically discontinuous, hence a large block of entries are entirely missing in time-space region (see the third row of Figure 14), and the blocked missingness may affect the proposed methods and increase the estimation errors.

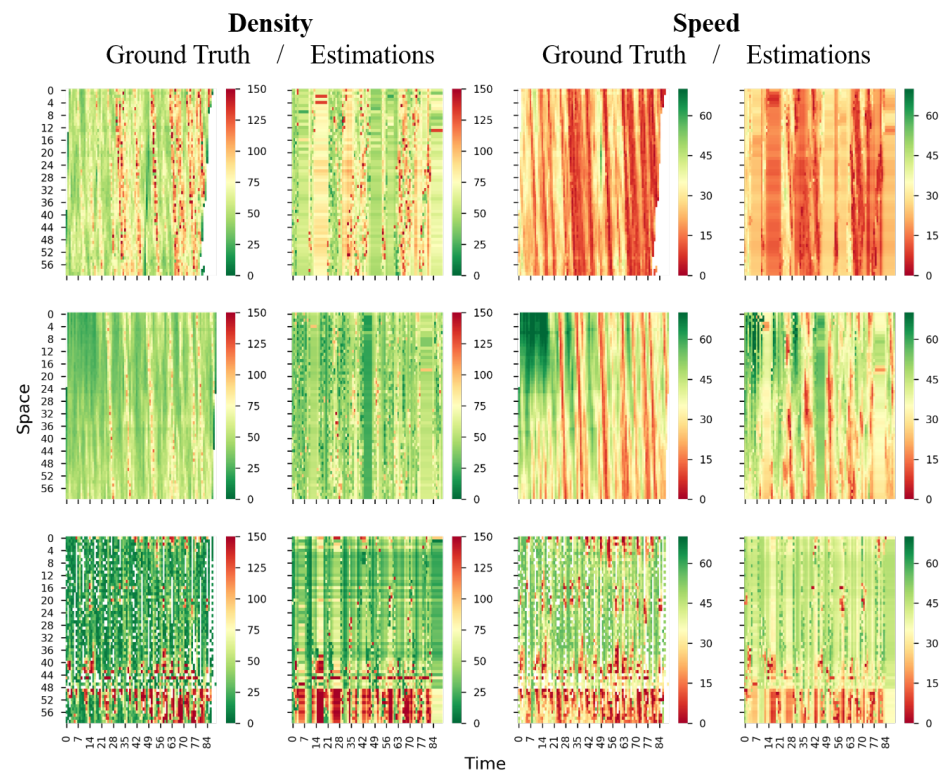


Figure 13. True and estimated density and speed for lane 2 (first row: I-80, second row: US-101, third row: Lankershim Blvd; first column: ground truth density, second column: estimated density, third column: ground truth speed, fourth column: estimated speed; density unit: veh/km, speed unit: km/h).

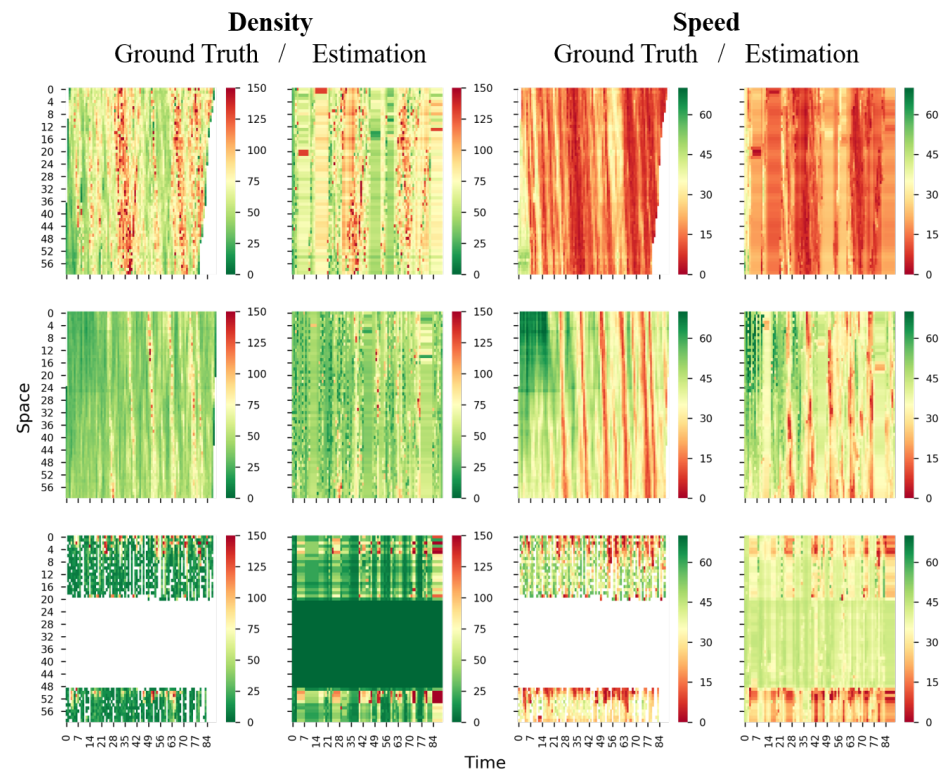


Figure 14. True and estimated density and speed for lane 4 (first row: I-80, second row: US-101, third row: Lankershim Blvd; first column: ground true density, second column: estimated density, third column: ground true speed, fourth column: estimated speed; density unit: veh/km, speed unit: km/h).

Effects of Densities on Speeds in the Regression-Based Models

We use the LR2 method to estimate the speed. After fitting the LR2 model, we look at the fitted regression coefficients and interpret the coefficients from the perspective of traffic flow theories. In particular, we select Lane 2 on US-101 and summarize the fitted coefficients in Table 7. The regression coefficients for other lanes and networks can be found in the supplementary material.

Table 7. Coefficients of Lasso regression for Lane 2 on US-101 (x1 to x12 correspond to the cell number in Figure 10 or Figure 11).

Variables	Coefficients	Standard Error	t-Statistic	p-Value	2.5% Quantile	97.5% Quantile
Intercept	0.0778	0.000	254.909	0.000	0.077	0.078
x1	−0.4738	0.023	−20.676	0.000	−0.519	−0.429
x2	−0.2630	0.018	−14.326	0.000	−0.299	−0.227
x3	−0.4146	0.023	−17.700	0.000	−0.461	−0.369
x4	−0.4781	0.024	−20.346	0.000	−0.524	−0.432
x5	−0.2043	0.023	−9.030	0.000	−0.249	−0.160
x6	−0.0947	0.017	−5.726	0.000	−0.127	−0.062
x7	−0.1620	0.022	−7.352	0.000	−0.205	−0.119
x8	−0.2354	0.022	−10.562	0.000	−0.279	−0.192
x9	−0.1727	0.025	−6.790	0.000	−0.223	−0.123
x10	−0.1788	0.019	−9.477	0.000	−0.216	−0.142
x11	−0.1553	0.025	−6.198	0.000	−0.204	−0.106
x12	−0.2067	0.025	−8.256	0.000	−0.256	−0.158

The R-square for Lane 2 on US-101 is 0.832, indicating that the regression model is fairly accurate. From Table 7, one can see that the Intercept is positive and it represents the free flow speed when the density is zero. Coefficients for x1 to x12 are all negative with high confidence, and this implies that higher density generally yields lower speed.

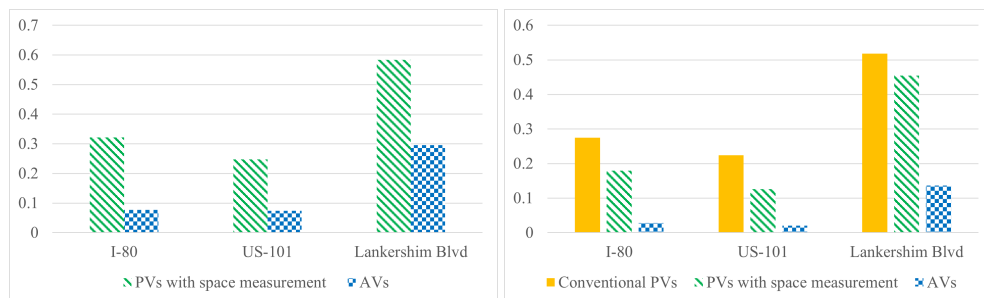
Recall Figure 10; suppose we want to estimate the speed for cell 1, we refer to cell 1~4 as the surrounding cells in the current lane and cell 5~12 as the surrounding cells in the nearby lanes. The coefficients of x1 to x4 are the most negative, indicating the densities of the surrounding cells in the current lane have the highest impact on the speed. The densities of surrounding cells in the nearby lanes also have negative impact on the speed but the magnitude is lower.

5.3. Comparing TSE Methods with Different Types of Probe Vehicles (PVs)

In this section, we compare the proposed method with other TSE methods using different types of PVs. Consistent with Table 1, we consider the following three types of PVs:

- **Conventional PVs:** Only speed can be estimated using the conventional PVs, and the estimation method is adopted from Yu et al. [61].
- **PVs with spacing measurement:** the TSE can be conducted and both speed and density can be estimated. The estimation method is adopted from Seo et al. [46].
- **AVs:** The TSE can be conducted and both speed and density can be estimated. The estimation method is proposed in this paper.

All three methods are implemented with baseline setting, and we make 5% of PVs conventional PVs, PVs with spacing measurement, or AVs. The estimation accuracy in terms of SMAPE1 is presented in Figure 15.



(a) Estimation accuracy for density. (b) Estimation accuracy for speed.

Figure 15. Average SMAPE1 for different types of PVs with baseline setting.

One can see that, by using more information collected by AVs, the proposed method outperforms the other TSE methods using different PVs for both speed and density estimation. This experiment further highlights the great potential of using AVs for TSE.

5.4. Comparing Different Algorithms

In this section, we examine different methods in estimating density and speed. Recall in Section 3.6 that the matrix completion-based methods can estimate both density and speed, while the regression-based methods can only estimate the speed. We run the proposed estimation method with different combinations of estimation methods for density and speed, and the rest of the settings are the same as the baseline setting. To be precise, three methods are used to estimate density: naive Imputation (NI), k-nearest neighbor imputation (KNN) and SOFTIMPUTE (SI). Seven methods will be used to estimate speed, and they are NI, KNN, SI, LR1, LR2, RF and RF2. We plot the heatmap of SMAPE1 for each road separately, as presented in Figure 16.

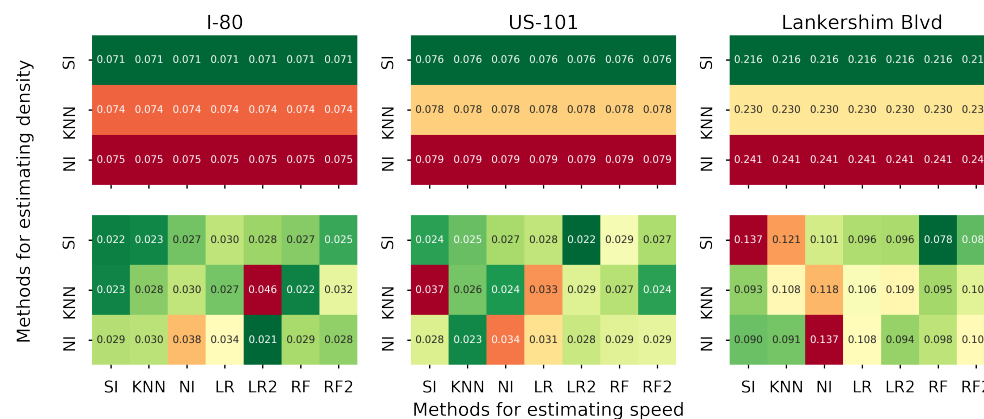


Figure 16. Average MSAPe1 for different estimation methods on each road (in terms of MSAPe1, first row: density, second row: speed).

The speed estimation does not affect the density estimation, as the density estimation is conducted first. SI always outperforms KNN and NI for density estimation. Different combinations of algorithms perform differently on each road. We use A-B to denote the method that uses A for density estimation and B for speed estimation. NI-LR2 on I-80, SI-LR2 on US-101 and SI-RF on Lankershim Blvd outperform the rest of the methods in terms of speed estimation. Overall, the SI-LR2 generates accurate estimation for all three roads.

5.5. Impact of Sensing Power

We analyze the impact of sensing power of AVs on the estimation accuracy. Recalling Sections 2.2 and 3.5, we consider three levels of perception for AVs. Based on Equations (5)–(7), more entries in the time-space region are directly observed when the

perception level increases. We run the proposed estimation method with different perception levels and different methods for speed estimation. Other settings are the same as the baseline setting. The heatmap of MSAPE1 for each road is presented in Figure 17.

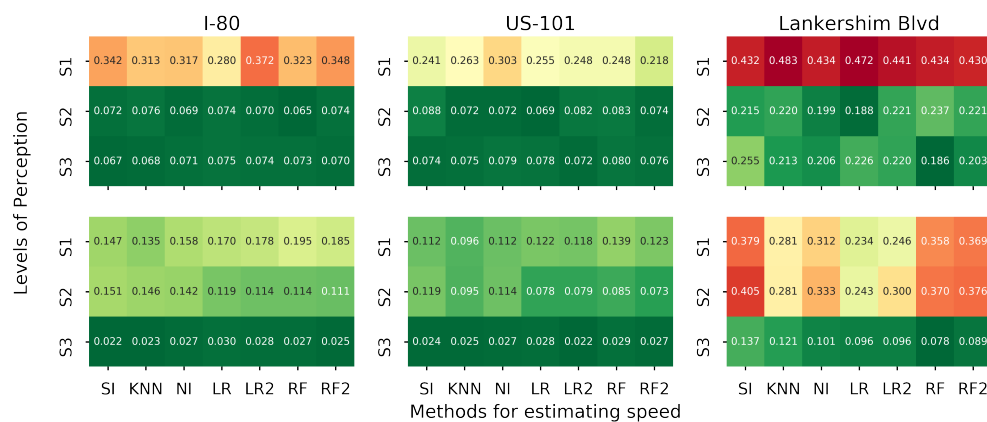


Figure 17. Estimation accuracy under three levels of perception (in terms of MSAPE1, first row: density, second row: speed).

As shown in Figure 17, the proposed method performs the best on US-101 and the worst on Lankershim Blvd. With 5% market penetration rate, at least S2 is required for I-80 and US-101 to obtain an accurate traffic state estimation. Similarly, S3 is required for Lankershim Blvd to ensure the estimation quality. Later, we will discuss the impact of market penetration rate on the estimation accuracy under different perception levels.

The estimation accuracy improves for all speed estimation algorithms and all three roads when the perception level increases. Different speed estimation algorithms perform differently on different roads within the same perception level. For example, in S2, the imputation-based methods outperform the regression-based method on I-80 and US-101, while the Lasso regression outperforms the rest on Lankershim Blvd. In S3, all the density estimation methods perform similarly on I-80 and US-101, while the regression-based method significantly outperforms the imputation-based methods on Lankershim Blvd in terms of density estimation.

5.6. Impact of AV Market Penetration Rate

To examine the impact of AV market penetration rate, we run the proposed method with different market penetration rates ranging from 0.03 to 0.7, and the rest of the settings are the baseline settings. The experimental results are presented in Figure 18.

Generally, the estimation accuracy increases when the AV market penetration rate increases. Moreover, 5% penetration rate is a tipping point for an accurate estimation for I-80 and US-101, while Lankershim Blvd requires larger penetration rate. To further investigate the impact of market penetration rate under different levels of perception, we run the experiment with different penetration rate under three levels of perception, and the results are presented in Figure 19.

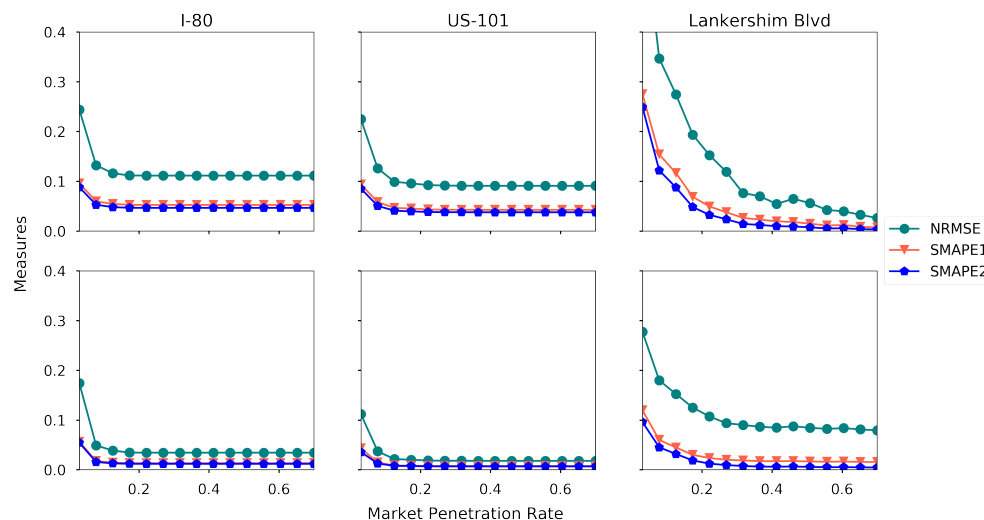


Figure 18. Estimation accuracy under different AV market penetration rate (first row: density, second row: speed).

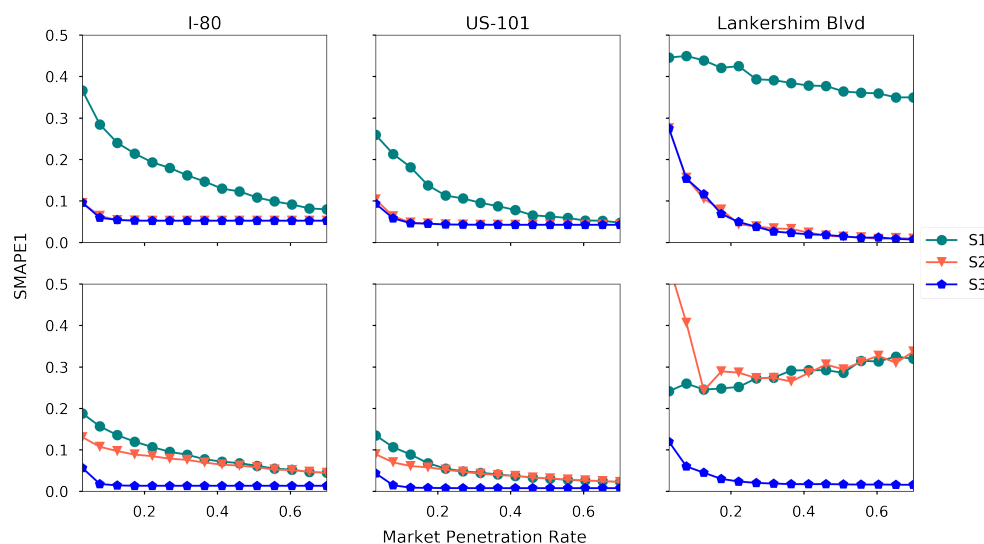


Figure 19. SMAPE1 under different market penetration rates and perception levels (first row: density, second row: speed).

One can read that S2 and S3 yield the same density estimation, as the vehicle detection is enough for density estimation. Better speed estimation can be achieved on S3, since more vehicles are tracked and the speeds are measured. Again, Figure 19 indicates that at least S2 is required for I-80 and US-101 to obtain an accurate traffic state estimation, and S3 is required for Lankershim Blvd to ensure the estimation quality. For S1 and S2 in Lankershim Blvd, the estimation accuracy for speed reduces when the market penetration rate increases, probably due to the overfitting issue of the LR2 method.

We remark that AVs in S1 level are equivalent to connected vehicles with spacing measurements [46], and hence Figure 19 also presents a comparison between the proposed framework and the existing method. The results demonstrate that by using more information collected by AVs, the proposed framework outperforms the existing methods significantly when the market penetration rate is low.

In addition to the above findings, another interesting finding is that when the market penetration rate is low, the regression methods usually outperform the matrix completion-based methods, while the matrix completion-based methods outperform the regression-based methods when the market penetration rate is high.

5.7. Platooning

In the baseline setting, AVs are uniformly distributed in the fleet, while many studies suggest that a dedicated lane for platooning can further enhance mobility [88]. In this case, AVs are not uniformly distributed on the road. To simulate the dedicated lane, we view all vehicles on Lane 1 of I-80, Lane 1 of US-101, and Lane 1,2 of Lankershim Blvd as AVs, and all the vehicles on other lanes as conventional vehicles. To compare, we also set another scenario with the same number of vehicles, which are treated as AVs, uniformly distributed on roads. We run the proposed method on both scenarios with the rest of settings being the baseline setting, and the results are presented in Table 8.

Table 8. Estimation accuracy with AVs on dedicated lanes and uniform distribution (A/B: A is for the uniform distribution scenario, B is for the dedicated lane scenario, unit: %).

Measures	Density			Speed		
	NRMSE	SMAPE1	SMAPE2	NRMSE	SMAPE1	SMAPE2
I-80	31.71/31.65	12.45/12.47	11.86/11.85	19.78/19.84	8.08/7.81	6.92/6.95
US-101	25.93/25.91	10.18/10.16	9.62/9.61	15.13/15.15	6.05/6.05	4.90/4.91
Lankershim	66.67/66.71	31.17/31.58	28.89/28.70	38.99/37.60	20.03/19.74	16.00/15.67

As can be seen from Table 8, the distribution of AVs has a marginal impact on the estimation accuracy. The proposed method performs similarly on the scenarios of the dedicated lane and uniform distribution for all three roads, which is probably because the detection range of LiDAR is large enough to cover the width of the roads.

5.8. Effects of Sensing Errors

As the object detection and tracking depend on the accuracy of sensors and algorithms, the ability of AVs in sensing varies. In this section, we study the effects of sensing errors on the estimation accuracy, and the sensing errors can be categorized into detection missing rate, speed detection noise, and distance measurement errors.

Detection missing rate. The AVs might overlook a certain vehicle during the detection, and we use the missing rate to denote the probability. We examine the impact of the missing rate by running the proposed estimation method with different missing rate ranging from 0.01 to 0.9, and the rest of the settings are the baseline settings. We plot the estimation accuracy for each road separately, as presented in Figure 20.

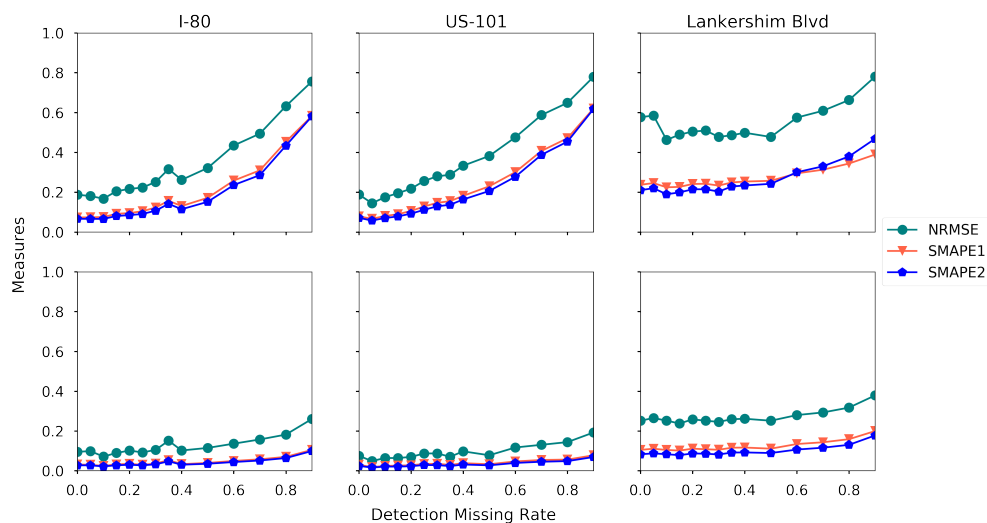


Figure 20. Estimation accuracy under different detection missing rate (first row: density, second row: speed).

From Figure 20, one can read that the estimation error increases when the missing rate

increases for all three roads. The density estimation is much more sensitive to the missing rate than the speed estimation. This is because overlooking vehicles have a significant impact on density estimation, while speed estimation only needs a small fraction of vehicles being observed.

Noise level in speed detection. We further look at the impact of noise in speed detection. We assume that the speed of a vehicle is detected with noise, and the noise level is denoted as ξ . If the true vehicle speed is ν , we sample $\bar{\xi}$ from the uniform distribution $\text{Unif}(-\xi, \xi)$, and then the detected vehicle speed is assumed to be $\nu + \nu\bar{\xi}$. The reason that we define this noise is that the observation noise is usually proportional to the scale of the observation. We run the proposed estimation method by sweeping ξ from 0.0 to 0.4, and the rest of the settings are baseline settings. The estimation accuracy is presented in Figure 21.

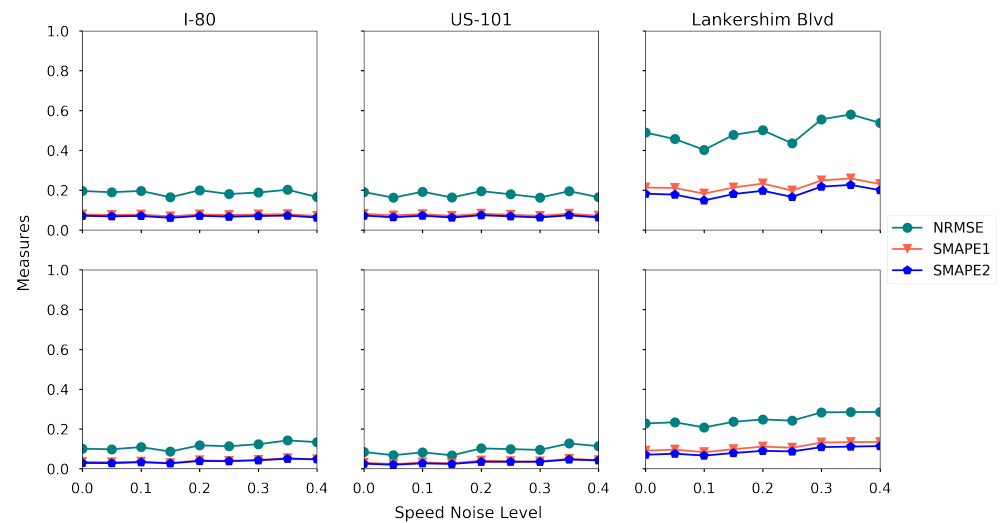


Figure 21. Estimation accuracy under different levels of speed noise (first row: density, second row: speed).

Surprisingly, the proposed method is robust to the noise in speed detection, as the estimation errors remain stable when the speed noise level increases. One explanation for this is that the speed of each cell is computed by averaging the detected speeds from multiple vehicles, hence the detection noise is complemented and reduced based on the law of large numbers.

Distance measurement errors. When AVs detect or track certain vehicles, it measures the distance between itself and the detected/tracked vehicles in order to locate the vehicles. The distance measurement is either conducted by the sensors (e.g., LiDAR) or computer vision algorithms, and hence the measurement might incur errors.

We categorize the distance measurement errors into two components: (1) frequency of the error happening, which is quantified by the percentage of the distance measurement that are associated with an error; (2) magnitude of the error, which is quantified by the number of cells that are offset from the true cell. For example, we assume that the distance measurement is 10% and 5 cells, that means 10% of the distance measures are associated with an error, and the detected location is at most 5 cells away from the true cell (the cell in which the detected vehicle is actually located).

To quantify the effects of the distance measurement errors, two experiments are conducted. Keeping the other settings the same as the baseline setting, we set the magnitude of the distance measurement as 5 cells, and then we vary the percentage of the errors from 5% to 50%. The results are presented in Figure 22. Similarly, we set the percentage of the errors to be 10% and vary the magnitude from 1 cell to 9 cells, and the results are presented in Figure 23.

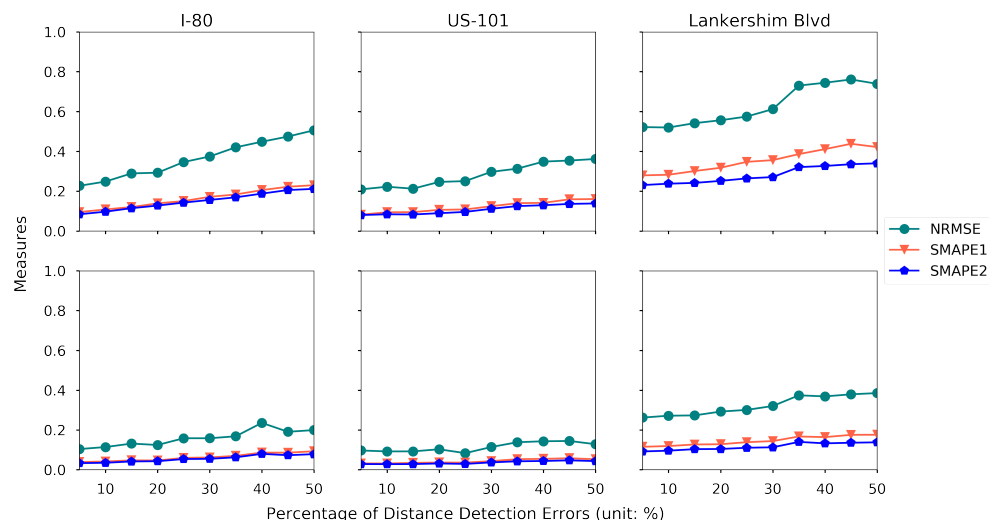


Figure 22. Estimation accuracy under different percentages of distance measurement errors (first row: density, second row: speed).

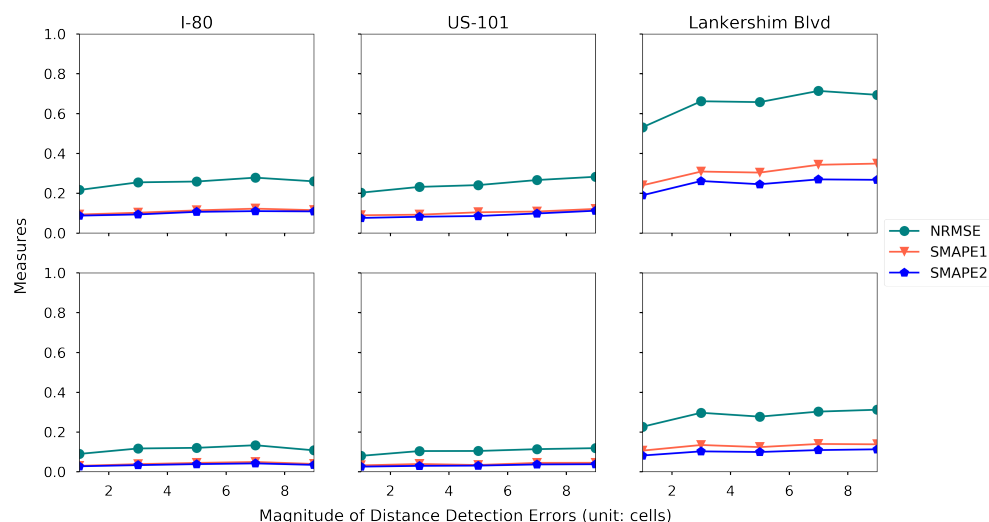


Figure 23. Estimation accuracy under different magnitude of distance measurement errors (first row: density, second row: speed).

Both figures indicate that the increasing frequency and magnitude of the distance measurement errors could reduce the estimation accuracy. The proposed framework is more robust on the magnitude of the measurement error, while it is more sensitive to the frequency of the measurement error.

5.9. Sensitivity Analysis

In this section, we examine the sensitivity of other important factors, (e.g., LiDAR detection range, sampling rate, and discretization size) in our experiments.

LiDAR detection range. The detection range of LiDAR varies in a wide range for different brands [89]. We run the proposed estimation method using different detection range ranging from 10 m to 70 m, and the rest of the settings are the baseline settings. The estimation accuracy for each road is presented in Figure 24.

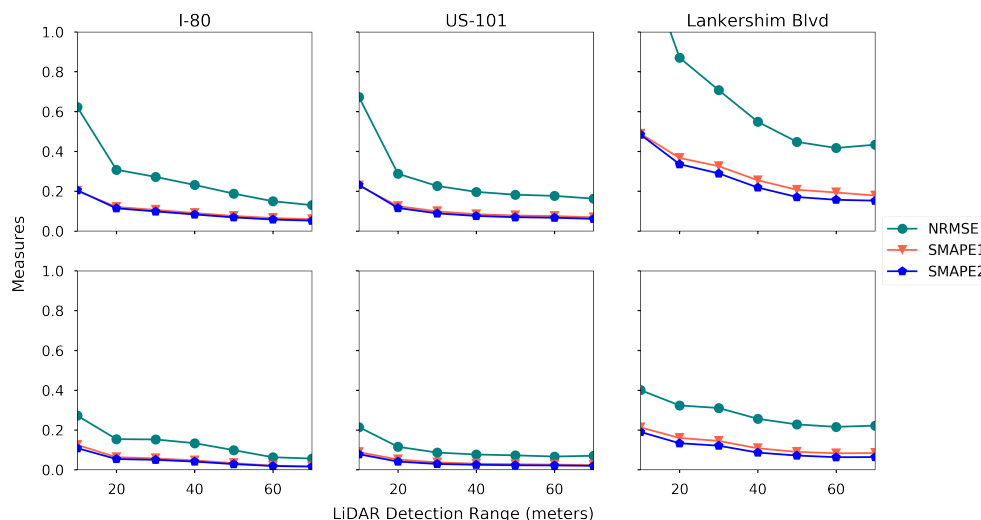


Figure 24. Estimation accuracy with different LiDAR detection range (first row: density, second row: speed).

One can read that the estimation error reduces for both density and speed when the detection range increases. The gain in estimation accuracy becomes marginal when the detection range is large. For example, when the detection range exceeds 40 m on US-101, the improvement of the estimation accuracy is negligible. Another interesting observation is that, on Lankershim Blvd, even a 70-m detection range cannot yield a good density estimation with a 5% market penetration rate.

Sampling rate. Recall that the sampling rate denotes the frequency of the message (which contains the location/speed of itself and detected vehicles) sent to the data center, as discussed in Section 2.4. When the sampling rate is low, we conjecture that the data center received fewer messages, which increases the estimation error. To verify our conjecture, we run the proposed estimation method with different sampling rate ranging from 0.3 Hz to 10 Hz, and the rest of the settings are the base settings. The estimation accuracy on each road is further plotted in Figure 25.

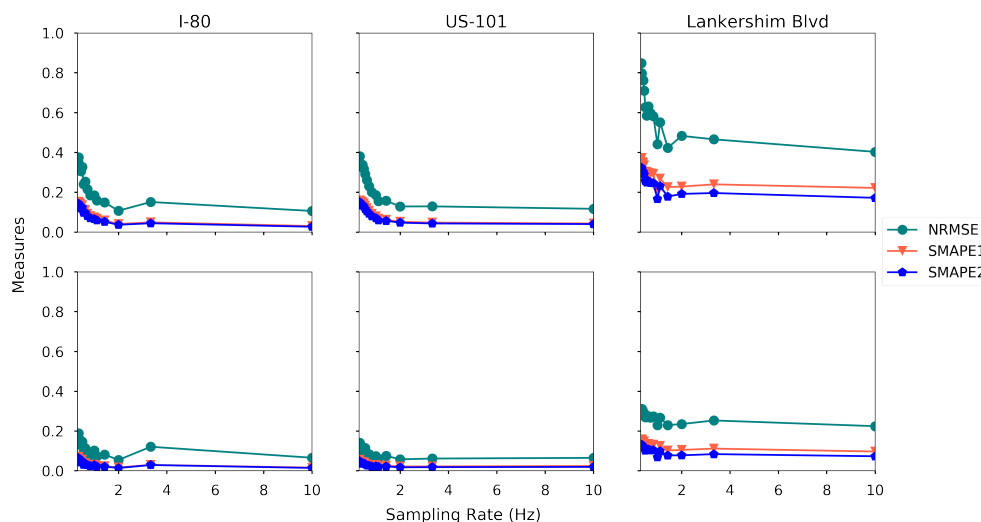


Figure 25. Estimation accuracy with different sampling rate (first row: density, second row: speed).

The estimation accuracy increases when the sampling rate increases for all three roads, as expected. The density estimation is more sensitive to the sampling rate than the speed estimation. This is probably because the density changes dramatically in time-space region, while the speed is relatively stable.

Different discretization sizes. In this section, we demonstrate how the different discretization sizes affect the estimation accuracy. $|H| = 90$ and $|S| = 60$ in the baseline setting, and we change that to $|H| = 60$ and $|S| = 40$. The other settings remain the same, and the comparison results are presented in Table 9.

Table 9. Estimation accuracy with AVs on baseline discretization and larger segmentation (A/B: A is for the baseline setting, B is for 40/60 discretization, unit: %).

Measures	Density			Speed		
	NRMSE	SMAPE1	SMAPE2	NRMSE	SMAPE1	SMAPE2
I-80	18.61/11.18	7.65/4.51	6.87/4.08	9.40/6.84	3.17/2.34	2.73/2.10
US-101	18.28/13.66	7.76/5.66	6.89/5.25	7.49/6.47	2.88/2.02	2.40/1.89
Lankershim	50.94/39.73	22.73/17.27	19.71/15.39	24.08/20.90	10.01/8.89	8.00/7.72

One can see that a bigger discretization size yields better estimation accuracy, because the speed and density change more stably on larger cells. This result also suggests that higher-granular TSE is more challenging and hence it requires more coverage of the observation data. A proper discretization size should be chosen based on the requirements for the estimation resolution and the available data coverage.

6. Conclusions

This paper proposes a high-resolution traffic sensing framework with probe autonomous vehicles (AVs). The framework leverages the perception power of AVs to estimate the fundamental traffic state variables, namely, flow, density and speed, and the underlying idea is to use AVs as moving observers to detect and track vehicles surrounded by AVs. We discuss the potential usage of each sensor mounted on AVs, and categorize the sensing power of AVs into three levels of perception. The powerful sensing capabilities of those probe AVs enable a data-driven traffic sensing framework which is then rigorously formulated. The proposed framework consists of two steps: (1) directly observation of the traffic states using AVs; (2) data-driven estimation of the unobserved traffic states. In the first step, we define the direct observations under different perception levels. The second step is done by estimating the unobserved density using matrix-completion methods, followed by the estimation of unobserved speed using either matrix completion methods or regression-based methods. The implementation details of the whole framework are further discussed.

The next generation simulation (NGSIM) data are adopted to examine the accuracy and robustness of the proposed framework. The proposed estimation framework is examined extensively on I-80, US-101 and Lankershim Boulevard. In general, the proposed framework estimates the traffic states accurately with a low AV market penetration rate. The speed estimation is always easier than density estimation, as expected. Results show that, with 5% AV market penetration rate, at least S2 is required for I-80 and US-101 to obtain an accurate traffic state estimation, while S3 is required for Lankershim Blvd to ensure the estimation quality. During the estimation of speed, all the coefficients in the Lasso regression are consistent with the fundamental diagrams. In addition, a sensitivity analysis regarding AV penetration rates, sensor configurations, speed detection noise and perception accuracy is conducted.

This study would help policymakers and private sectors (e.g Uber, Waymo and other AV manufacturers) understand the values of AVs in traffic operation and management, especially the values of massive data collected by AVs. Hopefully, new business models to commercializing the data [90] or collaborations between private sectors and public agencies can be established for smart communities. In the near future, we will examine the sensing capabilities of AVs at network level and extend the proposed traffic sensing framework to large-scale networks. We also plan to develop a traffic simulation environment to enable the comprehensive analysis of the proposed framework under different traffic conditions.

Another interesting research direction is to investigate the privacy issue when AVs share the observed information with the data center.

Author Contributions: Conceptualization, W.M. and S.Q.; methodology, W.M. and S.Q.; validation, W.M. and S.Q.; formal analysis, W.M. and S.Q.; resources, S.Q.; data curation, W.M.; writing—original draft preparation, W.M.; writing—review and editing, M.W. and S.Q.; visualization, W.M.; supervision, S.Q.; project administration, S.Q.; funding acquisition, S.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research is funded in part by the NSF grant CMMI-1751448 and Carnegie Mellon University’s Mobility 21, a National University Transportation Center for Mobility sponsored by the US Department of Transportation, and by a grant funded by the Hong Kong Polytechnic University (Project No. P0033933). The contents of this paper reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The proposed traffic sensing framework is implemented in Python and open-sourced on Github (<https://github.com/Lemma1/NGSIM-interface>). Readers can reproduce all the experiments in Section 5. Additionally, the Github repository also contains some analysis that is omitted in Section 5.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Determining the Detection Areas by AVs

Suppose the detection area of AV $i \in I^A$ at time t is $R^i(t)$, and $R^i(t)$ consists of two parts, which are the detection area for preceding vehicles (R^{i,D_1}) and the detection area for surrounding vehicles (R^{i,D_2}), as discussed in Section 2.3. We further denote the detection area D_j of vehicle i in road segment s on lane l by $R_l^{i,D_j}(t, s)$, as presented in Equation (A1).

$$R_l^{i,D_j}(t, s) = R^{i,D_j}(t) \cap X_{ls}, \forall j \in \{1, 2\}, i \in I^A \quad (\text{A1})$$

As discussed in Section 2.4, the detected information of all AVs will be aggregated by the data center. Hence the whole detection area by all AVs is denoted by $R_l^{D_j}(t, s)$, as presented in Equation (A2).

$$R_l^{D_j}(t, s) = \cup_{i \in I^A} R_l^{i,D_j}(t, s), \forall j \in \{1, 2\} \quad (\text{A2})$$

The next step is to discretize the detection area into the time-space region. We define $O_l^{D_j}$ as the set of time-space indices (h, s) such that X_{ls} is covered by the detection range D_j in time interval h , as presented in Equation (A3).

$$O_l^{D_j} = \{(h, s) | \exists t \in T_h, \text{ s.t. } \mu(R_l^{D_j}(t, s)) \geq (1 - \varepsilon)\mu(X_{ls})\}, j = \{1, 2\} \quad (\text{A3})$$

where ε is the tolerance and is set to 0.05. In the main paper, we simplify the notation such that $O_l^j = O_l^{D_j}, \forall j \in \{1, 2\}$.

We further have $T_h^o(l, s) = \{t \in T_h | \mu(R_l^{D_2}(t, s)) \geq (1 - \varepsilon)\mu(X_{ls})\}$ to represent the set of time indices when X_{ls} is covered by the D_2 in T_h , and $I_l^o(t, s) = \{i \in I | x^i(t) \in R_l^{D_2}(t, s)\}$ represents the set of vehicles detected by D_2 in X_{ls} at time t .

References

- Litman, T. *Autonomous Vehicle Implementation Predictions*; Victoria Transport Policy Institute Victoria: Victoria, BC, Canada, 2017.
- Assidiq, A.A.; Khalifa, O.O.; Islam, M.R.; Khan, S. Real time lane detection for autonomous vehicles. In Proceedings of the 2008 International Conference on Computer and Communication Engineering, Kuala Lumpur, Malaysia, 13–15 May 2008; pp. 82–88.
- Tientrakool, P.; Ho, Y.C.; Maxemchuk, N.F. Highway capacity benefits from using vehicle-to-vehicle communication and sensors for collision avoidance. In Proceedings of the 2011 IEEE Vehicular Technology Conference (VTC Fall), San Francisco, CA, USA, 5–8 September 2011; pp. 1–5.
- Stern, R.E.; Cui, S.; Delle Monache, M.L.; Bhadani, R.; Bunting, M.; Churchill, M.; Hamilton, N.; Pohlmann, H.; Wu, F.; Piccoli, B.; et al. Dissipation of stop-and-go waves via control of autonomous vehicles: Field experiments. *Transp. Res. Part C Emerg. Technol.* **2018**, *89*, 205–221. [\[CrossRef\]](#)
- Vahidi, A.; Sciarretta, A. Energy saving potentials of connected and automated vehicles. *Transp. Res. Part C Emerg. Technol.* **2018**, *95*, 822–843. [\[CrossRef\]](#)
- Gawron, J.H.; Keoleian, G.A.; De Kleine, R.D.; Wallington, T.J.; Kim, H.C. Life cycle assessment of connected and automated vehicles: Sensing and computing subsystem and vehicle level effects. *Environ. Sci. Technol.* **2018**, *52*, 3249–3256. [\[CrossRef\]](#) [\[PubMed\]](#)
- Chen, Z.; He, F.; Zhang, L.; Yin, Y. Optimal deployment of autonomous vehicle lanes with endogenous market penetration. *Transp. Res. Part C Emerg. Technol.* **2016**, *72*, 143–156. [\[CrossRef\]](#)
- Chen, Z.; He, F.; Yin, Y.; Du, Y. Optimal design of autonomous vehicle zones in transportation networks. *Transp. Res. Part B Methodol.* **2017**, *99*, 44–61. [\[CrossRef\]](#)
- Duarte, F.; Ratti, C. The impact of autonomous vehicles on cities: A review. *J. Urban Technol.* **2018**, *25*, 3–18. [\[CrossRef\]](#)
- Zhang, W.; Guhathakurta, S.; Fang, J.; Zhang, G. Exploring the impact of shared autonomous vehicles on urban parking demand: An agent-based simulation approach. *Sustain. Cities Soc.* **2015**, *19*, 34–45. [\[CrossRef\]](#)
- Harper, C.D.; Hendrickson, C.T.; Samaras, C. Exploring the Economic, Environmental, and Travel Implications of Changes in Parking Choices due to Driverless Vehicles: An Agent-Based Simulation Approach. *J. Urban Plan. Dev.* **2018**, *144*, 04018043. [\[CrossRef\]](#)
- Millard-Ball, A. The autonomous vehicle parking problem. *Transp. Policy* **2019**, *75*, 99–108. [\[CrossRef\]](#)
- Lutin, J.M. Not If, but When: Autonomous Driving and the Future of Transit. *J. Public Transp.* **2018**, *21*, 10. [\[CrossRef\]](#)
- Salonen, A.O. Passenger's subjective traffic safety, in-vehicle security and emergency management in the driverless shuttle bus in Finland. *Transp. Policy* **2018**, *61*, 106–110. [\[CrossRef\]](#)
- Korosec, K. Waymo's Autonomous Vehicles Are Driving 25,000 Miles Every Day. 2018. Available online: <https://techcrunch.com/2018/07/20/waymos-autonomous-vehicles-are-driving-25000-miles-every-day/> (accessed on 10 July 2019).
- Korosec, K. Uber Reboots Its Self-Driving Car Program. 2018. Available online: <https://techcrunch.com/2018/12/20/uber-self-driving-car-testing-resumes-pittsburgh/> (accessed on 10 July 2019).
- Zhao, L.; Malikopoulos, A.A. Enhanced Mobility with Connectivity and Automation: A Review of Shared Autonomous Vehicle Systems. *arXiv* **2019**, arXiv:1905.12602.
- Levin, M.W. Congestion-aware system optimal route choice for shared autonomous vehicles. *Transp. Res. Part C Emerg. Technol.* **2017**, *82*, 229–247. [\[CrossRef\]](#)
- Wang, J.; Peeta, S.; He, X. Multiclass traffic assignment model for mixed traffic flow of human-driven vehicles and connected and autonomous vehicles. *Transp. Res. Part B Methodol.* **2019**, *126*, 139–168. [\[CrossRef\]](#)
- Shida, M.; Nemoto, Y. Development of a small-distance vehicle platooning system. In Proceedings of the 16th ITS World Congress and Exhibition on Intelligent Transport Systems and Services ITS America ERTICO ITS Japan, Stockholm, Sweden, 21–25 September 2009.
- Li, X.; Ghiasi, A.; Xu, Z.; Qu, X. A piecewise trajectory optimization model for connected automated vehicles: Exact optimization algorithm and queue propagation analysis. *Transp. Res. Part B Methodol.* **2018**, *118*, 429–456. [\[CrossRef\]](#)
- Yu, C.; Feng, Y.; Liu, H.X.; Ma, W.; Yang, X. Integrated optimization of traffic signals and vehicle trajectories at isolated urban intersections. *Transp. Res. Part B Methodol.* **2018**, *112*, 89–112. [\[CrossRef\]](#)
- Li, S.E.; Zheng, Y.; Li, K.; Wang, J. An overview of vehicular platoon control under the four-component framework. In Proceedings of the 2015 IEEE Intelligent Vehicles Symposium (IV), Seoul, Korea, 28 June–1 July 2015; pp. 286–291.
- Gong, S.; Shen, J.; Du, L. Constrained optimization and distributed computation based car following control of a connected and autonomous vehicle platoon. *Transp. Res. Part B Methodol.* **2016**, *94*, 314–334. [\[CrossRef\]](#)
- Chong, Z.; Qin, B.; Bandyopadhyay, T.; Wongpiromsarn, T.; Rankin, E.; Ang, M.; Frazzoli, E.; Rus, D.; Hsu, D.; Low, K. Autonomous personal vehicle for the first-and last-mile transportation services. In Proceedings of the 2011 IEEE 5th International Conference on Cybernetics and Intelligent Systems (CIS), Qingdao, China, 17–19 September 2011; pp. 253–260.
- Moorthy, A.; De Kleine, R.; Keoleian, G.; Good, J.; Lewis, G. Shared autonomous vehicles as a sustainable solution to the last mile problem: A case study of ann arbor-detroit area. *SAE Int. J. Passeng.-Cars-Electron. Electr. Syst.* **2017**, *10*, 328–336. [\[CrossRef\]](#)
- Van Brummelen, J.; O'Brien, M.; Gruyer, D.; Najjaran, H. Autonomous vehicle perception: The technology of today and tomorrow. *Transp. Res. Part C Emerg. Technol.* **2018**, *89*, 384–406. [\[CrossRef\]](#)
- Pendleton, S.; Andersen, H.; Du, X.; Shen, X.; Meghjani, M.; Eng, Y.; Rus, D.; Ang, M. Perception, planning, control, and coordination for autonomous vehicles. *Machines* **2017**, *5*, 6. [\[CrossRef\]](#)

29. Seo, T.; Bayen, A.M.; Kusakabe, T.; Asakura, Y. Traffic state estimation on highway: A comprehensive survey. *Annu. Rev. Control.* **2017**, *43*, 128–151. [\[CrossRef\]](#)
30. Ma, W.; Qian, S. Measuring and reducing the disequilibrium levels of dynamic networks with ride-sourcing vehicle data. *Transp. Res. Part C Emerg. Technol.* **2020**, *110*, 222–246. [\[CrossRef\]](#)
31. Xu, S.; Chen, X.; Pi, X.; Joe-Wong, C.; Zhang, P.; Noh, H.Y. iLOCuS: Incentivizing Vehicle Mobility to Optimize Sensing Distribution in Crowd Sensing. *IEEE Trans. Mob. Comput.* **2019**. [\[CrossRef\]](#)
32. Mahmoudzadeh, A.; Golroo, A.; Jahanshahi, M.R.; Firoozi Yeganeh, S. Estimating Pavement Roughness by Fusing Color and Depth Data Obtained from an Inexpensive RGB-D Sensor. *Sensors* **2019**, *19*, 1655. [\[CrossRef\]](#) [\[PubMed\]](#)
33. Alipour, M.; Harris, D.K. A big data analytics strategy for scalable urban infrastructure condition assessment using semi-supervised multi-transform self-training. *J. Civ. Struct. Health Monit.* **2020**, 1–20. [\[CrossRef\]](#)
34. Sun, Z.; Jin, W.L.; Ritchie, S.G. Simultaneous estimation of states and parameters in Newell’s simplified kinematic wave model with Eulerian and Lagrangian traffic data. *Transp. Res. Part B Methodol.* **2017**, *104*, 106–122. [\[CrossRef\]](#)
35. Jain, N.K.; Saini, R.; Mittal, P. A Review on Traffic Monitoring System Techniques. In *Soft Computing: Theories and Applications*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 569–577.
36. Antoniou, C.; Balakrishna, R.; Koutsopoulos, H.N. A synthesis of emerging data collection technologies and their impact on traffic management applications. *Eur. Transp. Res. Rev.* **2011**, *3*, 139–148. [\[CrossRef\]](#)
37. Wardrop, J.G.; Charlesworth, G. A method of estimating speed and flow of traffic from a moving vehicle. *Proc. Inst. Civ. Eng.* **1954**, *3*, 158–171. [\[CrossRef\]](#)
38. Wang, Y.; Papageorgiou, M. Real-time freeway traffic state estimation based on extended Kalman filter: A general approach. *Transp. Res. Part B Methodol.* **2005**, *39*, 141–167. [\[CrossRef\]](#)
39. Wright, C. A theoretical analysis of the moving observer method. *Transp. Res.* **1973**, *7*, 293–311. [\[CrossRef\]](#)
40. Zheng, Y. Trajectory data mining: An overview. *ACM Trans. Intell. Syst. Technol. (TIST)* **2015**, *6*, 29. [\[CrossRef\]](#)
41. O’Keeffe, K.P.; Anjomshoa, A.; Strogatz, S.H.; Santi, P.; Ratti, C. Quantifying the sensing power of vehicle fleets. *Proc. Natl. Acad. Sci. USA* **2019**, *116*, 201821667. [\[CrossRef\]](#) [\[PubMed\]](#)
42. Herrera, J.C.; Bayen, A.M. Incorporation of Lagrangian measurements in freeway traffic state estimation. *Transp. Res. Part B Methodol.* **2010**, *44*, 460–481. [\[CrossRef\]](#)
43. van Erp, P.B.; Knoop, V.L.; Hoogendoorn, S.P. Macroscopic traffic state estimation using relative flows from stationary and moving observers. *Transp. Res. Part B Methodol.* **2018**, *114*, 281–299. [\[CrossRef\]](#)
44. Wilby, M.R.; Díaz, J.J.V.; Rodríguez González, A.B.; Sotelo, M.Á. Lightweight occupancy estimation on freeways using extended floating car data. *J. Intell. Transp. Syst.* **2014**, *18*, 149–163. [\[CrossRef\]](#)
45. Seo, T.; Kusakabe, T.; Asakura, Y. Traffic state estimation with the advanced probe vehicles using data assimilation. In Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems, Las Palmas, Spain, 15–18 September 2015; pp. 824–830.
46. Seo, T.; Kusakabe, T.; Asakura, Y. Estimation of flow and density using probe vehicles with spacing measurement equipment. *Transp. Res. Part C Emerg. Technol.* **2015**, *53*, 134–150. [\[CrossRef\]](#)
47. Fountoulakis, M.; Bekiaris-Liberis, N.; Roncoli, C.; Papamichail, I.; Papageorgiou, M. Highway traffic state estimation with mixed connected and conventional vehicles: Microscopic simulation-based testing. *Transp. Res. Part C Emerg. Technol.* **2017**, *78*, 13–33. [\[CrossRef\]](#)
48. Puri, A. *A Survey of Unmanned Aerial Vehicles (UAV) for Traffic Surveillance*; Department of Computer Science and Engineering, University of South Florida: Tampa, FL, USA, 2005; pp. 1–29.
49. Kanistras, K.; Martins, G.; Rutherford, M.J.; Valavanis, K.P. Survey of unmanned aerial vehicles (UAVs) for traffic monitoring. In *Handbook of Unmanned Aerial Vehicles*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 2643–2666.
50. Ke, R.; Li, Z.; Tang, J.; Pan, Z.; Wang, Y. Real-time traffic flow parameter estimation from UAV video based on ensemble classifier and optical flow. *IEEE Trans. Intell. Transp. Syst.* **2018**, *20*, 54–64. [\[CrossRef\]](#)
51. Zhu, J.; Sun, K.; Jia, S.; Li, Q.; Hou, X.; Lin, W.; Liu, B.; Qiu, G. Urban Traffic Density Estimation Based on Ultrahigh-Resolution UAV Video and Deep Neural Network. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 4968–4981. [\[CrossRef\]](#)
52. Khan, M.; Ectors, W.; Bellemans, T.; Janssens, D.; Wets, G. Unmanned aerial vehicle-based traffic analysis: A case study for shockwave identification and flow parameters estimation at signalized intersections. *Remote Sens.* **2018**, *10*, 458. [\[CrossRef\]](#)
53. Jin, P.J.; Ardestani, S.M.; Wang, Y.; Hu, W. *Unmanned Aerial Vehicle (UAV) Based Traffic Monitoring and Management*; Technical Report; Rutgers University, Center for Advanced Infrastructure and Transportation: Piscataway, NJ, USA, 2016.
54. Niu, H.; Gonzalez-Prelcic, N.; Heath, R.W. A UAV-Based Traffic Monitoring System-Invited Paper. In Proceedings of the 2018 IEEE 87th Vehicular Technology Conference (VTC Spring), Porto, Portugal, 3–6 June 2018; pp. 1–5.
55. Li, M.; Zhen, L.; Wang, S.; Lv, W.; Qu, X. Unmanned aerial vehicle scheduling problem for traffic monitoring. *Comput. Ind. Eng.* **2018**, *122*, 15–23. [\[CrossRef\]](#)
56. Liu, X.; Peng, Z.R.; Zhang, L.Y. Real-time UAV Rerouting for Traffic Monitoring with Decomposition Based Multi-objective Optimization. *J. Intell. Robot. Syst.* **2019**, *94*, 491–501. [\[CrossRef\]](#)
57. Chen, B.; Yang, Z.; Huang, S.; Du, X.; Cui, Z.; Bhimani, J.; Xie, X.; Mi, N. Cyber-physical system enabled nearby traffic flow modelling for autonomous vehicles. In Proceedings of the 2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC), San Diego, CA, USA, 10–12 December 2017; pp. 1–6.

58. Qian, S.; Yang, S.; Plummer, A. *High-Resolution Ubiquitous Community Sensing with Autonomous Vehicles*; Technical Report; Carnegie Mellon University and Uber ATG: Pittsburgh, PA, USA, 2019.
59. Thai, J.; Bayen, A.M. State estimation for polyhedral hybrid systems and applications to the Godunov scheme for highway traffic estimation. *IEEE Trans. Autom. Control.* **2014**, *60*, 311–326. [[CrossRef](#)]
60. Herring, R.; Hofleitner, A.; Abbeel, P.; Bayen, A. Estimating arterial traffic conditions using sparse probe data. In Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems, Funchal, Portugal, 19–22 September 2010; pp. 929–936.
61. Yu, J.; Stettler, M.E.; Angeloudis, P.; Hu, S.; Chen, X.M. Urban network-wide traffic speed estimation with massive ride-sourcing GPS traces. *Transp. Res. Part C Emerg. Technol.* **2020**, *112*, 136–152. [[CrossRef](#)]
62. Shan, Z.; Zhu, Q. Camera location for real-time traffic state estimation in urban road network using big GPS data. *Neurocomputing* **2015**, *169*, 134–143. [[CrossRef](#)]
63. Bautista, C.M.; Dy, C.A.; Mañalac, M.I.; Orbe, R.A.; Cordel, M. Convolutional neural network for vehicle detection in low resolution traffic videos. In Proceedings of the 2016 IEEE Region 10 Symposium (TENSYP), Bali, Indonesia, 9–11 May 2016; pp. 277–281.
64. Aly, M. Real time detection of lane markers in urban streets. In Proceedings of the 2008 IEEE Intelligent Vehicles Symposium, Eindhoven, The Netherlands, 4–6 June 2008; pp. 7–12.
65. Dollár, P.; Wojek, C.; Perona, P.; Schiele, B. Pedestrian Detection: A new Benchmark. In Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 24–27 June 2009; pp. 304–311.
66. Takatori, Y.; Hasegawa, T. Stand-alone collision warning systems based on information from on-board sensors: Evaluating performance relative to system penetration rate. *IATSS Res.* **2006**, *30*, 39–47. [[CrossRef](#)]
67. Thakur, R. *Infrared Sensors for Autonomous Vehicles*. In *Recent Development in Optoelectronic Devices*; InTech: London, UK, 2018; [[CrossRef](#)]
68. SAE. *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*; SAE: Warrendale, PA, USA, 2018. [[CrossRef](#)]
69. Milan, A.; Leal-Taixé, L.; Reid, I.; Roth, S.; Schindler, K. MOT16: A benchmark for multi-object tracking. *arXiv* **2016**, arXiv:1603.00831.
70. Dale, A. Autonomous Driving-The Changes to Come. 2018. Available online: <https://cdn.ihs.com/www/pdf/Autonomous-Driving-changes-to-come-Aaron-Dale.pdf> (accessed on 27 March 2019).
71. nuScenes. nuScenes: Data Collection. 2018. Available online: <https://www.nuscenes.org/nuscenes> (accessed on 27 March 2019).
72. Waymo Team. Introducing Waymo’s Suite of Custom-Built, Self-Driving Hardware. 2017. Available online: <https://blog.waymo.com/2019/08/introducing-waymos-suite-of-custom.html> (accessed on 27 March 2019).
73. Wolcott, R.W.; Eustice, R.M. Fast LIDAR localization using multiresolution Gaussian mixture maps. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 2814–2821.
74. Edie, L.C. *Discussion of Traffic Stream Measurements and Definitions*; Port of New York Authority: New York, NY, USA, 1963.
75. Bressan, A.; Nguyen, K.T. Conservation law models for traffic flow on a network of roads. *NHM* **2015**, *10*, 255–293. [[CrossRef](#)]
76. Troyanskaya, O.; Cantor, M.; Sherlock, G.; Brown, P.; Hastie, T.; Tibshirani, R.; Botstein, D.; Altman, R.B. Missing value estimation methods for DNA microarrays. *Bioinformatics* **2001**, *17*, 520–525. [[CrossRef](#)]
77. Hastie, T.; Mazumder, R.; Lee, J.D.; Zadeh, R. Matrix completion and low-rank SVD via fast alternating least squares. *J. Mach. Learn. Res.* **2015**, *16*, 3367–3402.
78. Newell, G.F. A simplified theory of kinematic waves in highway traffic, part I: General theory. *Transp. Res. Part B Methodol.* **1993**, *27*, 281–287. [[CrossRef](#)]
79. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B (Methodol.)* **1996**, *58*, 267–288. [[CrossRef](#)]
80. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
81. Strobl, C. Dimensionally extended nine-intersection model (de-9im). *Encycl. GIS* **2017**, 470–476. [[CrossRef](#)]
82. Kanagal, B.; Sindhvani, V. Rank Selection in Low-Rank Matrix Approximations: A Study of Cross-Validation for NMFs. Available online: <http://www.vikas.sindhvani.org/nmfvcv.pdf> (accessed on 10 January 2021)
83. Alexiadis, V.; Colyar, J.; Halkias, J.; Hranac, R.; McHale, G. The next generation simulation program. *Inst. Transp. Eng. ITE J.* **2004**, *74*, 22–26.
84. He, Z. Research based on high-fidelity NGSIM vehicle trajectory datasets: A review. *Res. Gate* **2017**, 1–33.
85. FHWA. Next Generation Simulation (NGSIM). 2007. Available online: <https://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm> (accessed on 4 January 2021).
86. Li, L.; Chen, X.; Zhang, L. Multimodel ensemble for freeway traffic state estimations. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 1323–1336. [[CrossRef](#)]
87. Long Cheu, R.; Xie, C.; Lee, D.H. Probe vehicle population and sample size for arterial speed estimation. *Comput.-Aided Civ. Infrastruct. Eng.* **2002**, *17*, 53–60. [[CrossRef](#)]
88. Ramezani, M.; Machado, J.A.; Skabardonis, A.; Geroliminis, N. Capacity and delay analysis of arterials with mixed autonomous and human-driven vehicles. In Proceedings of the 2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS), Naples, Italy, 26–28 June 2017; pp. 280–284.

-
89. Hecht, J. Lidar for Self-Driving Cars. 2018. Available online: https://www.osa-opn.org/home/articles/volume_29/january_2018/features/lidar_for_self-driving_cars (accessed on 10 July 2019).
 90. Kuchinskas, S. *Mobility Data Marketplaces and Integrating Mobility into the Data Economy*; Technical Report; FC Business Intelligence Ltd.: London, UK, 2019. Available online: https://s3.amazonaws.com/external_clips/3027335/Mobility_Marketplace_Report.pdf?1553793639 (accessed on 17 July 2019)