Advanced Driver Assistance System-Equipped Vehicle Datasets Collected in Central Ohio: Final Report

PUBLICATION NO. FHWA-HRT-24-084

MARCH 2024

U.S. Department of Transportation
Federal Highway Administration

Research, Development, and Technology
Turner-Fairbank Highway Research Center
6300 Georgetown Pike
McLean, VA 22101-2296

FOREWORD

The technology available on vehicles traversing our Nation's roadways is rapidly changing in ways that may impact how drivers behave. To be able to plan for these vehicle types of the future, State and local infrastructure owners and operators (IOOs) must first be able to capture how these various vehicle types behave in transportation analysis, modeling, and simulation (AMS) tools. One current issue facing the developers and users of transportation AMS tools is a lack of high-resolution, naturalistic datasets that include advanced driver-assistance system (ADAS)- and automated driving system (ADS)-equipped vehicle operation on roadways.

The purpose of this report is to document the collection of a large dataset collected in Central Ohio using conspicuous and inconspicuous SAE Level 2™ ADAS-equipped vehicles navigating complex driving environments.(1) These publicly available datasets will enable researchers to develop models that characterize human-ADAS interactions under a diverse set of naturalistic traffic scenarios in highway and arterial environments.(2,3) Using these data, researchers will be able to develop appropriate models to assess the impact of ADAS technologies and inform industry on methods to update current traffic microsimulation products. Results from models developed using these datasets will enable researchers, analysts, and IOOs to better understand how ADAS-equipped vehicles will affect transportation system performance. Ultimately, the models developed using these data can support data-informed decisionmaking that maximizes ADAS-equipped vehicle benefits to system performance. This final report will be of interest to researchers and AMS tool users who are interested in improving human-ADAS interactions in decision support tools.

Carl Andersen
Acting Director, Office of Safety and Operations
Research and Development

Notice

This document is disseminated under the sponsorship of the U.S. Department of Transportation in the interest of information exchange. The U.S. Government assumes no liability for the use of the information contained in this document.

Non-Binding Contents

Except for the statutes and regulations cited, the contents of this document do not have the force and effect of law and are not meant to bind the States or the public in any way. This document is intended only to provide information regarding existing requirements under the law or agency policies.

Quality Assurance Statement

The Federal Highway Administration (FHWA) provides high-quality information to serve Government, industry, and the public in a manner that promotes public understanding. Standards and policies are used to ensure and maximize the quality, objectivity, utility, and integrity of its information. FHWA periodically reviews quality issues and adjusts its programs and processes to ensure continuous quality improvement.

Disclaimer for Product Names and Manufacturers

The U.S. Government does not endorse products or manufacturers. Trademarks or manufacturers' names appear in this document only because they are considered essential to the objective of the document. They are included for informational purposes only and are not intended to reflect a preference, approval, or endorsement of any one product or entity.

The U.S. Government does not endorse outside entities, products, or manufacturers. Links to content created by outside entities are provided for informational purposes only and are not intended to reflect a preference, approval, or endorsement of any one entity or product. External sites are not subject to Federal information quality, privacy, security, or accessibility guidelines.

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. FHWA-HRT-24-084	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Advanced Driver Assistance System-Equipped Vehicle Datasets Collected in Central Ohio: Final Report		5. Report Date: March 2024
		6. Performing Organization Code
7. Author(s) Timothy Seitz (ORCID: 0000-0003-2690-8415), Sayali Karanjkar (ORCID: 0000-0002-0575-873X), Dr. Jiaqi Ma (ORCID: 0000-0002-8184-5157), Dr. Xin Xia (ORCID: 0000-0002-5108-7578), Rachel James (ORCID: 0000-0001-9138-510X)		8. Performing Organization Report No.
9. Performing Organization Name and Address Transportation Research Center Inc. 0820 State Route 347 East Liberty, OH 43319		10. Work Unit No.
		11. Contract or Grant No. 693JJ321C000001
12. Sponsoring Agency Name and Address Office of Safety and Operations Research and Development Federal Highway Administration 6300 Georgetown Pike McLean, VA 22101-2296		13. Type of Report and Period Covered: Final Report; January 2020–June 2023
		14. Sponsoring Agency Code HRSO-50, HOIT-1
15. Supplementary Notes The Federal Task Manager was Jennifer Foley (HIT-RC, ORCID:0000-0002-2695-2339).		

16. Abstract

Connected and automated vehicle (CAV) technology has been driving multi-billion-dollar investments for over a decade now. While no agreement exists on the exact timeframe and capabilities of these systems, the transportation industry agrees that CAV adoption will eventually occur. Infrastructure owners and operators (IOOs) are making decisions now that will shape the transportation system of the future and need high-resolution, naturalistic datasets for improved traffic simulation that include CAV operation on roadways. While original equipment manufacturers (OEMs) and vehicle-technology companies do have high volumes of high-resolution datasets, not all the data are publicly available to characterize the behavior of CAVs and their impact on transportation system performance. Ultimately, the lack of well-calibrated traffic simulation tools is preventing IOOs from making necessary infrastructure investment decisions that account for the presence of CAVs and other emerging technologies. This project developed a methodology to collect and process the raw vehicle sensor data to extract trajectory data about the instrumented subject vehicle (SV) and all adjacent vehicles (AdjVs) in the traffic stream that were perceived by the sensors. These publicly available datasets will enable researchers to develop models that characterize human-ADAS interactions under a diverse set of naturalistic traffic scenarios in highway and arterial environments. Using these data, researchers will be able to develop appropriate models to assess the impact of ADAS technologies and update current driver behavioral models using data collected during this project, including trajectories of SVs and AdjVs.

17. Key Words	18. Distribution Statement
Connected and automated vehicles, advanced driver assistance systems, trajectory data, naturalistic data	No restrictions. This document is available to the public through the National Technical Information Service, Springfield, VA 22161. https://www.ntis.gov

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	94	N/A

SI* (MODERN METRIC) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH		
in	inches	25.4	millimeters	mm
ft	feet	0.305	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
		AREA		
in^2	square inches	645.2	square millimeters	mm^2
ft^2	square feet	0.093	square meters	m^2
yd^2	square yard	0.836	square meters	m^2
ac	acres	0.405	hectares	ha
mi^2	square miles	2.59	square kilometers	km^2
		VOLUME		
fl oz	fluid ounces	29.57	milliliters	mL
gal	gallons	3.785	liters	L
ft^3	cubic feet	0.028	cubic meters	m^3
yd^3	cubic yards	0.765	cubic meters	m^3
	NOTE: volumes greater than 1,000 L shall be shown in m^3			
		MASS		
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2,000 lb)	0.907	megagrams (or "metric ton")	Mg (or "t")
		TEMPERATURE (exact degrees)		
°F	Fahrenheit	5 (F-32)/9 or (F-32)/1.8	Celsius	°C
		ILLUMINATION		
fc	foot-candles	10.76	lux	lx
fl	foot-Lamberts	3.426	$candela/m^2$	cd/m^2
		FORCE and PRESSURE or STRESS		
lbf	poundforce	4.45	newtons	N
lbf/in^2	poundforce per square inch	6.89	kilopascals	kPa

APPROXIMATE CONVERSIONS FROM SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH		
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
		AREA		
mm^2	square millimeters	0.0016	square inches	in^2
m^2	square meters	10.764	square feet	ft^2
m^2	square meters	1.195	square yards	yd^2
ha	hectares	2.47	acres	ac
km^2	square kilometers	0.386	square miles	mi^2
		VOLUME		
mL	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m^3	cubic meters	35.314	cubic feet	ft^3
m^3	cubic meters	1.307	cubic yards	yd^3
		MASS		
g	grams	0.035	ounces	oz
kg	kilograms	2.202	pounds	lb
Mg (or "t")	megagrams (or "metric ton")	1.103	short tons (2,000 lb)	T
		TEMPERATURE (exact degrees)		
°C	Celsius	1.8C+32	Fahrenheit	°F
		ILLUMINATION		
lx	lux	0.0929	foot-candles	fc
cd/m^2	candela/m2	0.2919	foot-Lamberts	fl
		FORCE and PRESSURE or STRESS		
N	newtons	2.225	poundforce	lbf
kPa	kilopascals	0.145	poundforce per square inch	lbf/in^2

*SI is the symbol for International System of Units. Appropriate rounding should be made to comply with Section 4 of ASTM E380. (Revised March 2003)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

2D	two dimensional
3D	three dimensional
AADT	annual average daily traffic
ACC	adaptive cruise control
ADAS	advanced driver-assistance system
AdjV	adjacent vehicle
ADAP	ADS data acquisition and analytics platform
ADS	automated driving system
AMS	analysis, modeling, and simulation
API	application programming interface
BEV	bird's-eye view
CAN	controller area network
CAV	connected and automated vehicle
CC0	creative commons zero
CNN	convolution neural network
CP	check point
CSV	comma separated value
CV	connected vehicle
D-ADAS	discreet ADAS
DAQ	data acquisition system
DBW	drive by wire
DCP	data collection plan
DMP	data management plan
DS	data source
ECEF	Earth-centered, Earth-fixed
FHWA	Federal Highway Administration
FIR	finite impulse response
GNSS	global navigation satellite system
GPS	Global Positioning System
HD	high definition
ID	identification number
IMU	inertial measurement unit
IoI	interactions of interest
IOOs	infrastructure owners and operators
IoU	intersection of unit
IT	information technology
ITS	intelligent transportation system
ITS/JPO	Intelligent Transportation Systems Joint Program Office
LiDAR	light detection and ranging
MDMS	multidisciplinary data management system
MPL	multivariate piecewise linear
NDT	normal distribution transformation
NGSIM	Next-Generation Simulation
OC	operating condition

OEM	original equipment manufacturers
PCD	point cloud data
PII	personally identifiable information
PL	project lead
PSC	project subcontractor
RAID	redundant array of independent disks
RI-ADAS	readily identifiable ADAS
ROS	Robot Operating System
RTK	realtime kinematic
SAE	Society of Automotive Engineers
SFTP	secure file transfer protocol
SIEM	security information and event management
SSD	solid-state drive
SV	subject vehicle
USDOT	U.S. Department of Transportation
UTM	Universal Transverse Mercator
V2I	vehicle to infrastructure
V2V	vehicle to vehicle
V2X	vehicle to everything
WGS	world geodetic system

EXECUTIVE SUMMARY

As vehicle technology advancements accelerate, infrastructure owners and operators (IOOs) are making investment decisions today that will shape the transportation system of the future. Given the long design life of infrastructure, any changes to physical transportation infrastructure—such as adding lanes, installing intelligent transportation system (ITS) solutions (e.g., variable speed limits, ramp metering), and incorporating innovative intersection designs—will need to accommodate a wide variety of vehicle types, including SAE Level 0™ nonautomated vehicles, SAE Level 1™ and SAE Level 2™ ADAS-equipped vehicles, and SAE Level 3™, SAE Level 4™, and SAE Level 5™ ADS-equipped vehicles, even if these vehicle types are not widely available today (as in the case of the ADS-equipped vehicles).

To plan for these future vehicle types, State and local IOOs first need to capture how these various vehicle types behave in transportation analysis, modeling, and simulation (AMS) tools (e.g., microsimulation, four-step transportation planning model, dynamic traffic assignment, activity-based models, scenario-planning models, etc.). One current issue facing the developers and users of transportation AMS tools is a lack of high-resolution, naturalistic datasets that include ADAS- and ADS-equipped vehicle operation on roadways. OEMs, universities, and vehicle technology companies are beginning to release high-resolution, high-volume datasets of how their ADAS- and ADS-equipped vehicles behave in real-world environments. Unfortunately, these datasets are often raw sensor data (e.g., camera, light detection and ranging (LiDAR), radar) that are not suitable for traffic simulation model development unless processed into trajectories, which is not a trivial task. Thus, few to no large datasets exist that contain processed trajectories of ADAS- and ADS-equipped vehicles and surrounding nonautomated vehicles (i.e., 100-percent manually operated vehicles with no automated assistance features (SAE Level 0)) in naturalistic traffic conditions.[1]

In this project, the research team developed a methodology to collect and process ADAS-equipped vehicles' raw onboard sensor data to extract data about the instrumented ADAS-equipped subject vehicle and all adjacent vehicles in the traffic stream that were perceived by the on-board sensors. These publicly available datasets, available on the ITS DataHub, will enable researchers to develop models that characterize human-ADAS interactions under a diverse set of naturalistic traffic scenarios in highway and arterial environments.[2,3] Using these data, researchers can develop appropriate models to assess the impact of ADAS technologies and update current driver-behavior models. Results from models developed using these datasets will enable researchers, analysts, and IOOs to better understand how ADAS-equipped vehicles will affect transportation system performance. Ultimately, the models developed using these data can support data-informed decisionmaking that maximizes ADAS benefits to the transportation system performance.

CHAPTER 1. INTRODUCTION

BACKGROUND

Analysis, modeling, and simulation (AMS) tools are critically important to infrastructure owners and operators (IOOs) (e.g., State and local departments of transportation). In fact, the Fixing America's Surface Transportation (FAST) Act placed increasing emphasis on the application of AMS tools in the planning process. Section 1430 of the act explicitly states:

> The Department should utilize, to the fullest and most economically feasible extent practicable, modeling and simulation technology to analyze highway and public transportation projects authorized by this Act to ensure that these projects—
>
> (1) will increase transportation capacity and safety, alleviate congestion, and reduce travel time and environmental impacts; and
>
> (2) are as cost effective as practicable. (p. 117).[4]

AMS tools are valuable because they help IOOs understand the impact of changes to the transportation system (e.g., increases in demand, installation of intelligent transportation system (ITS) technology, new high-occupancy toll or general-purpose lanes, new signal timing plans) on system performance (e.g., capacity, speed, travel time). These tools help IOOs to better operate their existing system and to better plan their transportation system of tomorrow.

However, the users of transportation systems are rapidly changing in ways that may have a significant impact on the transportation system's performance. Specifically, the technology becoming available on vehicles traversing our Nation's roadways may impact how drivers behave. Many OEMs are now offering SAE Level 1™ advanced driver-assistance system (ADAS) technology packages as standard on new vehicles purchased by consumers.[1] Common ADAS features include adaptive cruise control (ACC), lane-keeping assistance, and blind-spot monitoring. As of May 2018, at least one ADAS feature is available on 92.7 percent of new vehicles available in the United States.[5] Additionally, many OEMs offer technology upgrade packages on their ADAS-equipped vehicles that make these vehicles capable of SAE Level 2™ partial driving automation (e.g., simultaneous lane centering and ACC).[1] ADAS-equipped vehicles are rapidly gaining market penetration in the United States and abroad.

At the time of this report's publication, no vehicles sold commercially in the U.S. market are equipped with a technology package that enables SAE Level 3™, SAE Level 4™, or SAE Level 5™ automation, which are identified as automated driving system (ADS)-equipped vehicles.[1] However, connected and automated vehicles (CAVs), which include connected and unconnected ADAS-equipped vehicles, connected and unconnected ADS-equipped vehicles, and connected vehicles (CVs), are driving multi-billion-dollar investments from OEMs and technology companies. While the exact timeframe and capabilities of these systems are a significant source of uncertainty, the transportation industry generally agrees that CAV adoption will occur over the next several decades.

As vehicle technology advancements continue and accelerate, IOOs are making investment decisions today that will shape the transportation system of the future. Given the long design life of infrastructure, any changes to physical transportation infrastructure (e.g., adding lanes, allowing hard shoulder use, incorporating innovative intersection designs) will need to accommodate the needs of a wide variety of vehicle types, including SAE Level 0™ human-driven vehicles (referred to as nonautomated vehicles in this report), SAE Level 1 and 2 ADAS-equipped vehicles, and SAE Levels 3–5 ADS-equipped vehicles, even if these vehicle types do not exist today, as in the case of the ADS-equipped vehicles.(1)

Thus, robust AMS tools are valuable to help IOOs understand the likely impacts of CAVs on their transportation system performance, to make better investment decisions now to ensure their transportation system is equipped for CAVs in the future, and to optimally operate their roadways once CAVs are deployed. Ultimately, having a transportation system that is better designed and equipped to accommodate CAVs will benefit all transportation system users.

To plan for these future vehicle types, State and local IOOs need to capture how these various vehicle types behave in transportation AMS tools, including, but not limited to, microsimulation, four-step transportation planning models, dynamic traffic assignment, activity-based models, scenario-planning models. In the cornerstone U.S. Department of Transportation (USDOT) publication, *Development of an Analysis/Modeling/Simulation Framework for Vehicle-to-Infrastructure (V2I) and Connected/Automated Vehicle Environment*, Mahmassani et al. detail a comprehensive methodological framework for developing CAV AMS tools.(6) As part of this previous research effort, the team identified existing gaps preventing the community from developing, calibrating, and validating tools to model CAVs and human behavior in the vicinity of CAVs. The research team identified one of the most significant gaps to be the availability of high-resolution, naturalistic CAV datasets that include connected and unconnected ADAS-equipped vehicles, connected and unconnected ADS-equipped vehicles, and CVs.

OEMs, universities, and vehicle-technology companies are beginning to release high-resolution, high-volume datasets of how their ADAS- and ADS-equipped vehicles behave in real-world environments. Unfortunately, these datasets are often raw on-board sensor data (e.g., camera, light detection and ranging (LiDAR), radar) and are not suitable for traffic simulation model development unless processed into trajectories, which is not a trivial task. Thus, few to no large datasets exist that contain processed trajectories of ADAS- and ADS-equipped vehicles and surrounding nonautomated vehicles (i.e., 100 percent manually operated vehicles with no automated assistance features (SAE Level 0)) in naturalistic traffic conditions.(1)

The lack of naturalistic datasets for CAV operation hinders IOOs from preparing and creating solutions for future roadway problems, such as traffic congestion induced by differences in CAV driving behavior. To appropriately characterize the behavior of CAVs and nonautomated vehicle responses to CAVs in naturalistic traffic environments, researchers need to collect data that includes following types:

- Vehicle-following and lane-changing behavior for a variety of CAV systems at different SAE automation and cooperation levels on different functional classifications of roadways.(7)

- CAV operation at intersections, including gap acceptance, varying intersection control types, and V2I communication.

- CAV behavior operation in the presence of bicycles, pedestrians, and vulnerable road users.

This project seeks to help fill the gap of naturalistic datasets about ADAS-equipped vehicle behavior in naturalistic traffic on both highways and arterials.

PROJECT OBJECTIVE

The objective of this project was to collect a large dataset of how conspicuous and inconspicuous ADAS-equipped vehicles navigating complex driving environments and how nonautomated vehicles in the traffic stream interact with both readily identifiable (RI)- and discreet (D)-ADAS-equipped vehicles. Specifically, this project collected 144 h of baseline (nonautomated vehicle), D-ADAS, and RI-ADAS driving data in central Ohio in a variety of different complex driving environments, including arterials and highways. The difference between these two classes of ADAS-equipped vehicles is a clearly visible sensor stack that the traveling public would associate with an automated vehicle. Figure 1 is a picture of an ADAS-equipped vehicle with a visible sensor stack, an RI-ADAS. Figure 2 is an ADAS-equipped vehicle without a visible sensor stack, a D-ADAS (a production ADAS-equipped vehicle commercially available to the public). The use of the D-ADAS- and the RI-ADAS-equipped vehicles will enable the research team to collect data about adjacent vehicles (AdjVs) in traffic that determine if nonautomated vehicles are altering their driving behavior because of differences in the ADAS-equipped vehicle's appearance or differences in the ADAS-equipped vehicle's driving behavior (e.g., gap distance, speed, perception reaction time, etc.).

Figure 1. Photo. RI-ADAS.

Figure 2. Photo. D-ADAS.

One limitation of this dataset is that all vehicles were operated by members of the project team who had familiarity with the available driver-assistance technology and not by recruited drivers who may or may not be familiar with the technology. Thus, researchers should not use this dataset to assess how different drivers interact with the technology inside of the SVs. For example, these data are not appropriate for assessing how ADAS drivers select their ACC following gap because research team members (and not the general public) were operating the ADAS vehicles. However, the data can help evaluate how the selection of the ADAS' ACC gap impacts another driver's willingness to change lanes in front of the ADAS vehicle. The focus of these datasets is how drivers in AdjVs interact with the technology on the subject vehicle (SV) and how these interactions impact traffic flow.

Researchers can use this study's data to develop new and update existing driver-behavior models that can help assess the impact of ADAS. The diversity of vehicles, sensor configuration, and driving environment designed into the data collection effort will enable researchers to answer a number of research questions including, but not limited to, the following:

- Are adjacent drivers altering their driving behavior (e.g., following distance, gap acceptance) when interacting with ADAS-equipped vehicles compared to their baseline behavior interacting with other manually driven vehicles in traffic? An adjacent driver is defined as a driver in a nonautomated vehicle in either the same or an adjacent lane whose behavior is passively captured through the instrumented vehicle's sensor stack. Examples of adjacent drivers include the vehicle leading and following the SV and vehicles that are passing (or being passed by) the SV in an adjacent lane.

- Are the behavioral changes, if any, of AdjVs due to the behavior changes associated with the ADAS-equipped vehicle (e.g., more consistent following distance, larger headways), or are drivers altering their behavior due to the appearance of the ADAS-equipped vehicle (e.g., visibility of sensor suite)?

- How does single ADAS-equipped vehicle operation impact traffic flow compared with a string of two ADAS-equipped vehicles?

- What is the impact of driving environment (e.g., freeway versus arterial; dry roads versus wet roads) on ADAS performance?

Results from models developed using these datasets should enable researchers to better understand how ADAS-equipped vehicles will affect transportation system performance. Ultimately, the models developed using these data can support data-informed decisionmaking that maximizes ADAS benefits to system performance.

REPORT TERMINOLOGY

The following terms have been defined for this report:

- CAV: Includes connected and unconnected ADAS-equipped vehicles, connected and unconnected ADS-equipped vehicles, and CVs. CAV is used broadly to speak about the impacts of various levels of automation and communication classes. However, subsequent chapters of this report are specific to the type of vehicle used to collect data.

- ADAS-equipped vehicle: Has SAE Level 1 (driver assistance) or SAE Level 2 (partial driver automation) capabilities. All data collected in this project were from an ADAS-equipped vehicle with SAE Level 2 capabilities (except the baseline data).[(1)]

- ADS-equipped vehicle: Has SAE Level 3 (conditional driving automation), SAE Level 4 (high driving automation), or SAE Level 5 (full driving automation) capabilities.[(1)]

- Nonautomated vehicle: Indicates an SAE Level 0 vehicle with no automation support capability that was manually driven by a human.[(1)]

- D-ADAS-equipped vehicle: Indicates a commercially available ADAS-equipped vehicle with no visible sensors.

- RI-ADAS-equipped vehicle: Indicates an ADAS-equipped vehicle with visible sensors that may visually indicate to adjacent drivers that something is different about this vehicle. The RI-ADAS-equipped vehicles in this study had SAE Level 2 automation support, but the visible sensor stack may have caused some adjacent drivers to assume the vehicles were capable of higher levels of automation.[(1)]

- Subject vehicles SV1 and SV2: Refers to the instrumented vehicles used to collect data. This project collected both single-vehicle and two-vehicle datasets. For the single-vehicle datasets, both D-ADAS- and RI-ADAS-equipped vehicles were operated using their SAE Level 2 partial driving automation capabilities to collect data. For the two-vehicle datasets, SV1 was operated as an ADAS-equipped vehicle with partial driving automation capabilities congruent with SAE Level 2 partial driving automation capability.[(1)] Due to the limitation of map data, SV2 was operated in traffic without its automated driving capabilities. When multiple SVs were on the road, these vehicles were operated independently and did not share information via vehicle-to-vehicle (V2V) communications.[(1)]

- Adjacent vehicle (AdjV): Refers to SAE Level 0 vehicles not instrumented or operated by a project team member that were interacting with the SVs (e.g., vehicles that were in the left or right lane beside the SV, following the SV, or leading the SV). The project team was interested in AdjV interactions, and AdjV behavior was collected passively through the sensors (e.g., LiDAR, camera) available on the SVs. The AdjV trajectories (e.g., time series position, speed, and acceleration) were extracted by processing the SV's sensor data using methods described in chapter 3.

Chapter 2 provides additional details about the RI-ADAS and D-ADAS-equipped vehicles in this report.

REPORT ORGANIZATION

The remainder of this report contains details about data collection, processing, and management throughout the lifecycle of the project. The remaining chapters are organized as follows:

- Chapter 2. Data Collection Plan: This chapter summarizes the methodology adopted for the collection of data and describes the vehicles' modes of operation, route choice, scenarios of interest, and driving environments. In addition, the authors discuss the vehicle's sensors, information collected, and software stack used for data collection.

- Chapter 3. Data Processing Method: This chapter describes the methodology for converting the raw vehicle data into processed trajectory data. Additionally, this chapter provides details about how the data were validated.

- Chapter 4. Data Management Plan: This chapter outlines the flow of the data after data acquisition was completed and describes the types of storage (e.g., local storage, cloud-based storage), authorization required for accessing these storage locations, and data flow to and from various team members.

- Chapter 5. Key Conclusions from the Project: This section describes the key findings from the data collection project.

CHAPTER 2. DATA COLLECTION PLAN

The team developed a data collection plan (DCP) that outlined the technical and management methodology adopted to collect naturalistic ADAS-equipped vehicle datasets in central Ohio. The DCP includes the following:

- The data collected throughout the project.
- The locations where data were collected.
- The data collection methods.
- The times and dates when data collection occurred.

The subsections in this chapter document the project team's decisions regarding vehicles used in data collection, scenarios of interest for data collection, and data collection location to ensure the team collected a diverse dataset of ADAS-equipped vehicle interactions with nonautomated vehicles in naturalistic traffic conditions.

DATA COLLECTION VEHICLES

For this project, the team used two different types of instrumented SVs for data collection: D-ADAS- and RI-ADAS-equipped vehicles. The difference between the two types was the use of a visible sensor stack that implies a different level of technology than on a production vehicle available for purchase. All vehicles were operated by members of the project team who had familiarity with the driver-assistance technology available on the vehicles. The focus of these datasets is how drivers in adjacent, nonautomated vehicles interact with the technology on the SV, not how the SVs' driver interacts with the technology.

To help isolate adjacent nonautomated driver behavior in the presence of advanced-looking ADAS-equipped vehicles, the project team utilized RI-ADAS- and D-ADAS-equipped vehicles to enable researchers to study how nonautomated vehicle driver perception of the technology impacts their interactions. To help further isolate the difference between RI-ADAS- and D-ADAS-equipped vehicles, the project team utilized the same vehicle for some of the testing to act as a RI-ADAS for some test days and a D-ADAS for the remainder. Using the same vehicle helped the project team capture the differences in adjacent nonautomated vehicle driver behavior due to the visible sensor stack versus the differences in adjacent nonautomated vehicle driver behavior due to the behavior of the ADAS (e.g., more consistent driving behavior, headways, etc.).

To establish baseline driving behavior, the data collection team operated both ADAS-equipped vehicles in manual mode (without driver-assistance technology enabled) to collect a small sample of data that established how central Ohio drivers interact with another nonautomated vehicle. The mode of operation according to type of vehicle is recorded in the "type_of_vehicle" column as RI, DI, or baseline. By comparing the baseline dataset to the datasets with the same vehicles operated with automated driving functionalities engaged, researchers can explore the changes in driver behavior and gain further insights into the influence of the ADAS-equipped vehicles' behavior on adjacent nonautomated vehicle driver behavior. The baseline data consisted of 8 h of manually driven data.

Additional details about the D-ADAS and RI-ADAS-equipped vehicles, including the addition of the sensor stack, are provided in the following sections.

RI-ADAS

The team used two different research vehicle makes and models with SAE Level 2 partial driving automation capabilities as RI-ADAS-equipped vehicles during this project. Drivers of SV1 used its OEM ACC, lane-centering assistance, lane-change assistance, traffic and stop sign control, and navigate-on-autopilot features. Drivers of SV2 (from a different manufacturer) operated the vehicle with SAE Level 0 automation (no ADAS support) due to limitations with the map data.[1]

The RI-ADAS-equipped vehicles were outfitted with several visible sensors, including radar and cameras, to collect additional data for monitoring and understanding the vehicles' dynamic driving environment. The team performed numerous controlled-environment runs before the data collection phase of the project and before deployment on public roads to validate the sensor data's frequency, dropouts, and transmission consistency. The sensor validation was completed through the data acquisition system's (DAQ) user interface for both vehicles. Due to the controlled environment containing data for other confidential users, these data were not retained once the process was validated. Table 1 denotes the sensors that were on each RI-ADAS.

Table 1. Summary of RI-ADAS data variables and sources.

Data Variable	Source	Availability
SV position	GPS/GNSS/IMU unit	Raw dataset
SV speed	GPS/GNSS/IMU unit	Raw dataset
SV acceleration	GPS/GNSS/IMU unit	Raw dataset
SV position	Output of LiDAR postprocessing	Public dataset[2,3]
SV speed	Output of LiDAR postprocessing	Public dataset[2,3]
SV acceleration	Output of LiDAR postprocessing	Public dataset[2,3]
Subject steering wheel angle	Vehicle DBW or CAN recorded	Raw dataset
Subject accelerator pedal position	Vehicle DBW or CAN recorded	Raw dataset
Subject brake pedal position	Vehicle DBW or CAN recorded	Raw dataset
Subject driver/ADAS transition points	ADAS or video recorded	Raw dataset
Subject system settings	Driver recorded	Raw dataset
SV attributes	Driver recorded	Raw dataset
SV LiDAR raw data	Raw sensor output in rosbags[8]	Raw dataset
SV frontal video	Commercial camera	Raw dataset
SV rear video	Commercial camera	Raw dataset
SV driver video	Commercial camera	Raw dataset
AdjVs(s) position	LiDAR postprocessing output	Public dataset[2,3]

Data Variable	Source	Availability
AdjVs(s) speed	LiDAR postprocessing output	Public dataset[2.3]
AdjVs(s) acceleration	LiDAR postprocessing output	Public dataset[2.3]
AdjVs(s) IDS	LiDAR postprocessing output	Public dataset[2.3]
Relative SV speed	LiDAR postprocessing output	Public dataset[2.3]
Relative SV position	LiDAR postprocessing output	Public dataset[2.3]

CAN = controller area network; DBW = drive by wire; GNSS = global navigation satellite system; GPS = Global Positioning System; ID = identification number; IMU = inertial measurement unit.
Note: rosbag is the file format for the Robot Operating System (ROS).[8]
Note: The raw dataset contains PII and is not publicly available.

Both vehicles had three cameras to monitor various portions of the driving scene:

- Front-facing camera to capture the area immediately in front of the RI-ADAS.
- In-cabin camera to capture the SV cabin environment.
- Rear-facing camera to capture the area immediately behind the RI-ADAS.

The project team used a commercial camera capable of producing a video resolution of 720 pixels at a frame rate of 30 Hz for external-facing cameras and 10 Hz for internal-facing cameras. These sensors streamed data into the DAQ in the RI-ADAS for storage. The DAQ's software package enabled synchronization of the data streams from the three on-board cameras with the other sensor data.[9]

D-ADAS

The team also used one production D-ADAS-equipped vehicle to collect vehicle data. This vehicle has partial driving automation capabilities congruent with SAE Level 2 automation as detailed in SAE J3016™ Levels of Driving Automation™.[1] Drivers used the OEM ACC, lane-centering assistance, lane-change assistance, traffic and stop sign control, and automated navigation features.

The purpose of driving the D-ADAS-equipped vehicle was to collect data about how drivers in AdjVs interact in traffic with inconspicuous vehicles while not realizing the SV is equipped with advanced technologies.

Because the D-ADAS is a production vehicle and not a research vehicle, the project team installed additional aftermarket sensors to enable the collection of trajectories for AdjVs in traffic. These additional sensors are discreet in appearance, and almost all are located within the vehicle. One notable exception is that the team placed a LiDAR unit in a cargo roof rack to collect AdjV trajectories (figure 3). A summary of data sources and variables for the D-ADAS is shown in table 2.

Source: Federal Highway Administration (FHWA).

Figure 3. Photo. D-ADAS with LiDAR installed on top of vehicle hidden discreetly to AdjV drivers by moving boxes.

Table 2. Summary of D-ADAS data variables and sources.

Data Variable	Source	Availability
SV position	GPS/GNSS/IMU unit	Raw dataset
SV speed	GPS/GNSS/IMU unit	Raw dataset
SV acceleration	GPS/GNSS/IMU unit	Raw dataset
SV position	LiDAR postprocessing output	Public dataset(2)
SV speed	LiDAR postprocessing output	Public dataset(2)
SV acceleration	LiDAR postprocessing output	Public dataset(2)
Subject steering wheel angle	Vehicle CAN	Raw dataset
Subject accelerator pedal position	Vehicle CAN	Raw dataset
Subject brake pedal position	Vehicle CAN	Raw dataset
Subject driver/ADAS transition points	Video recorded	Raw dataset
Subject system settings	Driver recorded	Raw dataset
SV attributes	Driver recorded	Raw dataset
SV frontal video	Commercial camera	Raw dataset
SV rear video	Commercial camera	Raw dataset
SV driver video	Commercial camera	Raw dataset
AdjVs(s) position	LiDAR postprocessing output	Public dataset(2)
AdjVs(s) speed	LiDAR postprocessing output	Public dataset(2)
AdjVs(s) acceleration	LiDAR postprocessing output	Public dataset(2)
AdjVs(s) IDs	LiDAR postprocessing output	Public dataset(2)
Relative SV speed	LiDAR postprocessing output	Public dataset(2)
Relative SV position	LiDAR postprocessing output	Public dataset(2)

Figure 3 shows the installed LiDAR sensor atop packing boxes on a commercially available luggage rack. The boxes mask the installation of the LiDAR sensor so as not to affect AdjV driver's behavior.

DATA COLLECTION SCENARIOS

This section discusses the scenarios of interest and driving environments in detail:

- Scenarios of interest: Includes intentional interactions between the instrumented SV and AdjVs in traffic (described in more detail in the Scenarios of Interest subsection). The project team targeted three types of scenarios of interest: vehicle following, lane change, and intersection approach and departure.

- Driving environment: Comprises two roadway functional classification and three levels of congestion experienced during data collection (described in more detail in the Driving Environments subsection):

    - Roadway functional classifications: Arterial, freeway.
    - Level of congestion: Light, moderate, heavy.

Scenarios of Interest

The project team identified three generalized test scenarios as the most valuable for the intended research goals of this project. The three scenarios were collected with both RI-ADAS and D-ADAS vehicles and during both single-vehicle and two-vehicle data collection. The three scenarios of interest are:

- Vehicle following on freeways and arterial roadways.
- Lane change on freeways and arterial roadways.
- Intersection approach and departure at signalized intersections.

Table 3 shows the parameters used for data collection on roadway.

Table 3. General scenario information.

Variables	Notes
SV speed	>20 mph*
Weather	Clear conditions or light rain
Road conditions	Dry or wet roadway
Lane information	Two or more lanes (total); bidirectional travel or divided highway
Average traffic density	Light, moderate, heavy

*ACC and lane-centering assistance can only be engaged above 20 mph, but data was still collected, and automation stayed engaged if traffic brought the vehicle to a lower speed.

Vehicle Following

The first generalized scenario of interest focused on ADAS-equipped vehicle operation within a specific lane on divided freeways and high-capacity arterial roads. The different runs of the vehicles following this scenario of interest included baseline, single ADAS-equipped vehicle operations, and two-vehicle operations:

- Baseline: In the baseline scenario (SV manually operated as an SAE Level 0 vehicle), the safety driver controlled the vehicle's longitudinal speed and steering.

- ADAS single-vehicle data collection: In both the D- and RI-ADAS-equipped vehicles, the safety driver utilized ACC with lane-centering assistance to control SV1's speed and location within the lane. The driver monitored the vehicle's actions to ensure they were appropriate and was ready to take over if anything deviated from an expected action.

- ADAS two-vehicle data collection: The safety driver in SV1 utilized ACC with lane-centering assistance to control the vehicle's speed and location within the lane. The driver monitored the actions of the vehicle to ensure they were appropriate and was ready to take over vehicle operation if anything deviated from an expected action. While SV2 was also RI-ADAS equipped, the driver did not use automation and instead mimicked the behavior of SV1.

This generalized testing scenario involved an instrumented SV operating within a single lane. The team collected data during different levels of congestion and, for each run, SV1's aggressiveness and following distance setting was set to be the same. Within each scenario of interest, the team hoped to produce specific interactions of interest (IoIs). The IoIs for the vehicle-following scenario involved AdjVs interactions with the SV(s), including:

- Human driver in adjacent nonautomated vehicle overtakes SV(s).
- Human driver in adjacent nonautomated vehicle cuts in ahead of SV(s).
- Human driver in adjacent nonautomated vehicle slows down behind SV(s).

For a vehicle-following scenario, the safety driver executed the following steps in sequential order:

1. The driver positioned the vehicle in the center of a lane of the public road route from a specified starting location and direction of travel.

2. The driver followed the specified route to the target destination for baseline runs. For D-ADAS and RI-ADAS single-vehicle runs using SV1, the driver engaged ACC and lane-centering assistance to follow the route to target destination with driver supervision. In the other cases (baseline and SV2 operation during the two-vehicle data collection runs), the driver was in full control.

3. The driver approached a lead vehicle in the specified lane and maintained a constant distance using ACC.

4. Each trial ended when the SV successfully traversed the route. Once reaching the desired destination, the driver disengaged the ACC feature.

Figure 4 is an example vehicle-following scenario for single-vehicle operation.

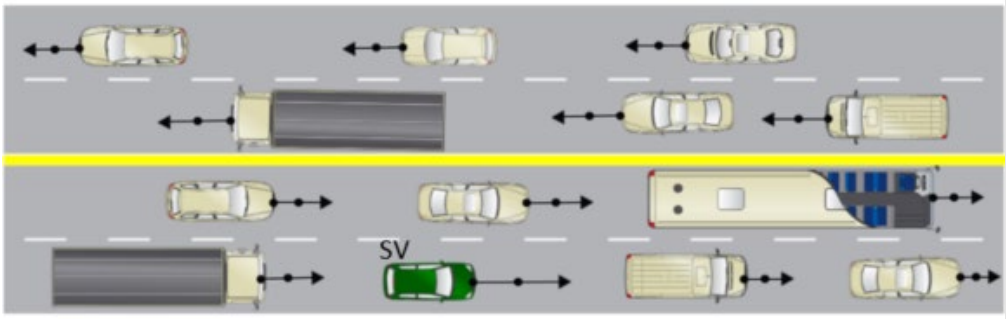


Figure 4. Illustration. Example of a vehicle-following scenario of interest.[10]

Lane Change

The second scenario was lane changing on a highway or divided arterial roadway during data collection with a single ADAS-equipped vehicle and two ADAS-equipped vehicles. The IoI of this testing scenario was to collect data for SV overtaking of nonautomated AdjVs. However, the team also documented lane changes in scenarios when the SV was not necessarily overtaking another vehicle, such as interstate exit and lane closures at work zones. If the AdjV was in proximity, the lane-change assist feature on SV1 did not permit the lane-change maneuver. In such cases, the lane change was initiated prior to approaching the AdjV. The role of the ADAS-equipped vehicle and the role of the safety driver varied between the baseline scenario, the single-vehicle data collection, and the two-vehicle data collection:

- Baseline: In the baseline scenario (SV manually operated as an SAE Level 0 vehicle), the safety driver controlled the longitudinal speed and steering of the vehicle.

- ADAS single-vehicle data collection: In the D- and RI-ADAS-equipped vehicles, the driver used ACC with lane-centering assistance to control the speed and position of the vehicle. Additionally, the driver used lane-change assist by engaging the turn signal, which initiated the ADAS-equipped vehicle to complete the lane change. When the lane-change assist feature was activated, the safety driver monitored the vehicle's actions to ensure they were appropriate and was ready to take over vehicle operation if anything deviated from an expected action. Team members collected data during different levels of congestion and included a focus on when the lane-change request occurred in relationship to other vehicles.

- ADAS two-vehicle data collection (both vehicles RI-ADAS): In the two-vehicle data collection, the safety driver operated SV1 identically to the single-vehicle data collection (i.e., the vehicle performed the lane-change maneuver after the driver activated the lane-change assist feature). For SV2, the driver performed the lane change by mimicking the behavior of SV1.

During baseline data collection and for SV2 operation during the two-vehicle data collection runs, the safety driver maintained manual control of the vehicle at all times and executed the following steps in sequential order:

1. The driver positioned the vehicle in the center of a lane of the public road route from a specified starting location and direction of travel. The driver controlled the longitudinal speed of the vehicle.

2. The driver maintained a safe separation distance from other vehicles and performed a lane change as needed until the vehicle reached its final destination.

During the single-vehicle data collection runs and for SV1 operation during the two-vehicle data collection runs, the safety driver engaged ACC, lane centering, and lane-change assist. For a lane-changing scenario, the safety driver executed the following steps in sequential order:

1. The safety driver positioned the vehicle in the center of a lane of the public road route from a specified starting location and direction of travel. The safety driver adjusted the set speed of the SV ahead of the speed of an AdjV to encourage a desired interaction.

2. Using ACC and lane-centering assistance features, the vehicle maintained a desired safe distance from other vehicles in its lane. The safety driver requested a lane change using the turn signal. The lane-change assist feature began the lane change once requested by the driver and after the vehicle determined that completing the maneuver was safe.

3. The safety driver monitored the vehicle as it followed the specified route to the given target destination.

4. At the end of the route, the safety operator disengaged the ADAS features.

5. Each scenario ended when the SV successfully changed lanes while merging with the traffic in the target lane or when the safety driver intervened.

Figure 5 shows an example test scenario for a single-vehicle lane change on a divided highway. The rate of lane change was adjustable. SV1 determined the presence of AdjVs and made the lane change when the action was safe.

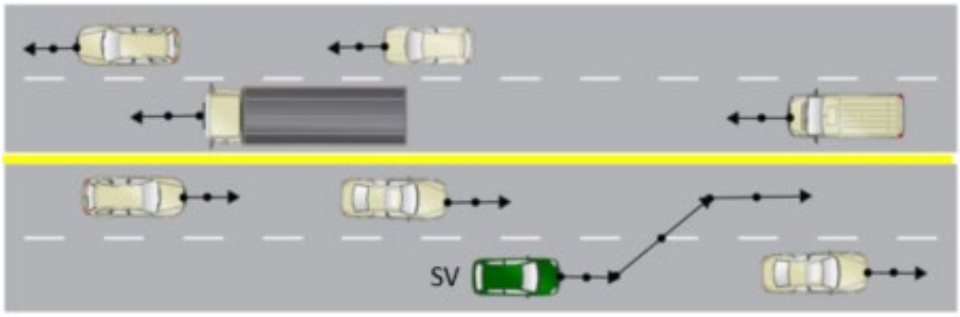


Figure 5. Image. Example of a lane-change scenario of interest.[10]

Intersection Approach and Departure

The last generalized test scenario of interest was intersection approach and departure. The role of the ADAS-equipped vehicle and the role of the safety driver varied between the baseline scenario, the single vehicle data collection, and the two-vehicle data collection:

- Baseline: In the baseline scenario (when the SV was manually operated as an SAE Level 0 vehicle), the safety driver controlled the longitudinal speed and lateral movement of the vehicle.[1]

- ADAS Single Vehicle Data Collection: In the D- and RI-ADAS-equipped vehicle, the safety driver utilized ACC with lane centering assistance to control the speed of the vehicle. This vehicle had limited traffic light and stop sign detection that was used to slow the vehicle to a stop. Human intervention was necessary for reengaging the vehicle when the light changed to green from red.

- ADAS Two Vehicle Data Collection: Both vehicles were RI-ADAS. In the two-vehicle data collection, SV1 was operated identically to how the vehicle was operated in the single vehicle data collection (i.e., ACC with lane centering assistance and traffic light/stop sign detection). SV2 was driven manually by the safety driver, following the same behavior as SV1.

Automated driver behavior at intersections is still an evolving area, so instead of specific IoIs, the intent here was to better understand ADAS operations at intersections. Figure 6 shows an example scenario for this generalized test.

Figure 6. Illustration. Example of four-way intersection navigation.[10]

For the intersection scenario using the RI- and D-ADAS-equipped vehicles, the safety driver executed the following steps in sequential order:

1. The safety driver positioned the SV in the center of the rightmost lane of the public road route approaching an intersection from a specified starting location and direction of travel.

2. For SV1, the driver engaged ACC, lane-centering assistance, and traffic light and stop sign control behind a lead vehicle so that the SV would remain in the correct path of travel as it traversed the route until it exited the intersection.

3. Each scenario ended when the SV successfully maneuvered the intersection or when the safety driver intervened.

4. At the end of the route, the safety operator disengaged the automated features.

Unfortunately, the research team was unable to collect CV data exchanging messages via V2V or V2I communications.

Driving Environments

The team selected candidate data collection sites based on the testing criteria needed for each generalized testing scenario. The vehicle-following and lane-change scenarios were executed on highway and arterial roadways that experience low, moderate, and high traffic density that vary by time and date. Drivers executed the intersection approach and departure scenario at signalized intersections.

To ensure a diverse set of data were collected, the team targeted a variety of roadway types that were known to produce low, medium, and high annual average daily traffic (AADT) as well as low, medium, and high speeds.

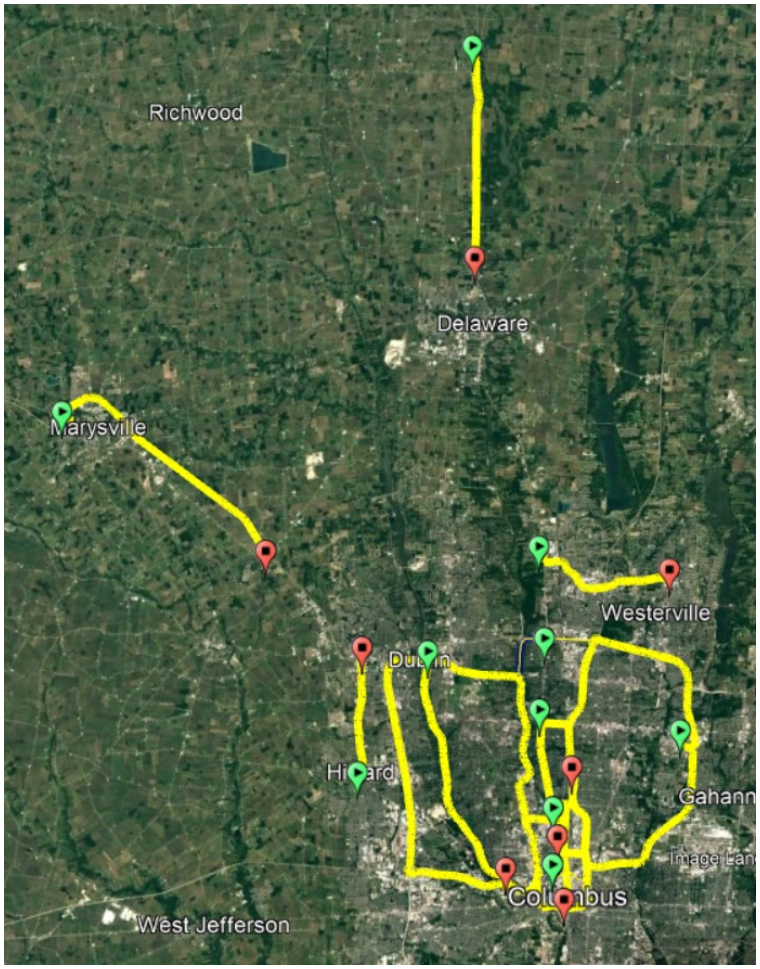
AADT is the total yearly volume of vehicles on a road divided by 365 d and was used to help the team identify data collection sites with a variety of traffic loads. AADT can be classified into three categories: high (AADT >50,000 vehicles/d), medium (400 vehicles/d ≤ AADT ≤ 50,000 vehicles/d) and low (AADT <400 vehicles/d) density.[11,12] Approximately 60 percent of the data were collected on high AADT roads and the remaining 40 percent on medium AADT roads.[13] No data were collected on low AADT roadways to avoid scenarios with insufficient vehicles with which to interact. Localized congestion level immediately surrounding the SV was not measured based on the SV's video data.

Roadway types were categorized as limited access freeway, nondivided arterial road, and divided arterial road. Limited access freeway route type includes U.S. and State numbered freeways and expressways and interstate routes where access to and from the facility is limited to interchanges with grade separations. Opposing directions of travel are separated by a median on these high-speed routes, which typically have posted speed limits ranging from 88.5 km/h (55 mph) in urban areas to 112.7 km/h (70 mph) for the roads selected. An arterial street through a predominately residential area primarily caters to through traffic. Posted speed limits generally range from 40.2 km/h (25 mph) to 72.4 km/h (45 mph). The pavement widths permit full-time operation of bidirectional traffic.[14] For this project, the team collected 40 percent of the data on limited access freeways, 35 percent on nondivided arterial roads, and 25 percent on divided arterial roads.

By selecting different roadway types that experience a wide range of AADT, the project team was able to collect data at a wide range of travel speeds: low (<62.8 km/h (39 mph)), medium (62.8 km/h (39 mph) to <96.5 km/h (60 mph)) and high (≥96.5 km/h) 60 mph)). For the scope of this project, the goal was to collect 20 percent low-speed data, and the remaining data divided into medium speed and high speed. Low-speed data were purposely minimized because, during the initial runs, this category produced the fewest IoIs.

Based on these considerations, the research team chose eight different routes in central Ohio for data collection (see figure 7):

- U.S. 33 from 1501 West 5th Street, Marysville, OH, to 10152 U.S. 42, Marysville, OH.
- U.S. 33 from 4555 West Granville Road, Dublin, OH, to 1093 Dublin Road, Columbus, OH.
- I–270 from 700 East North Broadway, Columbus, OH, to 4024 Morse Road, Columbus, OH.
- U.S. 23 from 2381 U.S. Hwy 23 North, Delaware, OH, to 262 North Marion Street, Waldo, OH.
- U.S. 315 from 4555 West Dublin Granville Road, Dublin, OH, to 1090 Dublin Road, Columbus, OH.
- I–750 from 8870 Columbus Pike, Lewis Center, OH, to 3760 Main Street, Hilliard, OH.
- U.S. 33 from 2500 Summit Street, Columbus, OH, to 3760 Main Street, Hilliard, OH.
- U.S. 23 from 2500 Summit Street, Columbus, OH, to 2520 Summit Street, Columbus, OH.



Original map: © 2022 Google® Earth™. Modifications by FHWA (see Acknowledgments section).

Figure 7. Map. Routes for deployments.[15]

The data-type details for a single day's runs were recorded in a run-summary file specific to each day. The run summary also contains the information regarding route start and end, Google™ Map route, number of passes made by the ADAS vehicle and other AdjVs(s), testing scenario type, and any other interesting scenarios.[15] Table 4 and table 5 show a sample run-summary file containing information for two runs. Table 4 shows the route and weather information, while table 5 shows the run-specific information. These summary files were converted into additional columns (table 10 and table 15) to make this rich metadata available for each instance of the collected data.

Table 4. Route information sample file.

Route Starting Point	Route Ending Point	Distance	Map (Route Start to Route End) North Loop (RS→RE)	Map (Route End to Route Start) South Loop (RE→RS)	AADT	Roadway Type	Route Speed (mph)	Temp. (°F)	Humidity (%)	Precip. (inches)	Road Condition
700 E N Broadway, Columbus, OH 43214	Parking lot, New Bond St, Columbus, OH 43219	13 mi	https://goo.gl/maps/QDf5zb3BuydkaFpSA	https://goo.gl/maps/fdS6V3J8RaCL8LAe9	1,081	Limited access	65	High: 75.7; Low: 45.3; Avg: 59.2	High: 99; Low: 38; Avg: 72	0	Dry

RS = route start; RE = route end; Temp. = temperature; Precip. = precipitation.

Table 5. Run information sample file.

Run Number	Direction	Run Type (Baseline/ RI/D)	Dewesoft/ Video Filenames	ROS LiDAR Filename	Processed Data	No. of Passes of Other Vehicles Made by Vehicle	No. of Passes Made by Other Vehicle	Did You Follow a Car? (Y/N)	Did You Go Through an Intersection? (Y/N)	Any Other Scenarios of Interest (Y/N)
1	North loop clockwise	RI	{Vehicle}_CAV_2021_11_18_102352.dxd	data_cav_2021-11-18-10-23-37.bag	{Vehicle}_CAV_2021-11-18-10-23-37_115_135_processed.csv, Tesla_CAV_2021-11-18-10-23-37_260_145_processed.csv, Tesla_CAV_2021-11-18-10-23-37_410_120_processed.csv, Tesla_CAV_2021-11-18-10-23-37_530_120_processed.csv, Tesla_CAV_2021-11-18-10-23-37_655_140_processed.csv	Not recorded due to heavy traffic	Not recorded due to heavy traffic	N	N	N

Y= yes; N= no.

DATA COLLECTION SCHEDULE

The project team collected data over two different deployments—from September through November 2021 and from January through March 2022. The deployments had a monthlong gap between them to accommodate system readiness for two-vehicle deployment and to assess and validate the data collected during the first deployment. The team categorized the collected data for the first deployment into buckets (speed, road type, and AADT), as mentioned in the Driving Environments section. The team planned the second deployment to collect the remaining number of hours for each of these buckets.

Table 6 contains the total number of hours budgeted on project for each deployment vehicle. Data collection hours were spread throughout the first and second deployment. The team collected a total of 144 h. For the first deployment, baseline, RI-ADAS, and D-ADAS data were collected. During the second deployment, data were collected using the D-ADAS and the RI-ADAS-equipped vehicles. Additionally, the two-vehicle data were only collected during the second deployment.

Table 6. Deployment schedule.

Vehicle	Data (h)	Data Sample: July 2021	First Deployment: Sept. 2021–Nov. 2021	Second Deployment: Jan. 2022–March 2022
SV1	120	Validation only	Baseline: 8 h D-ADAS: 30 h RI-ADAS: 58 h	D-ADAS 16 h RI-ADAS: 8 h
SV1 and SV2	24	—	—	2-RI-ADAS: 24 h

— No data.

In total, the team collected 8 h of data using a manually driven vehicle (i.e., the vehicle was operated as a nonautomated, SAE Level 0 vehicle). The team used the D-ADAS-equipped vehicle for 46 h of data collection. Finally, the team used the RI-ADAS-equipped vehicles to collect 90 h of data; of this 90 h, 66 h was collected using a single RI-ADAS-equipped vehicle (SV1), and 24 h of data were collected using multiple RI-ADAS-equipped vehicles (SV1 and SV2) concurrently.

SUMMARY

This chapter summarizes the development of the project's DCP, which answers the following questions about the collected data:

What data were collected throughout the project?

The team collected raw sensor data from the instrumented RI-ADAS-equipped SV and D-ADAS-equipped SV. Both SVs are ADAS-equipped vehicles with SAE Level 2 partial driving automation capabilities (e.g., ACC, lane centering, lane assist, limited sign and signal detection). The vehicle sensor suites are described in more detail in chapter 3.

After the data were collected, the project team processed the sensor data to obtain trajectories for the AdjV operating near the SV (detected by the on-board sensors of the SVs); this process and the specific variables collected for the SVs and the AdjVs are described in chapter 3.

Where were data collected?

The project team collected data in central Ohio on routes shown in figure 7.

How were the data collected?

The project team designed the DCP for three scenarios of interest: vehicle following, lane changing, and signalized intersection approach and departure. Each of these scenarios of interest have specific IoIs. The research team operated the SVs to encourage the occurrence of these IoIs.

When were the data collected?

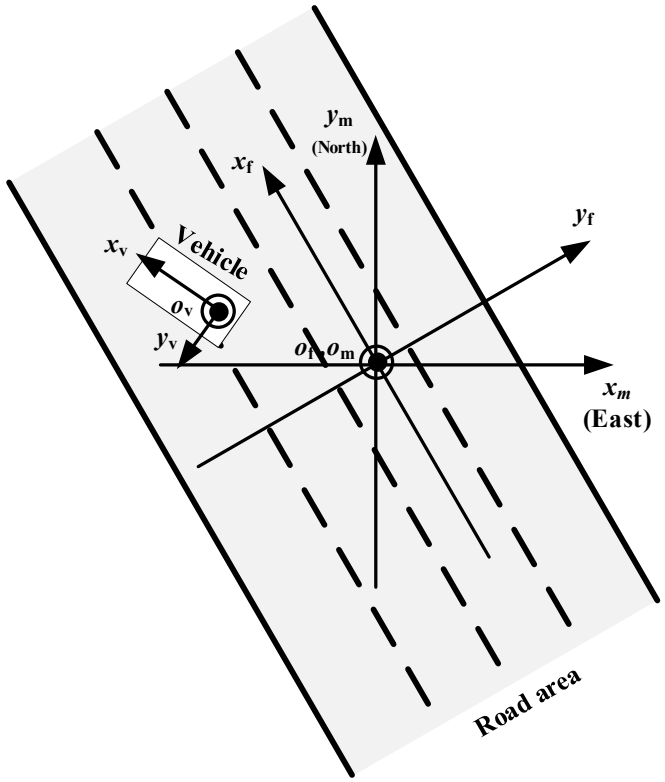
The data were collected between July 2021 and March 2022.

CHAPTER 3. DATA PROCESSING METHOD

This chapter introduces the data processing methods for datasets from a single ADAS-equipped vehicle and from two ADAS-equipped vehicles operating concurrently.

OUTPUT DATA FRAMES

This first section discusses the frames of reference that present the variables in the processed, publicly available comma separate values (CSVs) (see figure 8). Presenting the information of the SV and AdjVs in different frames allows future users to apply the datasets in different applications. The frames include vehicle frame (*v*), Frenet frame (*f*), map frame (*m*), and Earth-centered, earth-fixed (ECEF) frame. Table 7 through table 15 include a column that notes to which reference frame the data belongs. These definitions are described in the following subsections.



Source: FHWA.
o = origin.

Figure 8. Illustration. Frame definition.

Vehicle Frame

The vehicle frame is attached to the center of the SV's rear axle and rotated with the vehicle as the SV moves through space. The vehicle frame moves as the vehicle moves and the road

orientation changes. Figure 8 shows that the x_v, y_v, and z_v of the vehicle frame are toward the front, left, and upward directions, respectively. The origin o_v is at the center of the rear axle. The unit of the coordinates in this frame is meters. Information such as the bounding box of an AdjV is presented in the vehicle frame. The vehicle frame is useful for providing the relative position between the detected AdjVs and the SV as well as the size of the AdjV at the SV view.

Frenet Frame

The Frenet frame is attached to the center of the road, and its orientation will change as the road direction changes. As shown in figure 8, the x_f and y_f are the axes of Frenet frame. The longitudinal forward and lateral right directions of the road are the positive directions of x_f and y_f, respectively. The unit of the coordinate in this frame is meters. The start point of the SV's route is picked as the origin of the map frame (o_f).

The Frenet coordinates can be used to analyze the car-following and lane-changing behavior. The Frenet frame is particularly useful for understanding the relationship between vehicles on the roadway. For example:

- When the AdjV is ahead of the SV, the closest longitudinal distance is the distance from the SV's front bumper to the AdjV's rear bumper in the Frenet frame.
- When the AdjV is behind the SV, the closest longitudinal distance is the distance from the SV's rear bumper to the AdjV's front bumper in the Frenet frame.
- When the AdjV is on the right side of the SV, the closest lateral distance is the distance from the SV's right doors to the AdjV's left doors in the Frenet frame.
- When the AdjV is on the left side of SV, the closest lateral distance is the distance from the SV's left doors to the AdjV's right doors in the Frenet frame.

Map Frame

The map frame is a fixed frame attached to a certain selected point. Positive values for x_m, y_m, and z_m are toward east, north, and upward directions, respectively, as shown in figure 8. Note that the start point of SV's route is selected as the origin of the map frame (o_m). The unit of the coordinate in this frame is meters.

The x and y positions of the AdjVs along with the SV in the map frame can be used for researching the physical model of the vehicles in real traffic because the physical model of the vehicles is usually defined in a map frame.

ECEF Frame

The ECEF frame is a fixed frame attached to the center of mass of the earth. The coordinates in ECEF frame are described by longitude, latitude, and altitude. The longitude and latitude are in degrees and altitude is in meters.

The ECEF used for the GPS is the World Geodetic System (WGS) 84.[16]

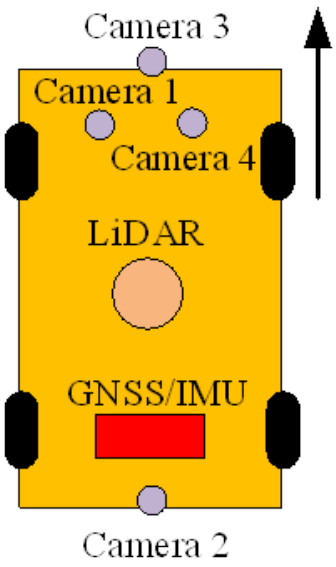
DATA COLLECTION SENSORS

This section discusses the data collection sensor configuration and raw data format. This project used two retrofitted vehicles to collect sensor data. Figure 9 shows the equipped commercial vehicle, referred to from here on as SV1 (note that both D-ADAS- and RI-ADAS-equipped vehicles were used as SV1). The team used SV1 exclusively in the single-vehicle data collection effort and during the two-vehicle data collection effort. To collect the raw sensory data, SV1 was equipped with a 32-line LiDAR, four monocameras (near the front bumper to monitor the area immediately in front of the ADAS-equipped vehicle; in front of the driver seat inside the SV cabin; in front of the passenger seat inside the SV cabin, and near the rear bumper to monitor the area immediately behind the ADAS-equipped vehicle), and a GNSS/IMU integration system with realtime kinematic (RTK) correction. The computer on the vehicle stored the sensory data collected by the vehicle sensors. The format of the raw data varied cross the different sensors:

- Raw LiDAR sensor data were collected in rosbag, which is a file format in ROS (Robot Operating System) for storing ROS message data.[8]
- Raw camera data were saved in videos.
- Raw GNSS/IMU integration system data were saved in comma separated value (CSV) files.

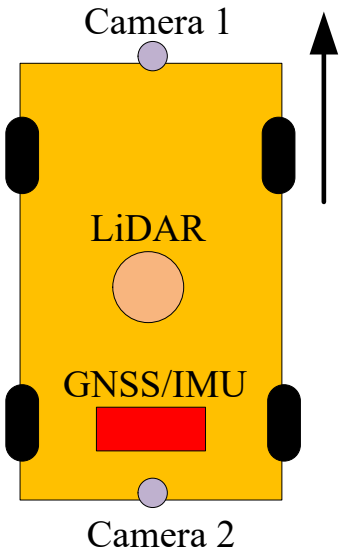
In addition to SV1, the team used a retrofitted DBW vehicle, referred to as SV2, during the second deployment to collect data. This vehicle is shown in figure 10. The project team only used a RI-ADAS as SV2; however, SV2 was operated as an SAE Level 0 vehicle with no ADAS features due to limitations with the map data in the study area. SV2 was equipped with a 32-line LiDAR, two mono-cameras (located near the front and rear bumpers to monitor the area immediately in front and behind of the ADAS-equipped vehicle), and a GNSS/IMU integration system. The computer on the vehicle collected all data from the SV2 sensors and stored the data in rosbags.[8]

The remainder of this chapter discusses the data processing pipeline developed for the one-vehicle (SV1) and two-vehicle (SV1 and SV2) datasets to convert raw sensor data into trajectories for the SV(s) and AdjVs in traffic.



Source: FHWA.

Figure 9. Illustration. Equipped SV1.

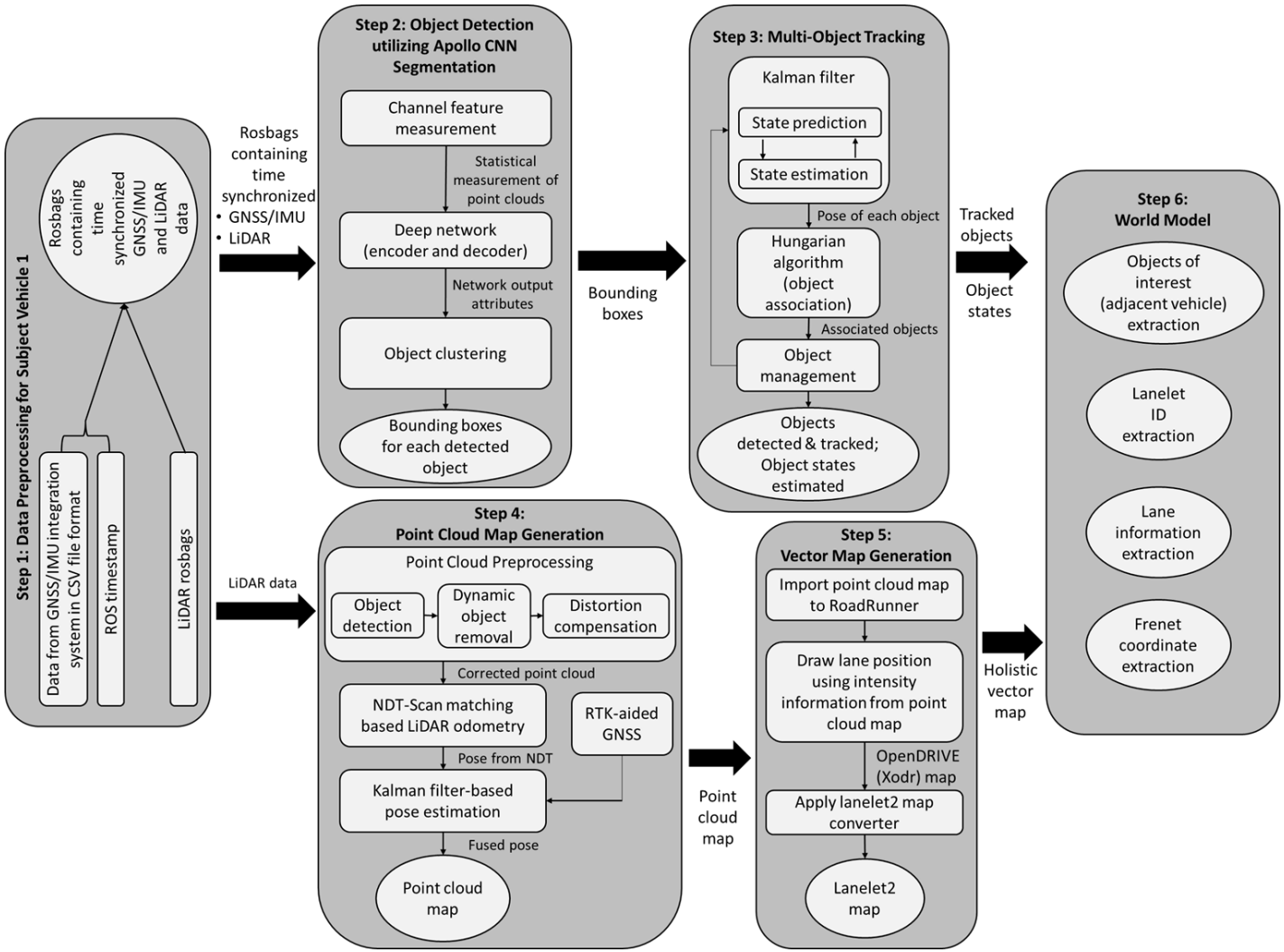


Source: FHWA.

Figure 10. Illustration. Equipped SV2.

DATA-PROCESSING PIPELINE FOR ONE-VEHICLE DATASETS

The data-processing pipeline for the one-vehicle dataset (figure 11) contains six steps: data preprocessing, object detection, multiobject tracking, point cloud map generation, vector map generation, and World Model creation.[17,18] This section details how the GNSS/IMU and LiDAR data were processed to produce trajectories for all AdjVs detected by the SV sensors.



Source: FHWA.
NDT = normal distribution transformation; CNN = convolutional neural network.

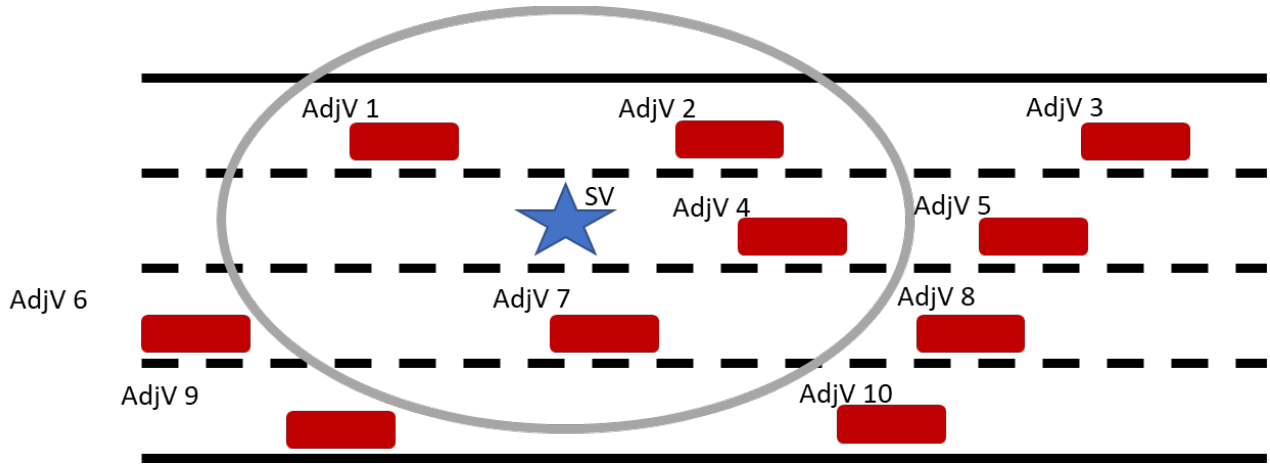
Figure 11. Illustration. Data-processing pipeline for one-vehicle dataset.

Step 1: Data Preprocessing for SV1

The project team used ROS[8] to program the data-processing method. However, the raw data collected on the equipped SV1 includes CSV files, videos, and rosbags. The team used the data-processing rosbag application programming interface (API) to convert sensor data collected in CSV files to the rosbag format and ensured all sensor data were synchronized to the correct timestep.[19] For the SV1 datasets, the CSVs included the sensor information from the timestamped GNSS/IMU integration system and the ROS time on the computer used to collect LiDAR data. As shown in step 1 of figure 11, the data-processing software synchronized the timestamped GNSS/IMU data based on the ROS timestamps in the CSVs.[38] The time-synchronized GNSS/IMU data were then added to the rosbags containing the LiDAR data for further processing in step 2.

Steps 2 and 3: Object Detection and Tracking

After preprocessing the SV1 datasets (i.e., the raw data were converted to rosbags), the data-processing pipeline, shown in figure 11, processed the LiDAR and GNSS/IMU module data to extract the information of the AdjVs near the SV (step 2 in figure 11). The inputs to step 2 were the rosbags produced in step 1. In step 2, AdjVs within the LiDAR sensing range were detected and tracked. In figure 12, the circle surrounding the SV represents the detection range of SV1's LiDAR sensor (around 40–60 m on the actual data collection vehicle). Step 2 produces bounding boxes for all AdjVs that can be detected by SV1's LiDAR. Step 3 estimates trajectories of all vehicles (i.e., position, speed, acceleration, orientation, and ID number) within the detection range of the SV1's LiDAR. In figure 12, trajectories for AdjVs 1, 2, 4, and 7 can be estimated because they are in the detection range of SV1's sensors, while data about AdjVs 3, 5, 6, 8, 9, and 10 cannot be estimated because they are outside of the detection range of the LiDAR sensor.



Source: FHWA.

Figure 12. Illustration. AdjV detection for one-vehicle datasets.

Step 2: Object Detection

In step 2, the data-processing pipeline uses Apollo CNN segmentation, an open-source object-detection package based on three-dimensional (3D) LiDAR data, to segment out the vehicles of interest in the LiDAR point cloud frame.[20] Apollo CNN segmentation accepts point clouds as input and segments out the objects detected in the point cloud. For this application, Apollo CNN detected AdjVs from the vehicle sensor data. The segmentation consists of three main steps:

- Step 2.1: Channel feature extraction. Divides the bird's-eye view (BEV) projection of the point cloud into cells and calculates the static measurements of each cell. The channel feature contains statistical measurements of the point cloud, such as height and intensity.

- Step 2.2: Encoder-decoder network. Takes the measurements and predicts the necessary attributes for each cell.

- Step 2.3: Cell clustering algorithm. Finds the candidate clusters and uses a postprocessing method to filter out some candidate clusters.

During channel feature extraction, Apollo CNN segments the LiDAR point cloud objects and applies a two-dimension (2D) convolutions architecture.[20] First, the algorithm converts the LiDAR point cloud into 2D BEV grid space and represents the point cloud by statistical measurements. Specifically, each cell in grid space computes eight different measurements of the point clouds, as follows:

- Maximum height of points in the cell.
- Intensity of the highest point in the cell.
- Mean height of points in the cell.
- Mean intensity of points in the cell.
- Number of points in the cell.
- Angle of the cell's center with respect to the origin.
- Distance between the cell's center and the origin.
- Binary value indicating whether the cell is empty or occupied.

The whole measurement $A \in \boldsymbol{R}^{W \times H \times 8}$ is used as the network input, where the W and H represent the number of rows and columns of the grid, and 8 is the statistical measurement of the point clouds for each cell. The deep network used by Apollo CNN consists of an encoder and decoder.

In step 2.2, the network takes the channel feature measurement A as input and predicts five cellwise attributes, as follows:

- Center offset (vector): Points to the center of objects.
- Object: Indicates whether a cell contains objects.
- Positiveness and object height: Filters out the background cluster and removes the points that are too high.
- Class probability: Classifies each cluster.

The encoder maps the computed channel feature A to an abstract feature map, whereas the decoder takes the feature map as input, processes it, and produces the attributes $Y \in \boldsymbol{R}^{WxHx5}$.

In step 2.3, after obtaining the network output attributes (step 2.2), Apollo CNN segmentation clusters cells based on their center offset.[20] The algorithm first judges whether the current cell contains objects and then clusters the adjacent cells pointed to by the current cell's center offset. After Apollo CNN segmentation iterates all the cells, it generates a number of candidate clusters, which include several cells. Based on these cells, Apollo CNN segmentation removes some candidate clusters having a small number of points or a low confidence score. Apollo CNN publishes the remaining clusters and classifies them into different categories such as vehicles. The point cloud in these clusters is used to fit bounding boxes as their envelope. These output bounding boxes are defined as $D = [x, y, z, l, h, w, \varphi]$, where x, y, and z are coordinates of the bounding box; l, h, and w are the length, height, and width of the bounding box, respectively; and φ is the heading angle. These bounding boxes ($D = [x, y, z, l, h, w, \varphi]$) for each detected object (AdjV) are the output of step 2 and input to step 3: the multiobject tracking algorithm.

The open-source algorithm used in step 2 for object detection is available through GitHub.[20]

Step 3: Multiobject tracking

The multiobject tracking pipeline (step 3 in figure 11) uses the bounding boxes from the object-detection algorithm (i.e., Apollo CNN) as inputs to track the objects and outputs estimations of the position, speed, orientation, and unique ID of the corresponding object (i.e., the AdjVs).[21,22] Taking the 3D bounding boxes ($D = [x, y, z, l, h, w, \varphi]$) as input, the tracking algorithm's objective is to associate the detected 3D bounding boxes with tracks and obtain the tracking state ($x, y, z, l, h, w, \varphi, v_x, v_y, v_z, ID$) for each vehicle, where v_x, v_y, and v_z are the velocities in 3D space, and ID is the unique ID number of each vehicle.

The object tracking algorithm has three steps:

- Step 3.1: Object state prediction and estimation using Kalman filter.[23]
- Step 3.2: Object association using Hungarian algorithm.[22]
- Step 3.3 Object management using a threshold-and-logic-based method as deduced by the research team.

The Kalman filter has two steps to estimate the states, also known as the pose, of the corresponding object: state prediction and state estimation.[23] Then, the Hungarian algorithm is

used for object association.[22] This step associates the current detection results with existing objects. The object-association algorithm calculates the distance between the predicted position of the tracked objects and the current detection results. Finally, the object management algorithm adds new objects and deletes old objects. The algorithm adds new objects if detected objects cannot be associated to any existing objects. Similarly, the object is deleted if it is not associated with any detected objects after a period of time. A unique ID number is assigned to the newly added objects and correspondingly for the object management purpose.

The Kalman filter used by the object tracking algorithm estimates the object's velocity in 3D space (v_x, v_y, and v_z). As the vehicle moves in the horizontal direction, the total speed v of the object is obtained by $v=(v_x^2+v_y^2)^{1/2}$.

The open-source algorithm used in step 3 for multiobject tracking is available through GitHub.[21]

Steps 4–6: Map Generation and World Model[17]

The result of the multiobject tracking algorithm step is that all objects near the SV are detected and tracked, and the states of the objects are estimated by the Kalman filter in a Cartesian coordinate system (i.e., the map frame described in the Output Data Frames section).[23] The map coordinates produced by the multiobject tracking algorithm are fixed frames in which the *x*-, *y*-, and *z*- axes are to the east, north and upward directions, respectively. However, Frenet coordinates are a method for representing positions in a structured environment (e.g., a road) in a more intuitive way than map/Cartesian coordinates.[24] Thus, having the information in Frenet coordinates is desired for vehicle interaction analyses, such as car-following behavior. To provide the information in Frenet coordinates, high-definition (HD) maps providing road information for the specific runs during the tests are required. This section introduces the process to generate the HD maps based on the collected sensor data.

Steps 4 and 5 in figure 11 show the HD map generation pipeline including the generation of the point cloud and vector maps.[25] First, the algorithm uses the point cloud map to infer the lanes of the road. However, dynamic objects (i.e., AdjVs) will block the lanes and prevent the algorithm from correctly inferring the lanes. Thus, the software first preprocesses the point cloud from the LiDAR to remove the point cloud belonging to dynamic objects such as AdjVs; in this step, the algorithm also compensates for the distortion due to the movement of the SV in the creation of the corrected point cloud of the lane lines. Next, an NDT scan matching algorithm is applied to compute the relative transformation between two consecutive LiDAR frames.[26] LiDAR odometry is constructed to provide the pose of the LiDAR; then the LiDAR pose is fused with the pose from an RTK-aided GNSS module within a Kalman filter.[23] Based on the fused pose, the algorithm in step 4 transforms the point clouds into map coordinates to generate the point cloud map. In addition, this point cloud map is imported into RoadRunner and is used to provide lane information of the road to generate the OpenDRIVE map, which is converted to a vector map through an OpenDRIVE to lanelet2 map converter.[27,28,29]

Step 4: Point Cloud Map Generation

First, a preprocessor adjusts the raw point cloud in the LiDAR frame before generating a map from the LiDAR point cloud. The preprocessing of the LiDAR point cloud includes compensating for the distortion due to the movement of the SV and removing the dynamic vehicles (AdjVs and other dynamic objects) on the roads. The Apollo CNN segmentation object-detection method is used to detect dynamic vehicles. Once the dynamic vehicles are detected, the vehicles' point clouds are removed from the current LiDAR frame. Then, the point cloud preprocessing algorithm compensates for the distortion in the point cloud without dynamic vehicles by combining the velocity information and angular rate information. Using the preprocessed point cloud as input, an NDT scan matching algorithm is applied to associate the two consecutive LiDAR frames and compute the relative transformation between them.[26] The basic principle of NDT scan matching is shown by equations 1–6. The current LiDAR points residing within a cell are transformed into a normal distribution. For cell *k* in the total *n* cells, the mean vector **q** of the points and the covariance matrix **C** are computed by equation 1 and equation 2.

$$q = \frac{1}{n}\sum_{k=1}^{n} x_k \tag{1}$$

$$C = \frac{1}{n-1}\sum_{k=1}^{n} (x_k - q)\,(x_k - q)^{\top} \tag{2}$$

Where:

x_k = the points and *k* means the *k*th number in total cells.

T = the transpose mathematical operator when in the superscript position.

Based on a certain transformation in equation 3, for points in the current LiDAR frame *L*, the coordinates of the transformed points in the previous LiDAR frame coordinates *L′* are

$$X' = R_{L'}^{L} x + t_{L'}^{L} \tag{3}$$

Where:

X = a vector containing the coordinates of the points in the current LiDAR frame.

X′ = from the previous frame of **X**.

$\mathbf{R}_{\mathrm{L}}^{\mathrm{L}'}$ = the rotation matrix defined by equation 4.
$\mathbf{t}_{\mathrm{L}}^{\mathrm{L}'}$ = the translation vector defined by equation 5.
prime (′) indicates data is from previous frame.

$$\mathbf{R}_{L'}^{L} = \begin{bmatrix} \mathrm{c}\varphi\mathrm{c}\theta & -\mathrm{s}\varphi\mathrm{c}\phi + \mathrm{c}\varphi\mathrm{c}\theta\mathrm{s}\phi & \mathrm{s}\varphi\mathrm{s}\phi + \mathrm{c}\varphi\mathrm{s}\theta\mathrm{c}\phi \\ \mathrm{s}\varphi\mathrm{c}\theta & \mathrm{c}\varphi\mathrm{c}\phi + \mathrm{s}\varphi\mathrm{s}\theta\mathrm{s}\phi & -\mathrm{c}\phi\mathrm{s}\phi + \mathrm{s}\varphi\mathrm{s}\theta\mathrm{c}\phi \\ -\mathrm{s}\theta & \mathrm{c}\theta\mathrm{s}\phi & \mathrm{c}\theta\mathrm{c}\phi \end{bmatrix} \tag{4}$$

Where:
φ, θ, and ϕ = the roll, pitch, and heading angles, respectively, between the LiDAR frame coordinate and the map coordinate and are denoted by the asterisk in the shorthand that follows.
$c*$ = cos(*).
$s*$ = sin(*).

$$\boldsymbol{t}_{\boldsymbol{L'}}^{\boldsymbol{L}} = [\boldsymbol{t}_{\boldsymbol{x}}, \boldsymbol{t}_{\boldsymbol{y}}, \boldsymbol{t}_{\boldsymbol{z}}]^{\top} \tag{5}$$

Where t_x, t_y and t_z are the translation in the x, y, and z directions of the map frame, respectively.

The NDT scan matching algorithm searches for the best transformation ($\boldsymbol{R}^{L}{}_{L'}$, $\boldsymbol{t}^{L}{}_{L'}$) in the transformation function for the points in the current LiDAR frame to match the points in the previous LiDAR frame. The cost function, J, for the parameters in the transformation function is defined as:

$$\boldsymbol{J} = -\sum_{\boldsymbol{k}=\boldsymbol{1}}^{\boldsymbol{n}} \boldsymbol{e}\,\boldsymbol{xp}\,\frac{(\boldsymbol{x}_{\boldsymbol{k}}' - \boldsymbol{q}')^{\top}(\boldsymbol{C}')^{-\boldsymbol{1}}(\boldsymbol{x}_{\boldsymbol{k}}' - \boldsymbol{q}')}{\boldsymbol{2}} \tag{6}$$

Where:
$\boldsymbol{q}'$ = the corresponding mean vector in the previous LiDAR frame.
$\boldsymbol{C}'$ = the covariance matrix of the point cloud in the previous LiDAR frame.
$\boldsymbol{x}_k^{'}$ = the points residing in cell k collected in the previous time stamp.

Newton's algorithm is used to iteratively solve the optimization problem to find the best transformation $\boldsymbol{T}^{L}{}_{L}{}'$ between the current LiDAR coordinates L and the previous LiDAR frame coordinates, L'.[30] Taking the transformation in equation 7,

$$\boldsymbol{T}_{\boldsymbol{L}'}^{\boldsymbol{L}} = \begin{bmatrix} \boldsymbol{R}_{\boldsymbol{L}'}^{\boldsymbol{L}} & \boldsymbol{t}_{\boldsymbol{L}'}^{\boldsymbol{L}} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \tag{7}$$

the LiDAR odometry from LiDAR frame 0 to LiDAR frame k can be constructed as $\boldsymbol{T}^{L(k)}{}_{L(0)'} = \boldsymbol{T}^{L(k)}{}_{L(k-1)'}, \ldots \boldsymbol{T}^{L(2)}{}_{L(1)'}, \boldsymbol{T}^{L(1)}{}_{L(0)'}$:

Where:

$\boldsymbol{T}^{L(k)}{}_{L(0)'}$ = the transformation from L(0)′ to L(k).
$\boldsymbol{T}^{L(k)}{}_{L(k-1)'}$ = the transformation from L(k−1)′ to L(k).
$\boldsymbol{T}^{L(2)}{}_{L(1)'}$ = the transformation from L(1)′ to L(2).
$\boldsymbol{T}^{L(1)}{}_{L(0)'}$ = the transformation from L(0)′ to L(1).

However, due to the noise contained in the LiDAR frame, $\boldsymbol{T}^{L}{}_{L}{}'$ has errors which will be accumulated in the transformation $\boldsymbol{T}^{L(k)}{}_{L(0)}{}'$. To compensate for these errors, the translation and heading information provided in an RTK-aided GNSS module is used to provide the measurements for the transformation from the LiDAR odometry because the information from the GNSS module is free of accumulated errors. To fuse the LiDAR odometry pose and the GNSS module pose, a Kalman filter is applied.[23] Then, using the fused pose from the Kalman filter, all the points in the LiDAR frames are transformed to the map coordinates for generating the map. The transformation function for generating the point cloud map is defined as

$$\boldsymbol{T}(\boldsymbol{\alpha}) = \boldsymbol{C}_{\boldsymbol{L}}^{\boldsymbol{m}}\boldsymbol{\alpha} + \boldsymbol{t} \tag{8}$$

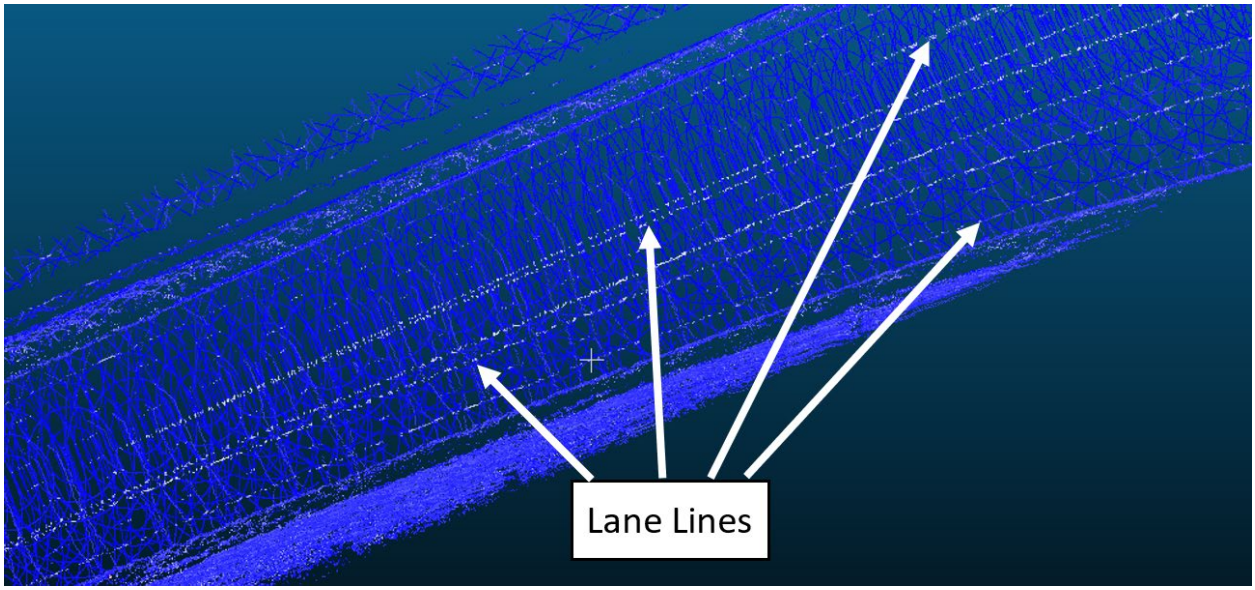
Where:

$\boldsymbol{\alpha} = [x, y, z]^T$ denotes the coordinates of the points in a LiDAR frame of the 3D LiDAR.
$\boldsymbol{C}^{m}{}_{L}$ = the rotation matrix between LiDAR coordinates and map coordinates.
$\mathbf{t} = [t_x, t_y, t_z]^T$ is the translation vector.

The LiDAR frames from time t_0 to t_k are aggregated to the LiDAR point cloud map using the point cloud in the map frame. Note that in real-time application, the range of the LiDAR sensor is large, so aggregating all the LiDAR frames is not necessary to generate maps. Involving all the point cloud frames in the map is too large, and this large point cloud map consumes too much hard drive space and computational resources. Therefore, to reduce the size of the point cloud map, after the SV travels a certain distance (i.e., 16.4 ft is the threshold in this application), adding a LiDAR frame into the point cloud map is sufficient for real-world application.

The result of step 4 is a point cloud map of the road surface.

Figure 13 presents a visualization of a point cloud map created by the LiDAR unit installed on the SV as collected during a freeway scenario. This point cloud map is the product of step 4 of the One-Vehicle Data Processing Pipeline (figure 11). This point cloud corresponds with the OpenDRIVE map in figure 14-A and the vector map in figure 14-B.[29]



Source: FHWA.

Figure 13. Illustration. Sample visualization of the point cloud map (input to RoadRunner).[27]

The color gradation of the points is indicative of their respective intensity information. The lane lines are illustrated in a lighter hue and are also annotated on the figure. This color distinction signifies intensity differentiation and can be leveraged to generate a vector map.

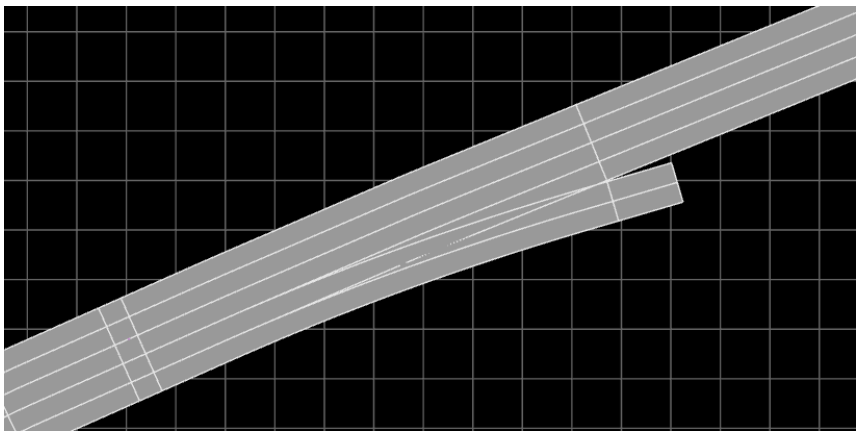
Step 5: Vector Map Generation

This section discusses the vector map generation method. To create the vector map, the generated point cloud map from the point cloud generation step (an example is shown in figure 13) is imported to RoadRunner 2021.[27] In the point cloud map, elements with a different color on the road surface have different intensity information. A project team member manually drew the road based on the exact lane position, which can be inferred by the intensity information in the point cloud map as the lane information can be visualized in RoadRunner.[27] A project team member generated the maps for all areas where data were collected.

Because the vector map (lanelet2 map) cannot be directly generated in RoadRunner, the output of this process is the OpenDRIVE (xodr) format map, which can be exported directly from RoadRunner. Next, the team member uses the lanelet2 map converter to convert the OpenDRIVE map to a lanelet2 map.[28,29] The lanelet2 map is used for two purposes. First, the lanelet map can provide the constraints to filter out off-road objects for object-detection and tracking modules (these off-road objects are not of interest because they have no interaction with the SV or AdjV). Second, the lanelet2 map provides the lane information of the road. By definition, a lanelet is an atomic interconnected drivable road segment. (Atomic means that

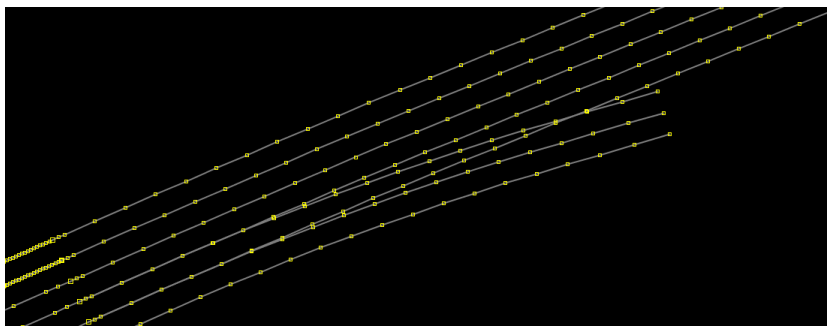
currently valid traffic rules do not change within a lanelet and that the topological relationships with other lanelets do not change.) Note that in this report, the researchers will use the terms lanelet2 map and vector map interchangeably as vector map is a general term and lanelet2 map is specific to the applications in this project.

Figure 14 provides a visualization of the OpenDRIVE map (figure 14-A), constructed using RoadRunner software, and the lanelet2 map (figure 14-B), produced by the lanelet2 map converter.[27,28,29] Both maps cover the area depicted in the point cloud map shown in figure 13. The primary roadway is composed of four lanes, and the adjoining ramp consists of two lanes.



Source: FHWA.

A. Visualization of OpenDRIVE map (output of RoadRunner; input of lanetlet2 map converter).[27,28,29]



Source: FHWA.

B. Visualization of lanelet2 map (output of lanelet2 map converter; input to World Model).[29,17]

Figure 14. Illustration. Visualization of the vector map.

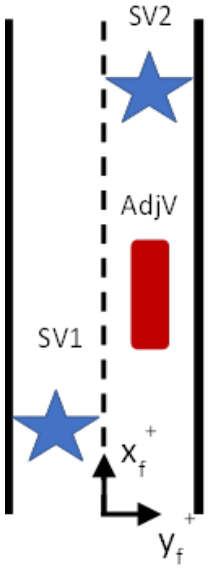
Together, steps 4 and 5 comprise the HD map-generation pipeline. The HD map-generation pipeline was generated by the research team for a previously completed project. The code for the HD map generation project is not open source, but documentation on the previous project are available for reference.[25]

The World Model, described in the next section, applies the road's lanelet information to provide the coordinates of the objects in the Frenet frame.[17]

Step 6: World Model

Step 6 of the one-vehicle data-processing pipeline (figure 11) is mostly about generating different perspectives of the data (e.g., map coordinates, Frenet coordinates). The data processing pipeline uses World Model, developed in CARMA platform, to determine if the objects are on the road.[17] World Model is an interface that provides the supported access functions for working with the lanelet map and object route, such as computing downtrack and crosstrack distances (Frenet coordinates). World Model is tightly coupled to the lanelet2 library and relies on lanelet2 primitives for functionality.[31] If the objects are on the road, the World Model obtains the coordinates of the objects as Frenet coordinates. The lanelet information and the lane information of the objects are also output from the World Model.[18]

As shown in figure 15, the downtrack distance is the *x* in Frenet coordinates along the longitudinal direction of the road. The crosstrack distance is the *y* in Frenet coordinate along the lateral direction of the road. The downtrack and crosstrack information of the objects can be used to analyze the car-following behavior and lane-change behavior of the objects (AdjVs). For example, in figure 15, SV2 is leading the AdjV and is located in the positive downtrack direction away from the AdjV, while SV1 is following the AdjV and in the left lane (so SV2 is located in the negative downtrack and crosstrack directions from the AdjV).



Source: FHWA.

Figure 15. Illustration. Relationship between SVs and an AdjV in Frenet frame.

For the AdjV, given its speed *v* (from object tracking) and the x-coordinate of the AdjV in Frenet coordinates (longitudinal direction), the project team takes the first order of the derivative of the speed to calculate the acceleration. The estimation of acceleration remains a significant challenge in research.

Due to the noise in the estimated speed from the object tracking algorithm, the project team used a discrete finite impulse response (FIR) filter to smooth the acceleration.[32] The window size in the FIR filter directly influences the smoothness of the acceleration. A larger window size results

in smoother acceleration but introduces a temporal delay. Conversely, a smaller window size leads to a quicker response time but an increase in noise within the acceleration.

While the team provides the estimated acceleration in the CSV files using specific FIR filter settings, this might not necessarily yield the most accurate estimation due to varying requirements for acceleration in different scenarios. Therefore, the team encourages users of the CSV files to adjust the parameters in the FIR filter according to different scenarios to enhance the estimation of acceleration, as AdjVs may exhibit varied dynamics. As an alternative, the Kalman filter is also a good solution to estimate the acceleration of the adjacent object. The details of the implementation of the Kalman filter can be found in Xiong et al.[33] Note that, like the FIR filter, the measurement covariance and system state covariance for the Kalman filter can be tuned to reflect differing dynamics under various scenarios (e.g., city roads versus highways). A smaller measurement covariance can lead to a faster response (but increase noise in the acceleration estimates), whereas a larger measurement covariance can provide a smoother estimation of acceleration at the cost of a larger temporal delay.

For the SV, the ground-truth position information from the GNSS/IMU system from the original CSVs are converted to the *x*-coordinate in the Frenet coordinates. The speed is acquired directly from the GNSS/IMU system. Both pieces of information are reliable and stored in the CSV files. Researchers can use this information to estimate the acceleration of the SV using either the aforementioned approaches or other advanced estimation techniques developed by the users of the CSV files.

The performance of speed and acceleration estimation is elaborated upon in the Validation of AdjV Speed and Acceleration section. The team used the FIR filter method to generate acceleration in the CSV files due to its marginally superior performance (in terms of the standard deviation error) over the Kalman filter method.[23,32,33]

OUTPUT OF ONE-VEHICLE DATASETS

This section discusses both the data produced by the one-vehicle data-processing pipeline and provides sample graphics that demonstrate how well the data processing pipeline can detect and track objects (i.e., AdjVs).

Output Description

The data processing pipeline outputs include the position, velocity, acceleration, and orientation of objects of interest in different reference frames. These data are all stored in an easily sharable format (CSVs). The contents of the CSVs produced by the data processing pipeline shown in figure 11 are described in the following subsections.

Variables for AdjVs

The variables for the AdjVs are described in table 7. The variables are presented in different frames shown in figure 8. Figure 16 and figure 17 provide the definitions for closest_distance_longitudinal, closest_distance_lateral, distance_adjv, lanelet_ID_adjv, and lane_ID_adjv.

Table 7. Variables for the AdjVs (one vehicle dataset).

Variable	Description	Column in CSV	Unit	Frame	Access Method
ID	Identification number of the AdjV (ascending by time of entry into the sensor range of the SV)	A	n/a	n/a	Calculated
Time	Timestamp (ascending by start time) of the corresponding row in CSV. Time is the ROS time converted into a more user-friendly format[8]	B	s	n/a	Calculated
distance_adjv (headway)[1]	Distance between the center of the AdjV and the SV	C	m	n/a	Calculated
pos_x_adjv_f	AdjV *x* position	D	m	Frenet	Calculated
pos_y_adjv_f	AdjV *y* position	E	m	Frenet	Calculated
pos_x_adjv_m	AdjV *x* position	F	m	Map	Calculated
pos_y_adjv_m	AdjV *y* position	G	m	Map	Calculated
heading_adjv_m	AdjV heading angle (orientation of vehicle)	H	degree	Map	Calculated
dim_x_adjv[2]	AdjV length	I	m	Vehicle	Calculated
dim_y_adjv[2]	AdjV width	J	m	Vehicle	Calculated
dim_z_adjv[2]	AdjV height	K	m	Vehicle	Calculated

Variable	Description	Column in CSV	Unit	Frame	Access Method
speed_adjv	AdjV speed	L	m/s	Frenet/ map	Calculated
acc_adjv	AdjV acceleration	M	m/s^2	Frenet	Calculated
closest_distance_longitudinal (gap)[1,3]	The closest distance between AdjV and SV in the longitudinal direction	X	m	Frenet	Calculated
closest_distance_lateral[4]	The closest distance between AdjV and SV in the longitudinal direction	Y	m	Frenet	Calculated
lanelet_id_adjv[5]	Lanelet ID of the AdjV's center point	AG	n/a	n/a	Calculated
lane_id_adjv[5]	Lane ID of the AdjV's center point	AH	n/a	n/a	Calculated
total_lanes	Total lanes at the current position	AK	n/a	n/a	Calculated

f = Frenet; m = map; v = vehicle; n/a = not applicable.

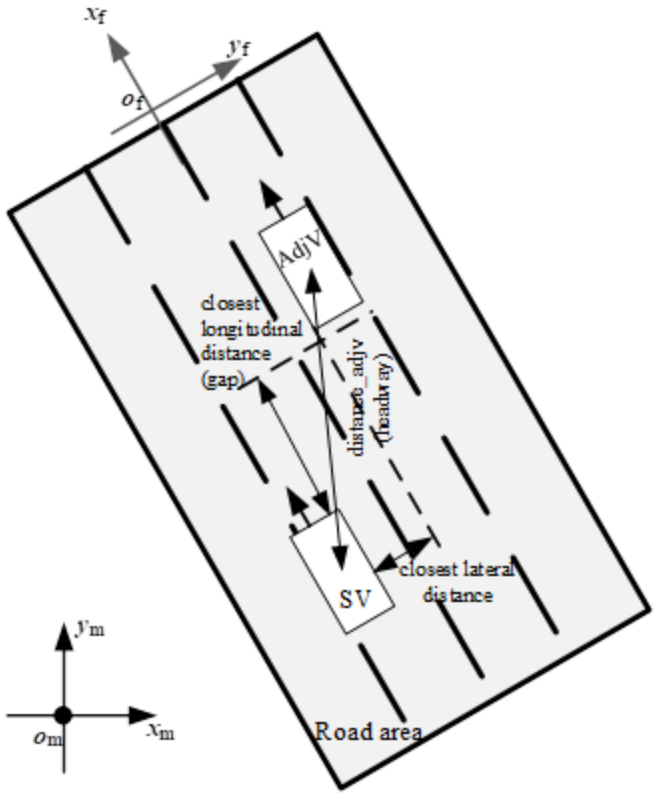
[1]As shown in figure 16, the headway between two vehicles (measured between the same point on the leader-follower pair) is the distance_adjv. The distance_adjv is measured from the centroid of the leading vehicle to the centroid of the following vehicle. The gap between vehicles (measured between the rear bumper of the leading vehicle and the front bumper of the following vehicle) is recorded as the closest_longitudinal_gap.

[2]The research team set fixed values of the bounding boxes to prevent the variation of the bounding box for the same object over different frames.

[3]As shown in figure 16, when the AdjV is ahead of the SV, the closest longitudinal distance is the distance from the front bumper of the SV to the rear bumper of the AdjV in the Frenet frame. When the AdjV is behind the SV, the closest longitudinal distance is the distance from the rear bumper of the SV to the front bumper of the AdjV in the Frenet frame.

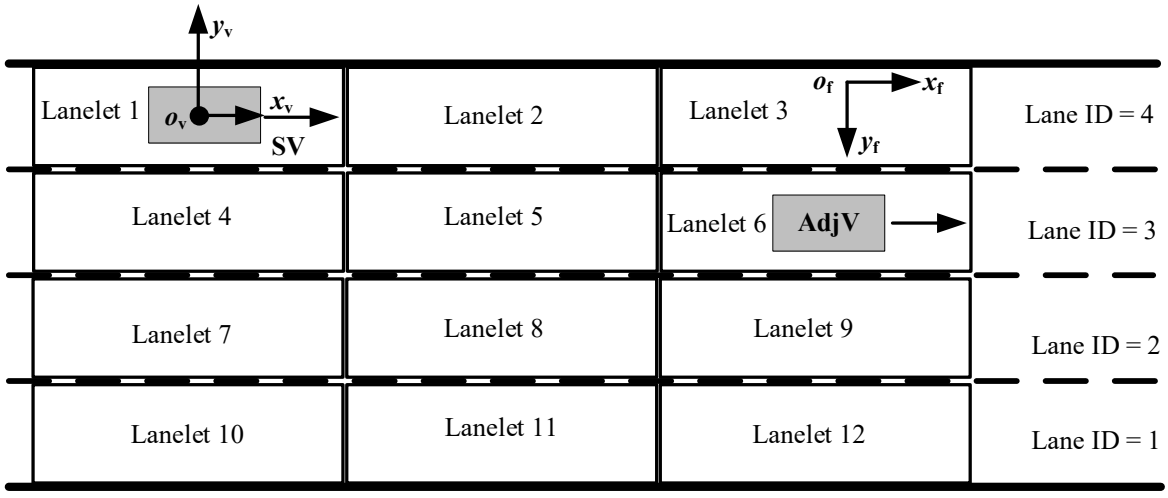
[4]As shown in figure 16, when the AdjV is on the right side of the SV, the closest lateral distance is the distance from the right doors of the SV to the left doors of the AdjV in the Frenet frame. When the AdjV is on the left side of SV, the closest lateral distance is the distance from the left doors of the SV to the right doors of the AdjV in the Frenet frame.

[5]As shown in figure 17, in Frenet coordinates, the road with four lanes is divided into lanelets from lanelet 1 to lanelet 12. The lanelet ID shows the exact ID where the vehicle is. The lane ID shows which lane the vehicle is. The lanelet ID and lane ID are based on the vector map information. The lane ID starts from the right side of the road and increases to the left side of the road. For instance, the lanelet ID for the SV is 1 and the lanelet ID for the AdjV is 6. For SV, its lane ID is 4 and the total number of lanes is 4. The lane ID for the AdjV is 3.



Source: FHWA.

Figure 16. Illustration. Relationship between distance_adjv, closest_longitudinal_distance, and closest_lateral_distance.



Source: FHWA.

Figure 17. Illustration. Demonstration of lanelet and lane information.

Variables for SVs

The variables for the SVs are described in table 8. Because the primary purpose of these datasets is to evaluate the interaction between the SV and any AdjVs, information regarding both the SV and AdjVs is recorded and stored together as a pair within the CSV files. Essentially, the SV's data is only necessary when an AdjV is present, indicating an interaction. If no AdjV is present, implying a lack of interaction, recording and storing the SV’s data is not necessary (e.g., without a leading AdjV, the SV is free to drive at its desired speed).

Table 8. Variables for the SVs.

Variable	Description	Column in CSV	Unit	Frame	Access Method
Time	Timestamp (ascending by start time) of the corresponding row in CSV	B	s	n/a	Calculated
pos_x_sv_f[2,3]	SV *x* position	N	m	Frenet	Calculated
pos_y_sv_f[2,3]	SV *y* position	O	m	Frenet	Calculated
pos_x_sv_m	SV *x* position	P	m	Map	Measured
pos_y_sv_m	SV *y* position	Q	m	Map	Measured
heading_sv	SV heading	R	degree	Map	Measured
dim_x_sv	SV length	S	m	Vehicle	Measured
dim_y_sv	SV width	T	m	Vehicle	Measured
dim_z_sv	SV height	U	m	Vehicle	Measured
speed_sv[2,3]	SV speed	V	m/s	Frenet/ map	Measured
acc_sv[2,3]	SV acceleration	W	m/s^2	Frenet	Calculated
lanelet_id_sv[1]	Lanelet ID of the SV's center point	AI	n/a	n/a	Calculated
lane_id_sv[1]	Lane ID of the SV's center point	AJ	n/a	n/a	Calculated

[1]As shown in figure 17, in Frenet coordinates, the road with four lanes is divided into lanelets from lanelet 1 to lanelet 12. The lanelet ID shows the exact ID where the vehicle is. The lane ID shows which lane the vehicle is. The lanelet ID and lane ID are based on the vector map information. The lane ID starts from the right side of the road and increases to the left side of the road. For instance, the lanelet ID for the SV is 1 and the lanelet ID for the AdjV is 6. For SV, its lane ID is 4 and the total number of lanes is 4. The lane ID for the AdjV is 3.

[2]A result of interpolating data and running it through the processing method detailed in chapter 3 is that multiple SV positions, velocities, and accelerations may exist for the same timestamp when the SV encounters multiple AdjVs. The SV's position (Frenet), velocity, and acceleration are all outputs of the processing method, and this can cause very minimal differences in these variables.

[3]If a future user of the data plots the SV's processed trajectories, these trajectories will appear discontinuous. This discontinuity is not because missing data is missing but is due to the SV's position only being provided if an AdjV is detected. This decision was made to simplify how data was stored (every row is an instance where an AdjV is detected near the SV). If an AdjV is not detected, one can assume that the SV will continue to operate at its desired speed until a new AdjV is detected and impedes the SV's path.

Variables for Origins of the Map and Road

The variables for the road and maps are described in table 9.

Table 9. Information of map origin and road origin (one vehicle dataset).

Variable	Description	Column in CSV	Unit	Frame
map_origin_x	Map origin longitude	Z	degree	ECEF
map_origin_y	Map origin latitude	AA	degree	ECEF
map_origin_z	Map origin altitude	AB	degree	ECEF
road_origin_x_m	Map origin *x* position	AC	m	Map
road_origin_y_m	Map origin *y* position	AD	m	Map
road_origin_x_ecef	Road origin longitude	AE	degree	ECEF
road_origin_y_ecef	Road origin latitude	AF	degree	ECEF

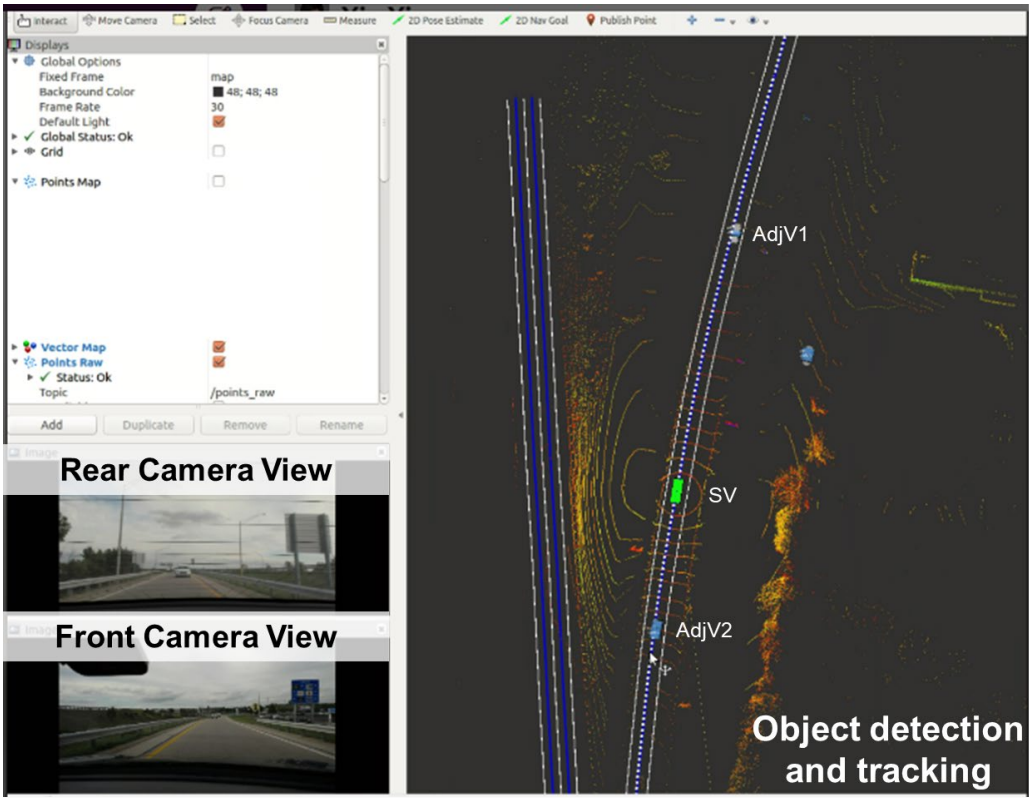
The variables for metadata are represented in table 10.

Table 10. Included metadata information (one vehicle dataset).

Variable	Description	Column in CSV	Unit
Total lanes	Number of lanes on one side of the road	AK	n/a
Run number	Number from the set of processed runs	AL	n/a
Sub run number	Number of sub run for respective run number from processed dataset	AM	n/a
Date	Date of data collection	AN	n/a
Time of day	Time stamp of data collection	AO	n/a
Sub run start time	Start time from the original run from where data was processed	AP	n/a
Route starting point(rs)	Google map start point[15]	AQ	n/a
Route ending point(re)	Google map end point[15]	AR	n/a
fdistance	Distance of the route	AS	miles
maplink	Google maps link of the route[15]	AT	n/a
Annual traffic density	AADT for the route	AU	n/a
Roadway type	Type of roadway: limited access/divided/nondivided arterial	AV	n/a
Speed limits	Speed limits along the route	AW	mph
Road condition	Condition of surface of road: wet/dry	AX	n/a
Type of vehicle	Appearance/operation of vehicle: RI/DI/baseline	AY	n/a
Aggressiveness	Aggressiveness setting for SV1	AZ	n/a
Following distance	Following distance setting for SV1	BA	n/a
Special notes	Any interesting observations	BB	n/a

Output Visualization

Figure 18 shows the object-detection and tracking results of a freeway scenario during the one-vehicle dataset data collection. The forward-facing camera data, rear-facing camera data, and LiDAR point cloud data are visible in figure 18. All three of the SV's sensors properly detect that SV1 is following AdjV1, while AdjV2 is following SV1. Further visualizations are shared in the Data Validation section.



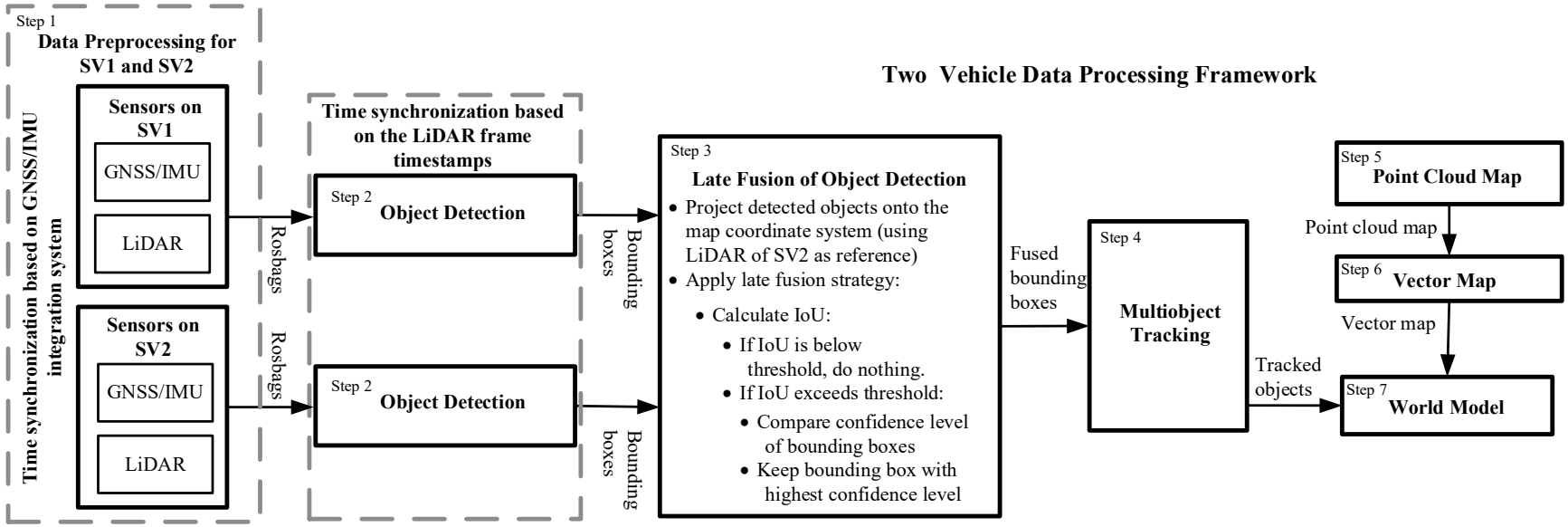
Source: FHWA.

Figure 18. Screenshot. Visualization of front and rear video data, vector map, and LiDAR point cloud for one-vehicle datasets in freeway scenario.

DATA-PROCESSING PIPELINE FOR TWO-VEHICLE DATASETS

This section discusses how the team adapted the data-processing pipeline for one-vehicle datasets to process the 24 h of data collected using two SVs simultaneously.

Figure 19 shows the data-processing pipeline for the two-vehicle datasets. This process is similar to the data processing pipeline for one-vehicle datasets (shown in figure 11). The primary differences are step 1 (Data Preprocessing for SV1 and SV2) and step 3 (Late Fusion of Object Detection). The reader is referred to the Data Processing Pipeline for One-Vehicle Datasets section for technical details on step 2 and steps 4–7 of the Data Processing Pipeline for Two-Vehicle Datasets.



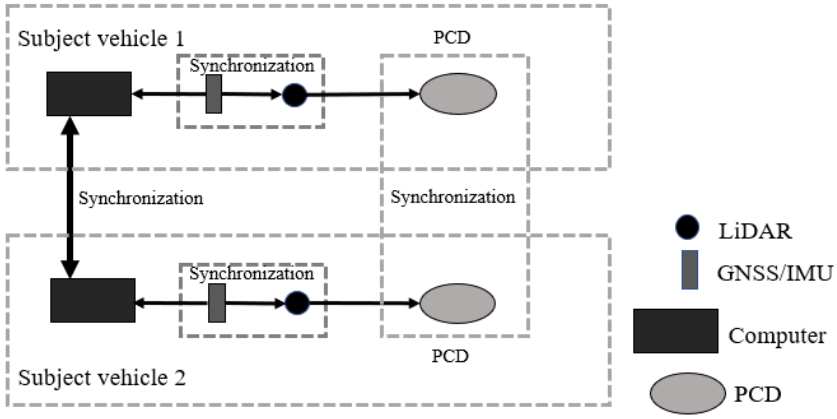
Source: FHWA.

Figure 19. Flowchart: Data-processing pipeline for two-vehicle datasets.

Step 1: Data Preprocessing for Two SVs Datasets

For the two-vehicle datasets, the project team collected both the sensor data from SV1 and SV2. Thus, step 1—data preprocessing for the two-SV datasets—is slightly different than the process used to preprocess the one-vehicle datasets. During the experiment, the project team used two separate computers on SV1 and SV2 to run the sensors and collect the sensor data. Because the computers run on different vehicles, clock synchronization errors can occur, even if the project team synchronizes the data before it is used for processing. Therefore, the project team used the timestamp of the GNSS/IMU integration systems on each vehicle, which are set to coordinated universal time (UTC), to synchronize the datasets from the two vehicles.

As shown in figure 20, the project team first synchronized the computers on each vehicle to one another based on the timestamps of the GNSS/IMU modules. The LiDAR data from each vehicle were also synchronized to the GNSS/IMU module on the same vehicle. In this way, all the data were synchronized with the GNSS/IMU timestamp (UTC). After processing the LiDAR point cloud data (PCD) from each vehicle, the team used the timestamp of the LiDAR frame to match the LiDAR frames from the two sensors at the same time to address the time delay due to the data processing.

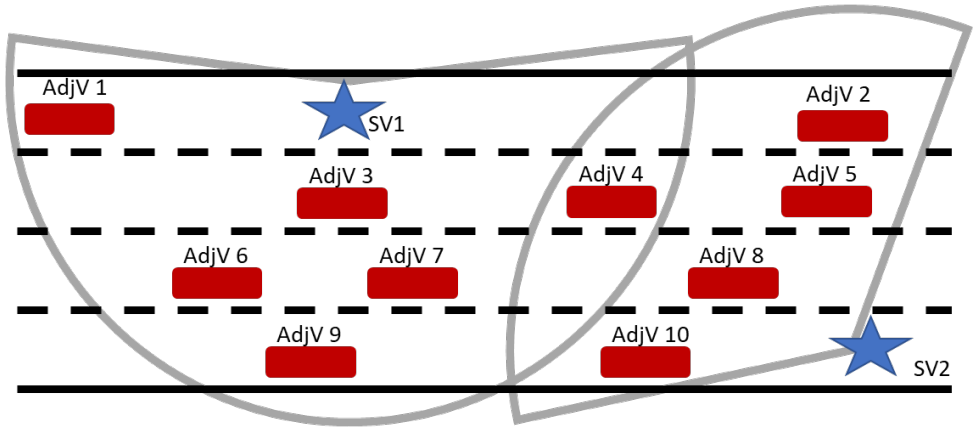


Source: FHWA.

Figure 20. Flowchart. Two-vehicle deployment data preprocessing.

Step 2 of the two-vehicle data-processing pipeline—object detection—is identical to the methodology applied in the one-vehicle data-processing pipeline. However, for the two-vehicle data-processing pipeline (figure 20), the project team collected the sensor data from both SV1 and SV2. This enabled fusion of the data from the two vehicles to perform cooperative perception to detect the objects. This not only extends the sensing range of each SV but also addresses the occlusion problem illustrated in figure 21. SV1 can perceive AdjV1, AdjV3, AdjV4, AdjV6, AdjV7, and AdjV9 because these vehicles are located within the sensor detection range of SV1. SV2 can perceive AdjV2, AdjV4, AdjV5, AdjV8, and AdjV10 because these vehicles are located within the sensor detection range of SV2. AdjV4 can be perceived by both vehicles because it is in the sensor detection range of both vehicles. All detected vehicles are included in the output CSVs. However, because some vehicles are detected by both SVs, an

additional step is required in the Data Processing Pipeline: step 3—late fusion of the objects detected by both SVs.



Source: FHWA.

Figure 21. Illustration. Data processing overview for two-vehicle datasets.

Step 3: Late Fusion Strategy

The use of both SVs complicates the data-processing pipeline because objects may be detected by one or both SVs. Thus, after step 2 (in which the object-detection algorithm also uses the Apollo CNN segmentation and object tracking of the SV processing pipeline), the bounding boxes must be fused for consistency.[20,21] The late fusion of object detection includes two parts: projecting objects onto one common coordinate system and applying the late fusion strategy.[34]

Step 3.1: Projecting Objects into One Common Coordinate

In step 2, the Apollo CNN segmentation algorithm is applied to produce bounding boxes for all objects detected by each of the two SVs. If an object is detected by both SVs, the bounding boxes must be fused together. To fuse the detected objects, first, all bounding boxes created using the LiDAR data collected by each vehicle were temporarily synchronized according to the timestamp of the LiDAR frames and then projected onto a common coordinate system. The selected coordinate system is the map coordinate system, where the *x*, *y*, and *z* are in the east, north and upward directions, respectively. The project team chose the LiDAR on SV2 as the reference data. However, no difference exists if one selects either the coordinate in SV1 or SV2 as the common coordinate.

The projection from SV1 to SV2 can be computed through (equation 9).[34]

$$T_{SV1}^{SV2} = {T_{SV2}^{m}}^{-1} T_{SV1}^{m} \quad (9)$$

Where:

$\boldsymbol{T}^{m}_{SV1}$ = the transformation from the SV1 coordinate to the map coordinate.

$\boldsymbol{T}^{SV2}_{SV1}$ = the transformation from SV1 to SV2.

$\boldsymbol{T}^{m}_{SV2}$ = the transformation from the SV2 coordinate to the map coordinate defined in equation 10.

$$\boldsymbol{T}^{\boldsymbol{m}}_{\boldsymbol{SV2}} = \begin{bmatrix} \boldsymbol{R}^{\boldsymbol{m}}_{\boldsymbol{SV2}} & \boldsymbol{t}^{\boldsymbol{m}}_{\boldsymbol{SV2}} \\ \boldsymbol{0} & \boldsymbol{1} \end{bmatrix} \tag{10}$$

Where:

m = the map coordinate.

$SV2$ = the SV2 coordinate.

$\boldsymbol{R}^{m}_{SV2}$ = the rotation matrix between SV2 and map coordinates, $\boldsymbol{R}$ is defined in equation 4.

$\boldsymbol{t}^{m}_{SV2}$ = translation vector defined in equation 11.

$$\boldsymbol{t}^{\boldsymbol{m}}_{\boldsymbol{SV2}} = [\boldsymbol{t_x}, \boldsymbol{t_y}, \boldsymbol{t_z}]^{\top} \tag{11}$$

t_x, t_y and t_z are the translation in the x, y, and z directions, respectively. The angles including yaw, pitch, and roll in the rotation matrix can be accessed from the GNSS/IMU integration system on the vehicle. The translation vector $\boldsymbol{t}^{m}_{SV2}$ is obtained by converting the longitude, latitude, and altitude from the GNSS/IMU integration system into x, y, and z in a Universal Transverse Mercator (UTM) coordinate.

Step 3.2: Late Fusion Strategy

When the two SVs are near each other, an intersection area exists where the detected objects' bounding boxes are predicted by the object-detection algorithms (step 2) using the LiDAR data collected by both SVs. An example of this is shown in figure 21, where AdjV4 is detected by both SVs.

In step 3.2, the bounding boxes from the object-detection algorithms applied to the data collected by SV1 and SV2 are fused together. To fuse the bounding boxes, first, the intersection of unit (IoU) between each bounding box and other bounding boxes is computed. The IoU is defined as the overlap area of the area of union.[35] In the late fusion strategy, if the IoU is below a threshold, nothing happens (indicating an object is only detected by SV1 or SV2) and the bounding box from step 2 is maintained. However, if the IoU is above a threshold, the confidence level (predicted by the object-detection algorithm) of the bounding boxes is compared between the object-detection algorithms applied to the data collected by SV1 and SV2. The bounding box with the higher confidence level is maintained for further processing. This effort used 0.1 as the IoU threshold.

This process produces fused bounding boxes, which are passed to step 4 of the data processing pipeline for two-vehicle datasets. The remaining steps of the data processing pipeline are identical to the one-vehicle data-processing pipeline.

OUTPUT OF TWO-VEHICLE DATASETS

This section discusses both the format of the data produced by the two-vehicle data processing pipeline and provides some sample graphics that demonstrate how well the pipeline can detect and track objects (i.e., AdjVs).

Output Description

Data-processing pipeline outputs include the position, velocity, acceleration, and orientation of objects of interest in difference reference frames. These data are all stored in CSV files. The contents of the CSVs produced by the data-processing pipeline in figure 19 are described in the following subsections.

Variables for AdjVs

The variables for the AdjVs are described in table 11. The variables are presented in different frames as shown in figure 8.

Table 11. Variables for the AdjVs (two-vehicle dataset).

Variable	Description	Column in CSV	Unit	Frame	Access method
ID	Identification number of the AdjVs (ascending by time of entry into the sensor range of the SV)	A	n/a	n/a	Calculated
Time	Timestamp (ascending by start time) of the corresponding row in CSV	B	s	n/a	Calculated
distance_adjv (headway)[1]	Distance between the center of the AdjV and SV	C	m	n/a	Calculated
pos_x_adjv_f	AdjV *x* position	D	m	Frenet	Calculated
pos_y_adjv_f	AdjV *y* position	E	m	Frenet	Calculated
pos_x_adjv_m	AdjV *x* position	F	m	Map	Calculated
pos_y_adjv_m	AdjV *y* position	G	m	Map	Calculated
heading_adjv_m	AdjV heading angle	H	degree	Map	Calculated
dim_x_adjv[2]	AdjV length	I	m	Vehicle	Calculated
dim_y_adjv[2]	AdjV width	J	m	Vehicle	Calculated
dim_z_adjv[2]	AdjV height	K	m	Vehicle	Calculated

Variable	Description	Column in CSV	Unit	Frame	Access method
speed_adjv	AdjV speed	L	m/s	Frenet/ map	Calculated
acc_adjv	AdjV acceleration	M	m/s^2	Frenet	Calculated
Closest_distance _longitudinal (gap)[1,3]	The closest distance between AdjV and SV1 in the longitudinal direction	AH	m	Frenet	Calculated
Closest_distance _lateral[4]	The closest distance between AdjV and SV1 in the longitudinal direction	AI	m	Frenet	Calculated
lanelet_id_adjv[5]	Lanelet ID of the AdjV's center point	AQ	n/a	n/a	Calculated
lane_id_adjv[5]	Lane ID of the AdjV's center point	AR	n/a	n/a	Calculated
total_lanes	Total lanes at the current position	AW	n/a	n/a	Calculated

[1]As shown in figure 16, the headway between two vehicles (measured between the same point on the leader-follower pair) is the distance_adjv. The distance_adjv is measured from the centroid of the leading vehicle to the centroid of the following vehicle. The gap between vehicles (measured between the rear bumper of the leading vehicle and the front bumper of the following vehicle) is recorded as the closest_longitudinal_gap.

[2]The research team set fixed values of the bounding boxes to prevent the variation of the bounding box for the same object over different frames.

[3]As shown in figure 16, when the AdjV is ahead of the SV, the closest longitudinal distance is the distance from the front bumper of the SV to the rear bumper of the AdjV in the Frenet frame. When the AdjV is behind the SV, the closest longitudinal distance is the distance from the rear bumper of the SV to the front bumper of the AdjV in the Frenet frame.

[4]As shown in figure 16, when the AdjV is on the right side of the SV, the closest lateral distance is the distance from the right doors of the SV to the left doors of the AdjV in the Frenet frame. When the AdjV is on the left side of SV, the closest lateral distance is the distance from the left doors of the SV to the right doors of the AdjV in the Frenet frame.

[5]As shown in figure 17, in Frenet coordinates, the road with four lanes is divided into lanelets from lanelet 1 to lanelet 12. The lanelet ID shows the exact ID where the vehicle is. The lane ID shows which lane the vehicle is. The lanelet ID and lane ID are based on the vector map information. The lane ID starts from the right side of the road and increases to the left side of the road. For instance, the lanelet ID for the SV is 1 and the lanelet ID for the AdjV is 6. For SV, its lane ID is 4 and the total number of lanes is 4. The lane ID for the AdjV is 3.

Variables for SVs

The variables for SV1 are described in table 12, and the variables for SV2 are described in table 13.

Table 12. Variables for SV1.

Variable	Description	Column in CSV	Unit	Frame	Access method
Time	Timestamp (ascending by start time) of the corresponding row in CSV	B	s	n/a	Measured
pos_x_sv1_f[2,3]	SV1 *x* position	N	m	Frenet	Calculated
pos_y_sv1_f[2,3]	SV1 *y* position	O	m	Frenet	Calculated
pos_x_sv1_m	SV1 *x* position	P	m	Map	Measured
pos_y_sv1_m	SV1 *y* position	Q	m	Map	Measured
heading_sv1	SV1 heading angle	R	degree	Map	Measured
dim_x_sv1	SV1 length	S	m	Vehicle	Measured
dim_y_sv1	SV1 width	T	m	Vehicle	Measured
dim_z_sv1	SV1 height	U	m	Vehicle	Measured
speed_sv1[2,3]	SV1 speed	V	m/s	Frenet/map	Measured
acc_sv1[2,3]	SV1 acceleration	W	m/s^2	Frenet	Calculated
lanelet_id_sv1[1]	Lanelet ID of the SV1's center point	AS	n/a	n/a	Calculated
lane_id_sv1[1]	Lane ID of the SV1's center point	AT	n/a	n/a	Calculated

[1]As shown in figure 17, in Frenet coordinates, the road with four lanes is divided into lanelets from lanelet 1 to lanelet 12. The lanelet ID shows the exact ID where the vehicle is. The lane ID shows which lane the vehicle is. The lanelet ID and lane ID are based on the vector map information. The lane ID starts from the right side of the road and increases to the left side of the road. For instance, the lanelet ID for the SV is 1 and the lanelet ID for the AdjV is 6. For SV, its lane ID is 4 and the total number of lanes is 4. The lane ID for the AdjV is 3.

[2]A result of interpolating data and running it through the processing method detailed in chapter 3 is that multiple SV positions, velocities, and accelerations may exist for the same timestamp when the SV encounters multiple AdjVs, The SV's position (Frenet), velocity, and acceleration are all outputs of the processing method, and this can cause very minimal differences in these variables.

[3]If a future user of the data plots the SV's processed trajectories, these trajectories will appear discontinuous. This discontinuity is not because missing data is missing but is due to the SV's position only being provided if an AdjV is detected. This decision was made to simplify how data was stored (every row is an instance where an AdjV is detected near the SV). If an AdjV is not detected, one can assume that the SV will continue to operate at its desired speed until a new AdjV is detected and impedes the SV's path.

Table 13. Variables for SV2.

Variable[4]	Description	Column in CSV	Unit	Frame	Access method
Time	Timestamp (ascending by start time) of the corresponding row in CSV	B	s	n/a	Measured
po_x_sv2_f[2,3]	SV2 *x* position	X	m	Frenet	Calculated
pos_y_sv2_f[2,3]	SV2 *y* position	Y	m	Frenet	Calculated
pos_x_sv2_m	SV2 *x* position	Z	m	Map	Measured
pos_y_sv2_m	SV2 *y* position	AA	m	Map	Measured
heading_sv2	SV2 heading angle	AB	degree	Map	Measured
dim_x_sv2	SV2 length	AC	m	Vehicle	Measured
dim_y_sv2	SV2 width	AD	m	Vehicle	Measured
dim_z_sv2	SV2 height	AE	m	Vehicle	Measured
speed_sv2[2,3]	SV2 speed	AF	m/s	Frenet/ map	Measured
acc_sv2[2,3]	SV2 acceleration	AG	m/s^2	Frenet	Calculated
lanelet_id_sv2[1]	Lanelet ID of the SV2's center point	AU	n/a	n/a	Calculated
lane_id_sv2[1]	Lane ID of the SV2's center point	AV	n/a	n/a	Calculated

[1]As shown in figure 17, in Frenet coordinates, the road with four lanes is divided into lanelets from lanelet 1 to lanelet 12. The lanelet ID shows the exact ID where the vehicle is. The lane ID shows which lane the vehicle is. The lanelet ID and lane ID are based on the vector map information. The lane ID starts from the right side of the road and increases to the left side of the road. For instance, the lanelet ID for the SV is 1 and the lanelet ID for the AdjV is 6. For SV, its lane ID is 4 and the total number of lanes is 4. The lane ID for the AdjV is 3.
[2]A result of interpolating data and running it through the processing method detailed in chapter 3 is that multiple SV positions, velocities, and accelerations may exist for the same timestamp when the SV encounters multiple AdjVs, The SV's position (Frenet), velocity, and acceleration are all outputs of the processing method, and this can cause very minimal differences in these variables.
[3]If a future user of the data plots the SV's processed trajectories, these trajectories will appear discontinuous. This discontinuity is not because missing data is missing but is due to the SV's position only being provided if an AdjV is detected. This decision was made to simplify how data was stored (every row is an instance where an AdjV is detected near the SV). If an AdjV is not detected, one can assume that the SV will continue to operate at its desired speed until a new AdjV is detected and impedes the SV's path.
[4]SV2 was an RI-ADAS-equipped vehicle but was operated as a level 0 nonautomated vehicle during data collection. Thus, although SV2 data is not considered reflective of ADAS-equipped vehicle behavior, it is reflective of a human-driven nonautomated vehicle. AdjVs interacting with SV2 may still exhibit behavioral changes due to the visibility of the sensor stack on the RI-ADAS.

Variables for Origins of the Map and Road

The variables for the road and maps are described in table 14.

Table 14. Information of map origin and road origin (two-vehicle dataset).

Variable	Description	Column in CSV	Unit	Frame
map_origin_x	map origin longitude	AJ	degree	ECEF
map_origin_y	map origin latitude	AK	degree	ECEF
map_origin_z	map origin altitude	AL	degree	ECEF
road_origin_x_m	map origin *x* position	AM	m	Map
road_origin_y_m	map origin *y* position	AN	m	Map
road_origin_x_ecef	road origin longitude	AO	degree	ECEF
road_origin_y_ecef	road origin latitude	AP	degree	ECEF

The variables for metadata are represented in table 15.

Table 15. Included metadata information (two-vehicle dataset).

Variable	Description	Column in CSV	Unit
Run number	Number from the set of processed runs	AX	n/a
Sub run number	Number of sub run for respective run number from processed dataset	AY	n/a
Date	Date of data collection	AZ	n/a
Time of day	Time stamp of data collection	BA	n/a
Sub run start time	Start time from the original run from where data was processed	BB	n/a
Route starting point (rs)	Google map start point[(15)]	BC	n/a
Route ending point (re)	Google map end point[(15)]	BD	n/a
Distance	Distance of the route	BE	miles
Maplink	Google maps link of the route[(15)]	BF	n/a
Annual traffic density	AADT for the route	BG	n/a
Roadway type	Type of roadway: limited access/divided/nondivided arterial	BH	n/a
Road condition	Condition of surface of road: wet/dry	BI	n/a
Speed limits	Speed limits along the route	BJ	mph
Type of vehicle	Mode of operation: RI/DI/baseline	BK	n/a
Aggressiveness	Aggressiveness setting for SV1	BL	n/a
Following distance	Following distance setting for SV1	BM	n/a
Special notes	Any interesting observations	BN	n/a
Gap level	Intended gap between SV1 and SV2 1: (30–60 m) or 2: (60–80 m)	BO	n/a

Output Visualization

Figure 22 shows a sample of data collected using the two SVs on a freeway. Figure 22-A is the view of Google Earth to visualize the freeway scenario, where the operation speed is approximately 112 km/h; the line overlaid on figure 22-A shows the trajectory of the SV, while the arrow shows the direction of travel.[(15)] Figure 22-B visualizes the vector map created during

step 6 of the data-processing pipeline (as shown in figure 19). As can be seen in figure 22-B, four lane lines (three lanes) exist in this freeway scenario.

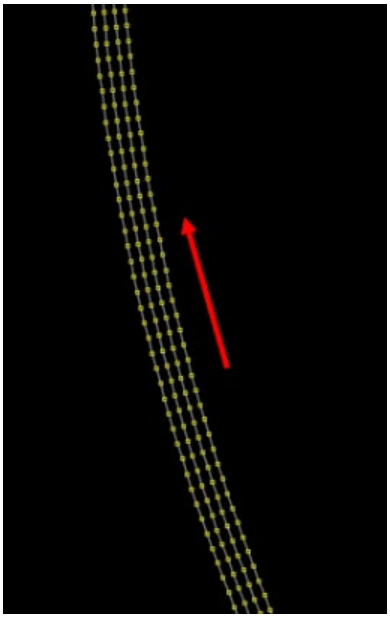
Figure 22-C provides the visualization of the object detection and tracking results created in step 4 of the data-processing pipeline in figure 19. As labeled on figure 22-C, the red point cloud is from SV1's LiDAR sensor, and the white point cloud is from SV2's LiDAR sensor. The AdjV bounding boxes are blue in figure 22-C and are labeled AdjV1, AdjV2, and AdjV3. SV1 (red) and SV2 (green) are both labeled. In figure 22-C, AdjV3 can only be sensed by SV2 vehicle sensors; AdjV2 can only be sensed by SV1 vehicle sensors; and AdjV1 can be sensed by both vehicles' sensors. By fusing the point clouds from the two SVs using the late fusion strategy (step 3 of the pipeline shown in figure 19), each SV's detection range can be extended; this process is also known as cooperative perception.

In the area where both SVs can detect AdjVs, AdjV1 is detected by both SVs, the object-detection results are fused, and only one bounding box is output, which validates the functionality of our late fusion strategy in terms of the object detection and tracking. The objects can be detected and tracked by the data-processing framework. Thanks to this cooperative perception, each SV's sensing area has been expanded, and more AdjVs can be detected for an individual SV.

The output of the data-processing framework provides the potential to analyze complex interactions between one SV and multiple AdjVs. For instance, in the lane where SV1 is located, SV1, AdjV1, and AdjV3 can be used to investigate the car-following behavior of both SV1 and AdjV1. If only using the LiDAR on SV1, AdjV3 cannot be detected as it is too far from SV1, and the sparsity issue of point cloud leads to failure of detection. Leveraging the LiDAR on SV2 provides the possibility to detect AdjV3 for SV1.

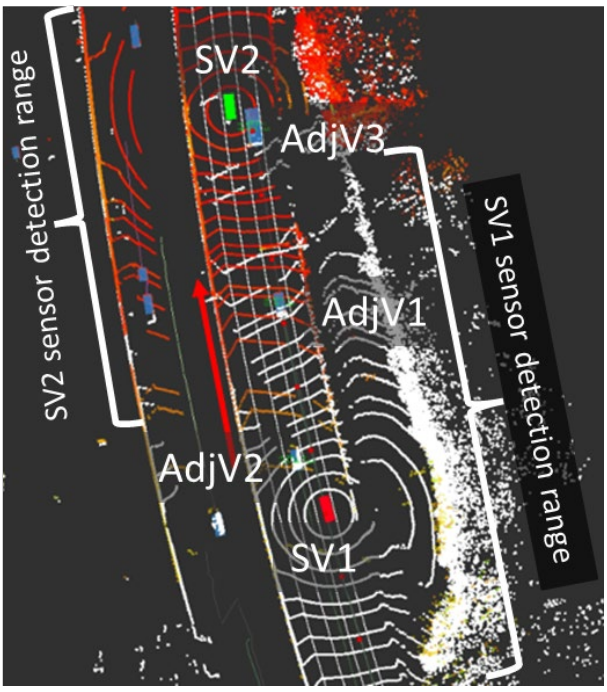
Original map: © Google® Earth™.
Modifications by FHWA (see Acknowledgments section).

A. Google Earth view of freeway (route map).[15]



Source: FHWA.

B. Vector map.



Source: FHWA.

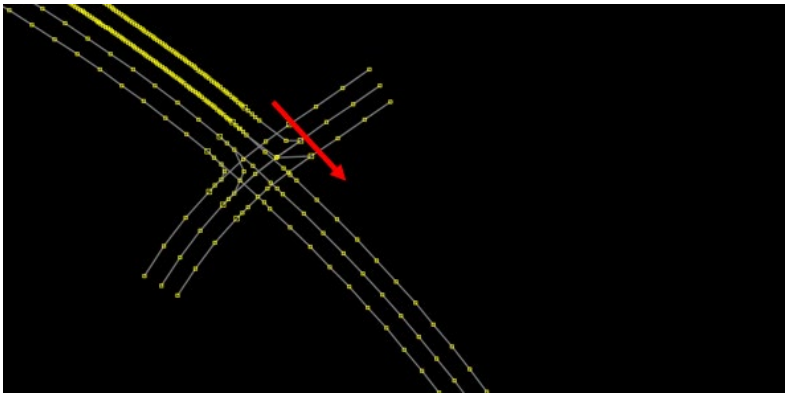
C. Point cloud map developed for object detection and tracking of LiDAR data.

Figure 22. Map. Visualization of test scenario, vector map, and object detection and tracking for a sample of the two-vehicle datasets in freeway scenario.

Figure 23 shows a sample of data collected using the two SVs in a city environment. Figure 23-A is the view of Google Earth to visualize the city road scenario; the line overlaid on figure 23-A shows the trajectory of the SV, while an arrow shows the direction of travel. Figure 23-B visualizes the vector map created for the city road scenario, which has two lanes in the direction of travel, created in step 6 of the data-processing pipeline shown in figure 19.[(15)] Figure 23-C visualizes the object-detection results created during step 4 of the data-processing pipeline for the city road scenario with two lanes. In figure 23-C, one can see that the data-processing pipeline can perceive the AdjVs around SV1 and SV2. In figure 23-C, SV2 (green) does not have any vehicles in its sensor detection range, while SV1 (red) is able to detect four AdjVs (blue). These AdjVs and SVs in two lanes can be used to analyze the car-following behavior and lane-change behavior.

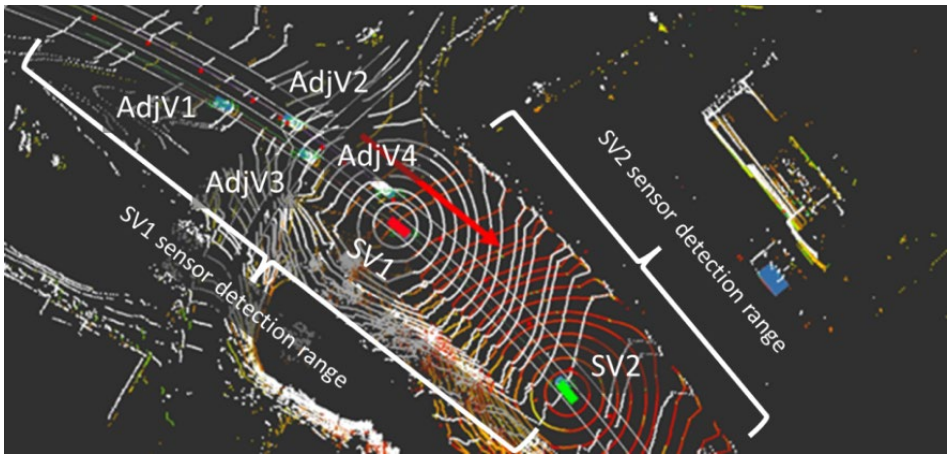
Original map: © 2022 Google® Earth™. Modifications by FHWA (see Acknowledgments section).

A. Google Earth view of intersection (route view).[15]



Source: FHWA.

B. Vector map.



Source: FHWA.

C. Point cloud map developed for object detection and tracking of LiDAR data.

Figure 23. Map. Visualization of test scenario, vector map, and object detection and tracking created for a sample of the two-vehicle datasets in city road scenario.

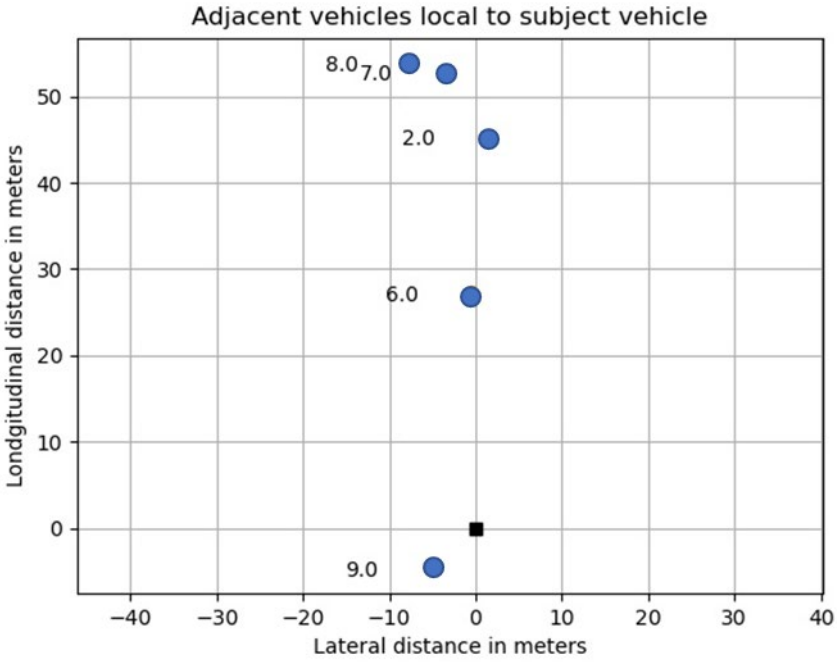
DATA VALIDATION

The primary outputs of the data-processing pipeline include the detection of an object (i.e., AdjV), the relative distance between the object and the SV, the speed that the object is moving, and the acceleration of the object. This section discusses how the project team validated the outputs of the data-processing pipeline.

Validation of Detection of an AdjV

First, the project team evaluated the processed data for consistency with the collected raw dataset. The team completed the validation by finding the location of the SV, number of AdjVs, and the AdjVs' relative position from the SV from the processed dataset. The ground truth of the data was established from the LiDAR point cloud obtained from the raw data. Front and rear camera data were used to visually confirm the number of surrounding vehicles and their position.

Figure 24 graphs the lateral and longitudinal position of the AdjVs relative to the SVs by applying the data-processing pipeline to the raw data. The SV is represented in the plot by a square located at 0,0. AdjVs are represented on the plot by circles and labeled with the vehicle IDs. The plot provides information about the relative location of the AdjVs with respect to the SV. Figure 24 shows that four vehicles are within 55 m of the front of the SV and one vehicle within 10 m behind the SV.



Source: FHWA.

Figure 24. Graph. AdjV positions.

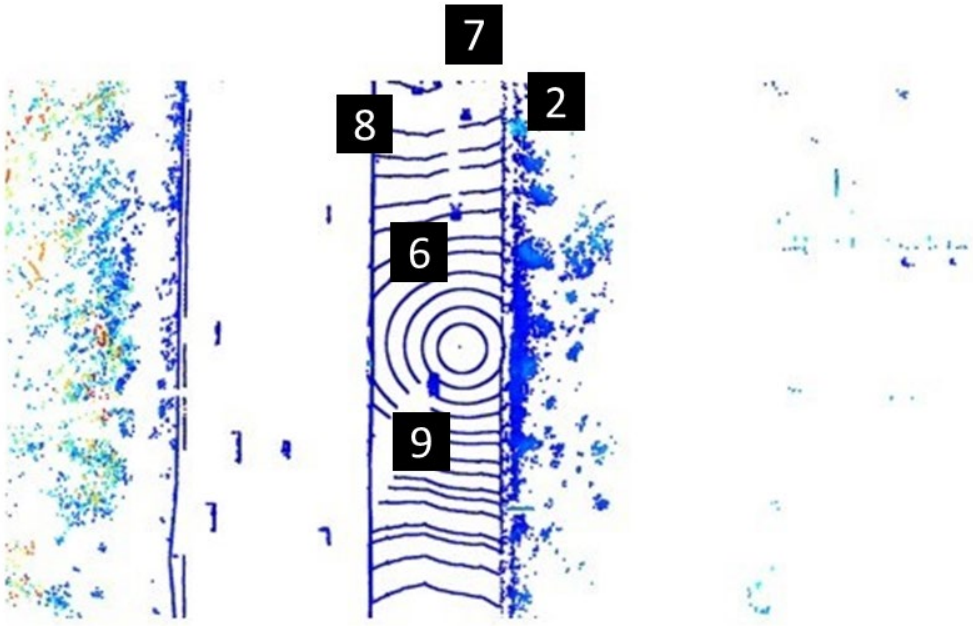
Figure 24 through figure 27 show how the position of the AdjVs relative to the SV can be validated using the forward-facing camera, rear-facing camera, and LiDAR point cloud, respectively.

Source: FHWA.

Figure 25. Photo. Front camera view.

Source: FHWA.

Figure 26. Photo. Rear camera view.



Source: FHWA.

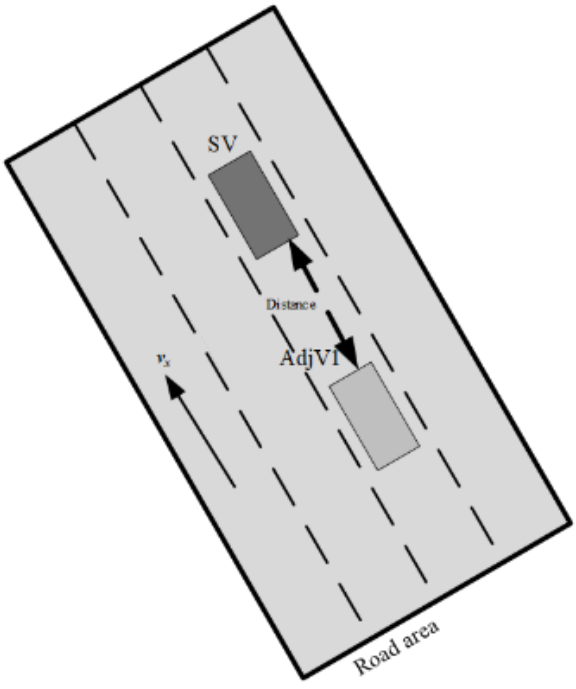
Figure 27. Illustration. Point cloud top view.

As shown in figure 25, figure 26, and figure 27, the object-detection module of the data processing pipeline can properly identify the vehicles in the camera data. These results indicate that the object-detection algorithm can properly identify AdjVs in traffic using the SV's LiDAR data.

Validation of AdjV Position

In sensor validation testing completed under previous projects, the project team demonstrated that the LiDAR model installed on both SVs has good performance, meaning that the objects around the SV can be detected properly when the distance between the LiDAR and objects is smaller than 35–45 m.[36] When the distance between the LiDAR and detected objects is larger than 45 m, the quality of the data degrades due to the sparsity of the LiDAR point cloud (i.e., properly identifying objects in the sparse PCD is more difficult). This section seeks to demonstrate that the data-processing pipeline performs well when using data collected by the LiDAR unit when the distance between the AdjVs and SVs is at or below 45 m (approximately six to eight car lengths away).

To verify the accuracy of the AdjV position produced by the data-processing pipeline, the project team conducted an experiment with SV1 and an AdjV under sunny weather conditions. The project team selected a case where one AdjV is following SV1 in a straight-line driving scenario at a gap of approximately 40 m. The project team selected this gap between the two vehicles as a worst-case scenario to validate the data-processing pipeline. The team was concerned that the sparse LiDAR PCD at large relative distances may lead to unstable bounding boxes and result in position error in the processed data. This experiment sought to demonstrate if the data-processing pipeline can successfully complete object detection with a sparse point cloud. The experimental design is shown in figure 28.



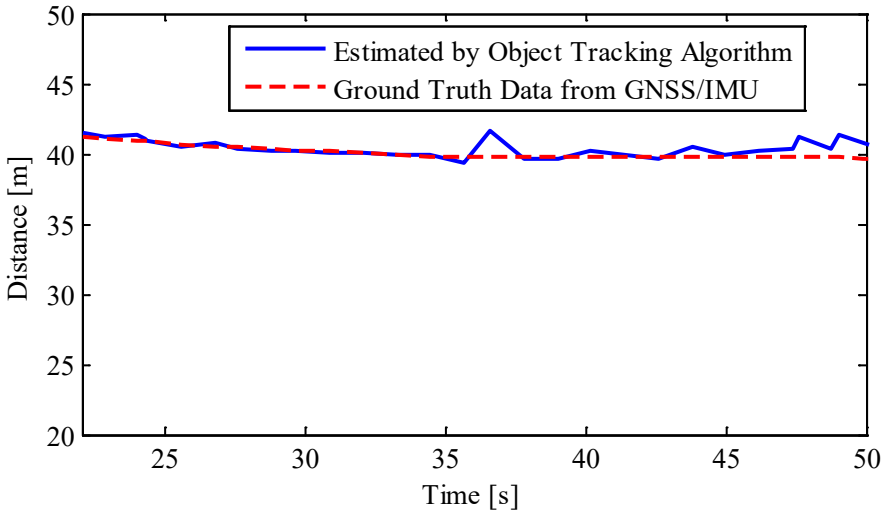
Source: FHWA.

Figure 28. Illustration. Trajectory verification scenario.

In the experiment, one of the AdjVs is equipped with the same sensors as SV1. The GNSS/IMU system on this AdjV is RTK-corrected, which provides the ground truth pose information.

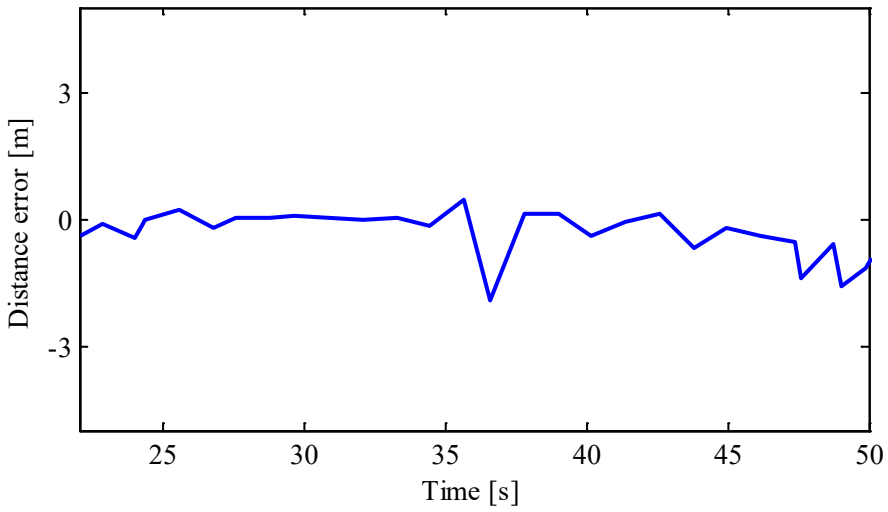
One can use the ground truth data collected using the GNSS/IMU systems to calculate the closest distance information. This distance can be directly compared to the "closest_distance_longitudinal" variable (gap between vehicles) from table 7 and table 11 calculated with the data-processing pipeline. When calculating the gap between the AdjV and SV1 to analyze the car-following behavior, the error in the objects' position will be transferred to the vehicle gap and accordingly affect the behavior analysis. In other words, the accuracy of the vehicle gap also reflects the accuracy of the SV's trajectory.

Figure 29 shows the results of the vehicle gap calculated using the data-processing pipeline between SV and AdjV1. In figure 29-A, the ground truth of the gap between the two vehicles is calculated from the RTK-aided GNSS/IMU system in both vehicles. This distance is represented by the dashed red line (ground truth) in figure 29-A. The vehicle gap computed by the position derived directly from the object tracking algorithm is shown in the solid blue line (Estimated) with small jumps. In figure 29-A, the car-following distance remains at around 40 m and the relative distance estimated by the data-processing pipeline follows the ground truth data reference well. Figure 29-B shows the relative distance error. Figure 29-B also shows that for most of the experiment, the error is within 1 m and the maximum vehicle gap error of the experiment is within 2 m. The results in figure 29 prove the effectiveness of the data-processing pipeline shown in figure 11 and figure 19.



Source: FHWA.

A. Vehicle gap between SV1 and AdjV1.



Source: FHWA.

B. Vehicle gap error.

Figure 29. Graphs. Relative distance between SV and AdjV and its error.

Note that when the vehicles are more than 45 m away from the SV, the position error may be more significant due to limitations with the LiDAR sensor detection range (only a few point clouds from the LiDAR sensor fall on the objects, and this may lead to the bounding boxes error) based on the project team's experience. However, the ultimate planned use of this dataset is to study the interactions between an ADAS-equipped SV and nonautomated AdjVs in traffic. Thus, AdjVs that are not sufficiently close to the SV (e.g., defined in this project as gaps between the subject and following vehicle more than 40 m, or six to eight car lengths away) are not necessarily of interest in car-following and lane-changing behavior analysis.

When the objects are near the SV (defined as located within 40 m), figure 29 confirms that the estimated vehicle gap ("closest_distance_longitudinal") error is smaller than 1 m. This meets the requirement of analyzing car-following or lane-change behavior and calibrating corresponding models. Thus, figure 29 validates the AdjV position and relative position data produced by the data-processing pipeline.

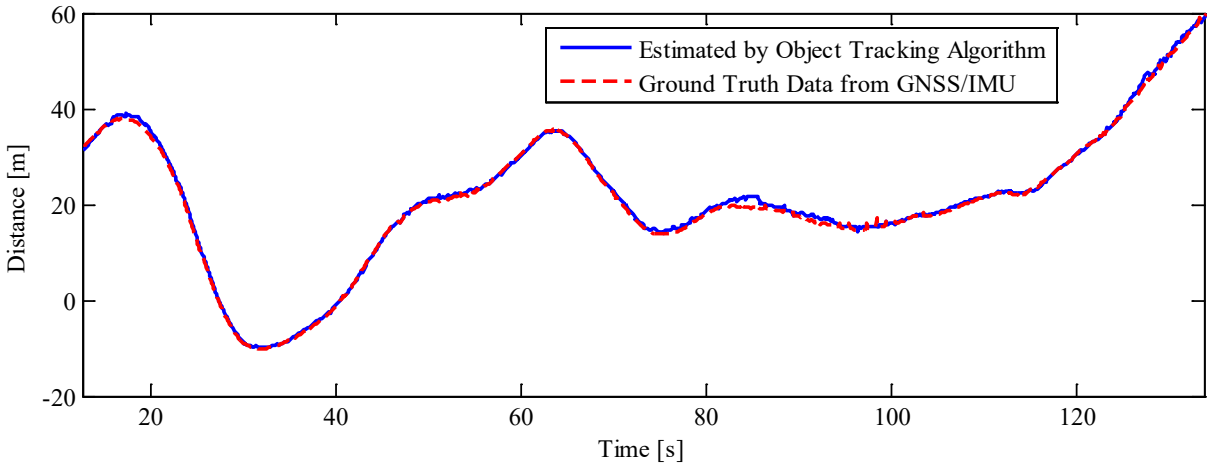
Validation of AdjV Speed and Acceleration

To rigorously validate the accuracy of the AdjV speed and acceleration, the project team conducted validation exercises based on a sample of two-vehicle datasets in both city road and highway conditions. In these scenarios, the ground truth data—including the distance headway (distance_adjv) between the SV and the AdjV and the speed and acceleration of the AdjV—were sourced from the RTK-aided GNSS/IMU systems installed on both SVs.

In this validation exercise, the research team deliberately chose instances where one SV was near the other SV. This arrangement allowed one SV to detect and track the other SV. In essence, the project team utilized one of the SVs as an AdjV that can be detected by its counterpart. This method enabled the team to acquire ground truth data for one AdjV (collected with the other SV)

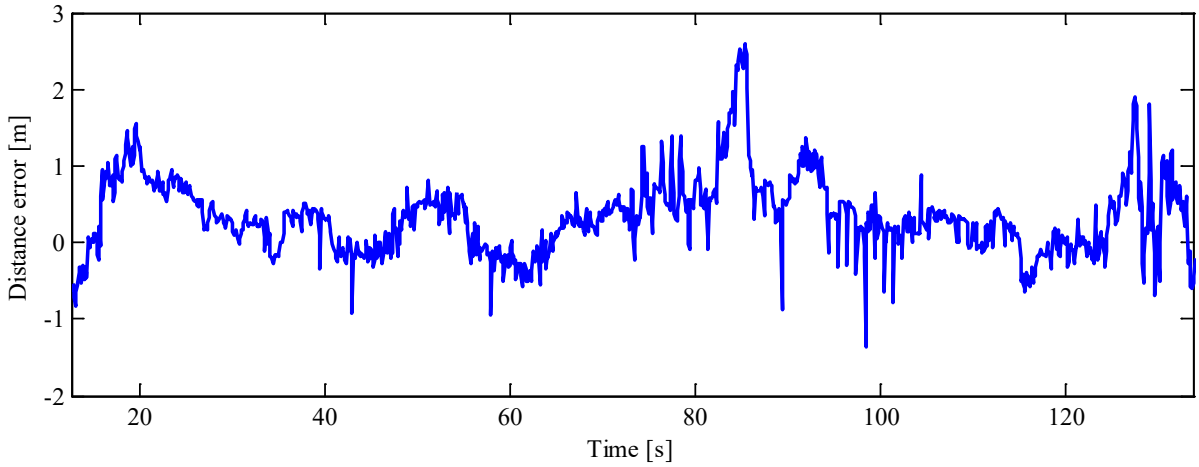
for validation purposes on real roads, differentiating this validation from the results in figure 29, obtained under proving-ground scenarios.

The results from city road and highway scenarios are graphically represented in figure 30 and figure 31, respectively. The mean error and standard deviation of headway, speed, and acceleration can be found in table 16 and table 17.



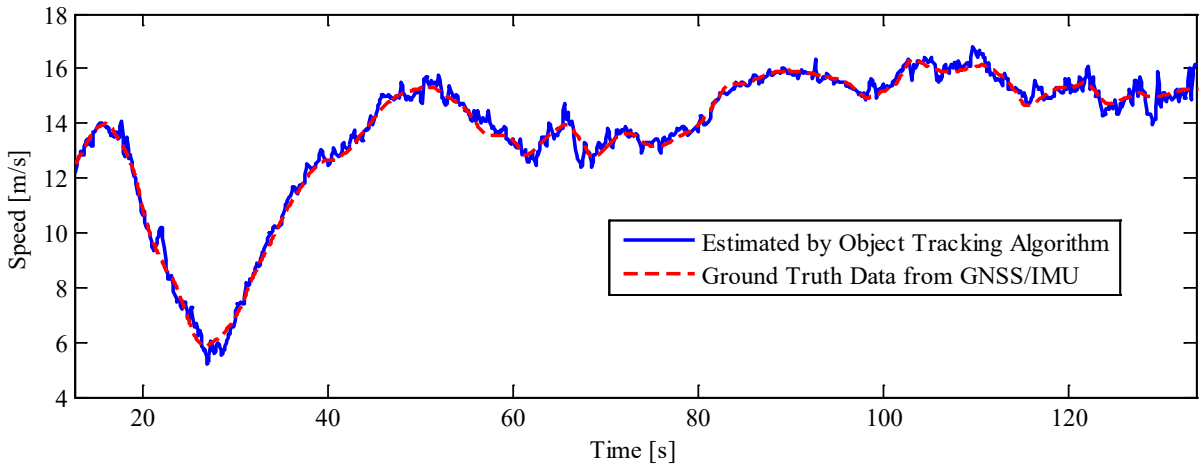
Source: FHWA.
Note: This distance goes negative because the vehicles are in different lanes and used the distance_adjv.

A. Distance headway between SV1 and SV2 (the instrumented AdjV that can be detected by SV1's sensors).



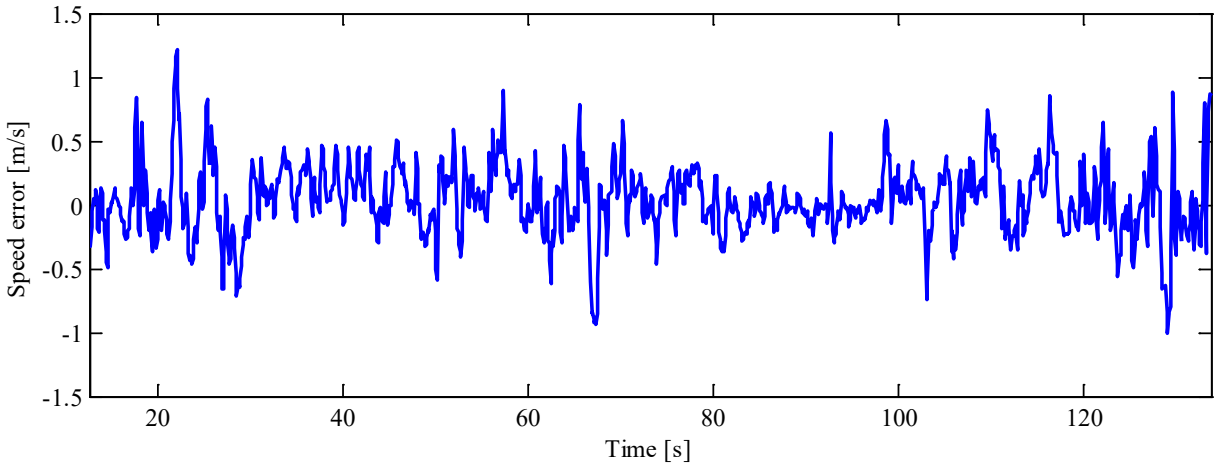
Source: FHWA.

B. Distance headway error.



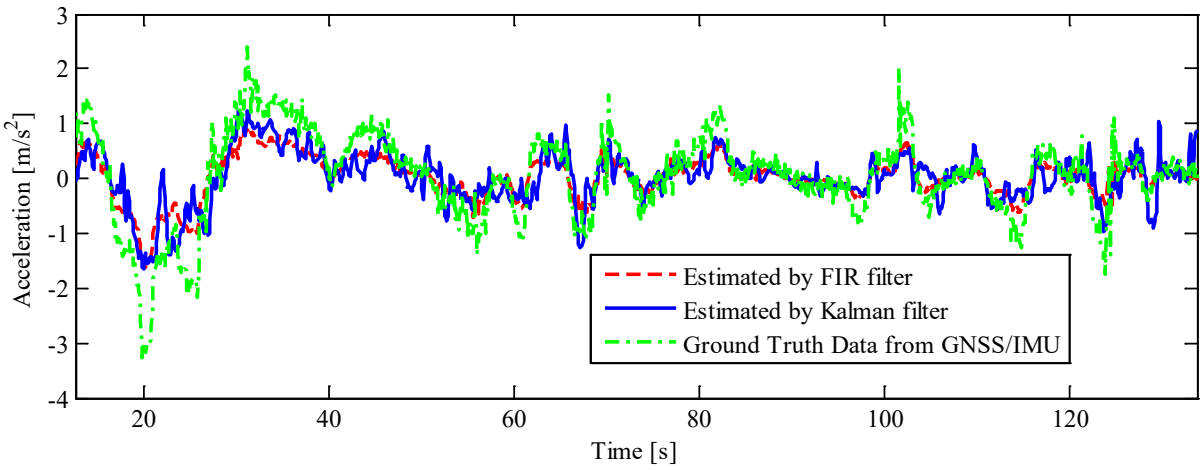
Source: FHWA.

C. Speed of SV2 (the instrumented AdjV that can be detected by SV1's sensors).



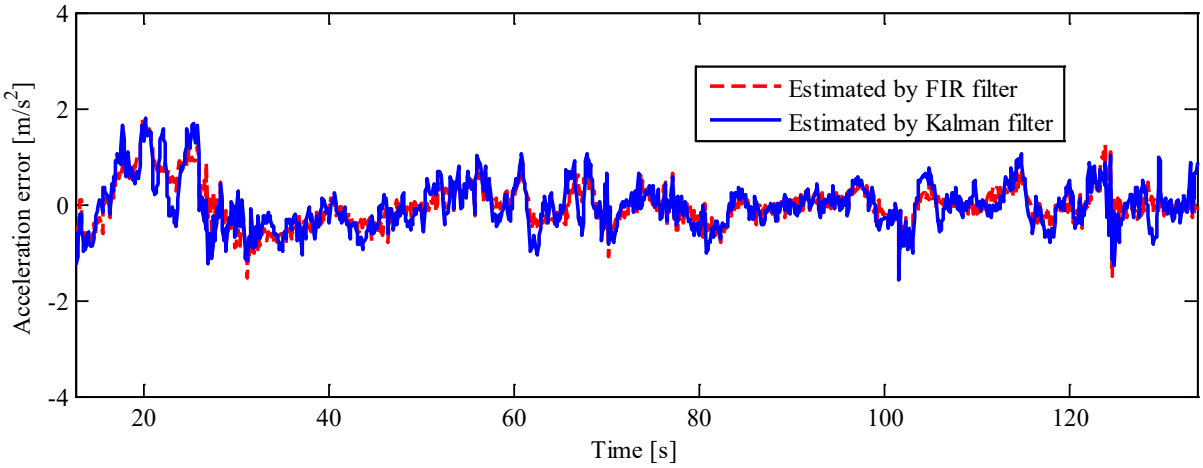
Source: FHWA.

D. Speed error of SV2 (the instrumented AdjV that can be detected by SV1's sensors).



Source: FHWA.

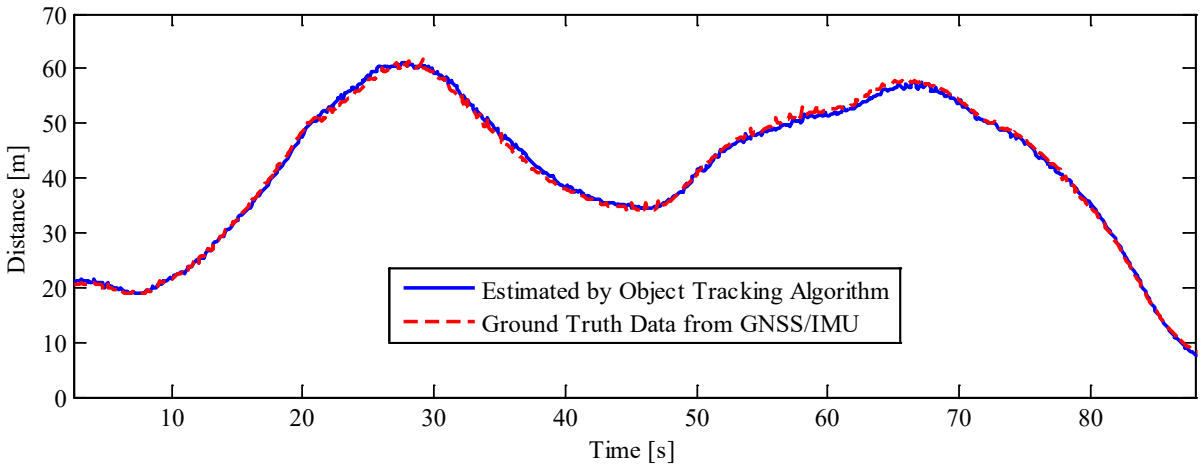
E. Acceleration of SV2 (the instrumented AdjV that can be detected by SV1's sensors).



Source: FHWA.

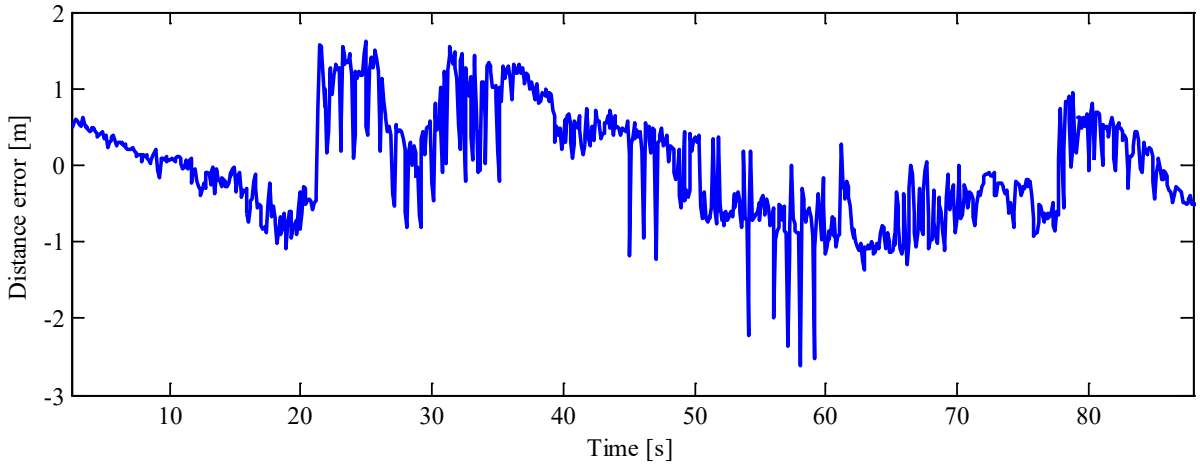
F. Acceleration error of SV2 (the instrumented AdjV that can be detected by SV1's sensors).

Figure 30. Graphs. Results of distance headway, speed, and acceleration between SV1 and SV2 (instrumented AdjV).



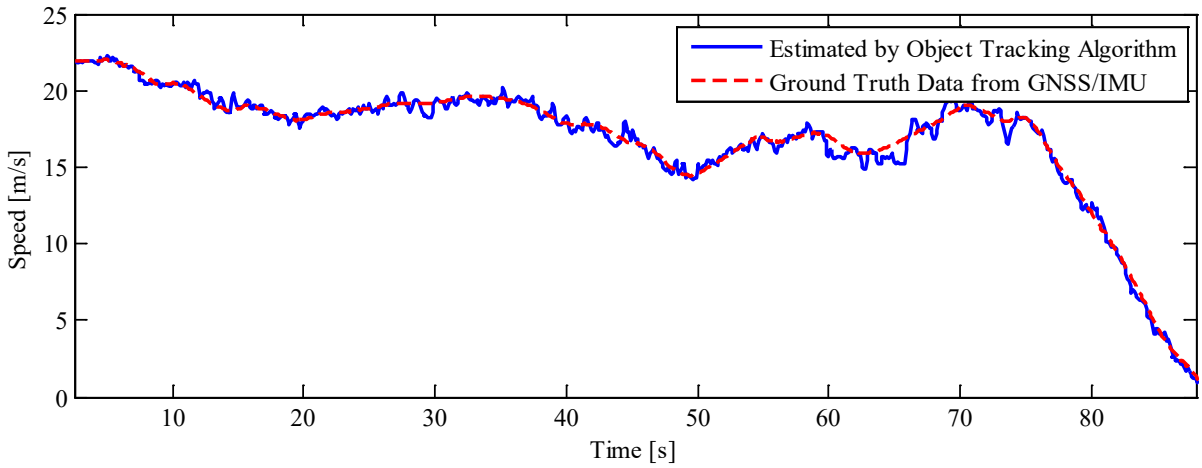
Source: FHWA.

A. Headway between SV1 and SV2 (the instrumented AdjV that can be detected by SV1's sensors).



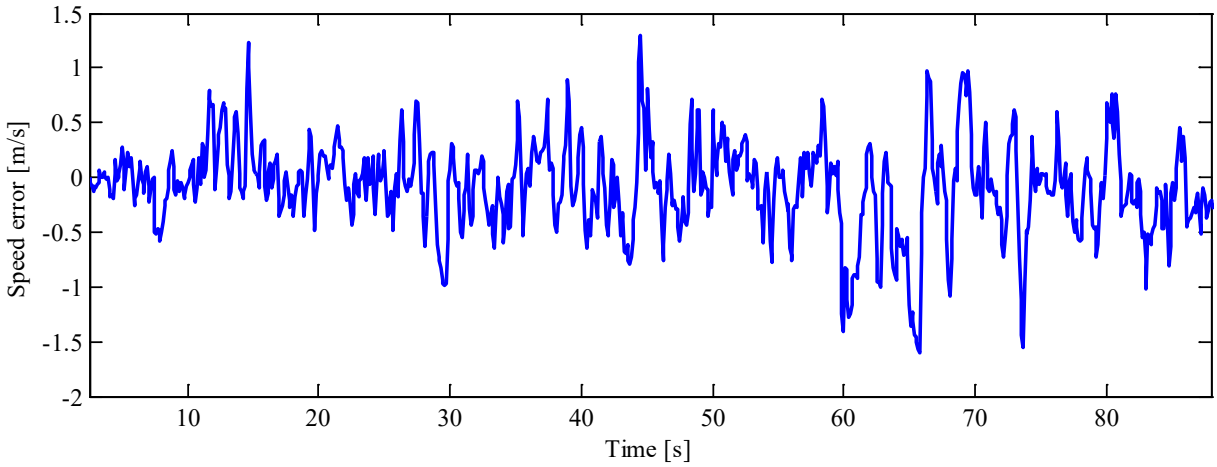
Source: FHWA.

B. Headway error.



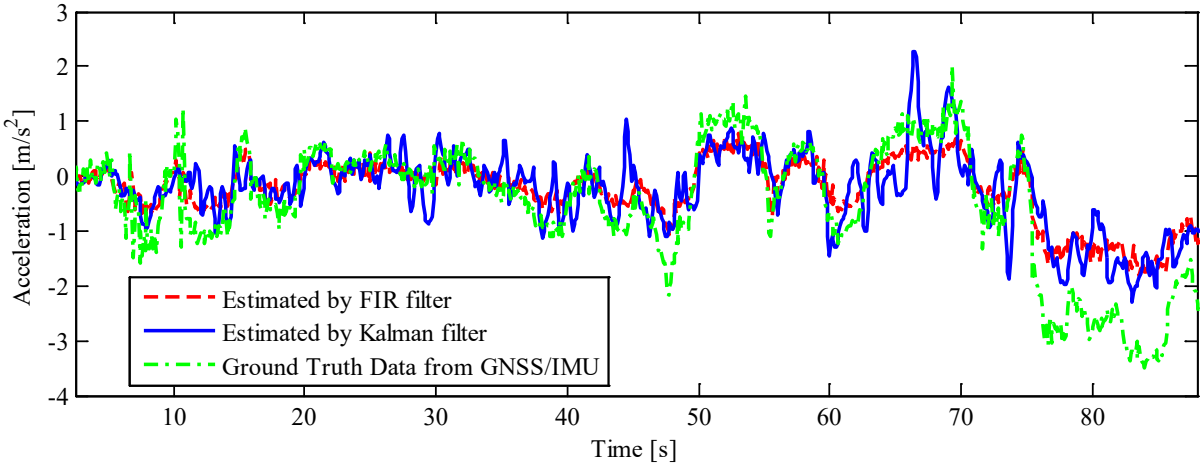
Source: FHWA.

C. Speed of SV2 (the instrumented AdjV that can be detected by SV1's sensors).



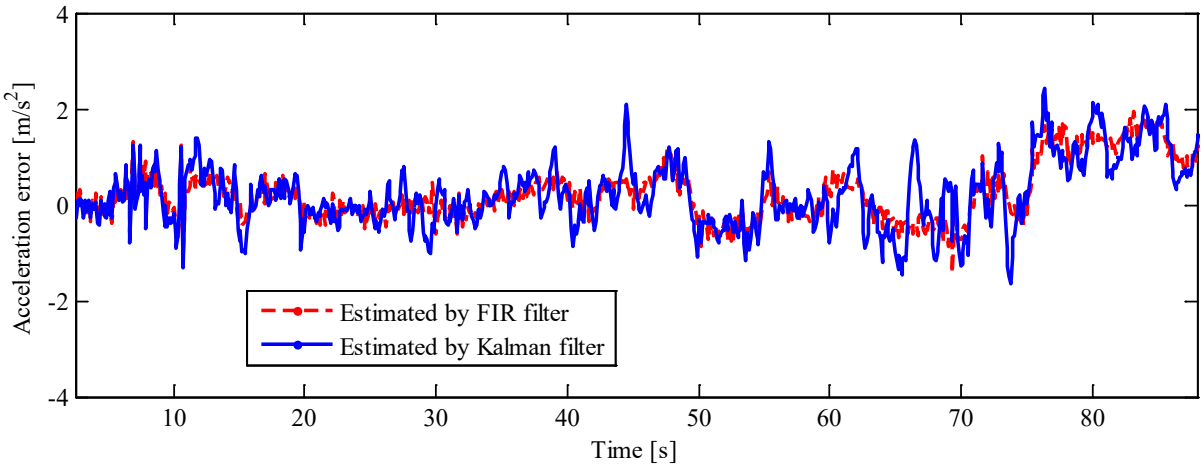
Source: FHWA.

D. Speed error of SV2 (the instrumented AdjV that can be detected by SV1's sensors).



Source: FHWA.

E. Acceleration of SV2 (the instrumented AdjV that can be detected by SV1's sensors).



Source: FHWA.

F. Acceleration error of SV2 (the instrumented AdjV that can be detected by SV1's sensors).

Figure 31. Graphs. Results of headway, speed, and acceleration between SV1 and the AdjV (SV2).

City Road Scenario

Figure 30-A was created by graphing the processed and ground truth distance headway (distance_adjv) between SV1 (operated as the SV) and SV2 (operated as the AdjV) data. The solid blue line is the distance headway between the two vehicles estimated by the data-processing algorithm, and the red dashed line is the ground truth headway data calculated by comparing the position of SV1 (obtained from the GNSS/IMU unit installed on SV1) and the position of SV2 (obtained from the GNSS/IMU unit installed on SV2). In this data sample, the headway between the two vehicles ranges between −1 and 60 m. Note that minus headway does not mean a collision but that the two vehicles are in different lanes (this distance goes negative because the vehicles are in different lanes and used the distance_adjv). Figure 30-A shows that the processed and ground truth headway data are very similar. Figure 30-B confirms that the error between the processed and ground truth headway data is mostly between ±1 m for the full data sample.

Figure 30-C was created by graphing the processed and ground truth speed data for the AdjV (SV2). The solid blue line is the estimated speed data calculated by the data-processing algorithm and the red dashed line is the ground truth AdjV speed data obtained from the GNSS/IMU unit installed on SV2. In this data sample, the AdjV's speed ranges between 0 and 20 m/s. Figure 30-C shows that the processed and ground truth speed data for the AdjV are very similar. Figure 30-D confirms that the error between the processed and ground truth speed data is mostly between ±1 m/s for the full data sample.

Finally, figure 30-E shows the processed AdjV acceleration data, derived using the ground truth velocity data (green dashed dotted line) and data-processing pipeline's estimated velocity data smoothed with either the Kalman filter (solid blue line) or the FIR filter (dashed red line). Figure

30-E demonstrates that the ground truth data collected using sensors on SV2 is sufficiently similar to the acceleration data estimated using the data processing pipeline. Figure 30-F confirms the error between the ground truth and processed acceleration data is small, within ±2 m/s^2.

Table 16. Mean error and standard deviation of headway, speed, and acceleration error (city scenario).

Metric	Mean Error	Standard Deviation
Distance headway error (m)	0.27	0.52
Speed error (m/s)	0.03	0.28
Acceleration FIR filter (m/s^2)	−0.06	0.45
Acceleration Kalman filter (m/s^2)	−0.05	0.50

These observations are further corroborated by the mean and standard deviation of headway, speed, and acceleration errors shown in table 16, which confirm the small error bounds described previously.

Highway Scenario

Figure 31-A was created by graphing the processed and ground truth distance headway between SV1 (operated as the SV) and SV2 (operated as the AdjV) data. The solid blue line is the distance headway between the two vehicles estimated by the data-processing algorithm, and the red dashed line is the ground truth headway data calculated by comparing the position of SV1 (obtained from the GNSS/IMU unit installed on SV1) and the position of SV2 (obtained from the GNSS/IMU unit installed on SV2). In this sample, the headway between the two vehicles ranges between 0 and 60 m. Figure 31-A shows that the processed and ground truth headway data are very similar. Figure 31-B confirms that the error between the processed and ground truth headway data is mostly between ±2 m for the full data sample.

Figure 31-C was created by graphing the processed and ground truth speed data for the AdjV (SV2). The solid blue line is the estimated speed data calculated by the data-processing algorithm, and the red dashed line is the ground truth AdjV speed data obtained from the GNSS/IMU unit installed on SV2. In this data sample, the speed of the AdjV ranges between 0 and 25 m/s. Figure 31-C shows that the processed and ground truth speed data for the AdjV are very similar. Figure 31-D confirms that the error between the processed and ground truth speed data is mostly between ±1 m/s for the full data sample.

Finally, figure 31-E shows the processed AdjV acceleration data, derived using the ground truth velocity data (green dashed dotted line) and data processing pipeline's estimated velocity data smoothed with either the Kalman filter (solid blue line) or the FIR filter (dashed red line). Figure 31-E demonstrates that the ground truth data collected using sensors on SV2 is sufficiently similar to the acceleration data estimated using the data processing pipeline. Figure 31-F confirms the error between the ground truth and processed acceleration data is small, within ±2 m/s^2.

Table 17. Mean error and standard deviation of headway, speed, and acceleration error (highway scenario).

Metric	Mean error	Standard deviation
Distance headway error (m)	−0.06	0.79
Speed error (m/s)	−0.03	0.32
Acceleration FIR filter (m/s^2)	0.02	0.61
Acceleration Kalman filter (m/s^2)	0.02	0.65

The mean and standard deviation of headway, speed, and acceleration errors, displayed in table 17, further affirm the above error bounds.

Key Takeaways from Data Validation Experiments

Based on the results presented in figure 30 and figure 31, the estimated speed and acceleration have been validated under both city road and highway scenarios, provided the headway ranges between 0–60 m, and the speed fluctuates between 0–23 m/s. The FIR filter demonstrates marginally superior performance than the Kalman filter, as evidenced by a smaller standard deviation error.

Despite figure 30 and figure 31 indicating that the AdjV can still be detected when the headway exceeds 45 m, the project team's experience suggests that detection and tracking can be impacted by minor occlusions due to the sparsity of objects. Consequently, the project team recommends that users primarily use datasets where the headway remains within 45 m for information reliability. Beyond this range, users should exercise caution regarding the accuracy of the vehicle detection (which impacts the AdjV position, speed, and acceleration data—all calculated using the data processing pipeline).

Important Notes About the Processed Data

This subsection attempts to address questions that have been raised by project teams working with the datasets before they were made publicly available.

- As discussed in step 6, the FIR filter used to filter the acceleration information requires proper initialization. In this case, if a high-speed AdjV is tracked by the SV, initializing the FIR filter properly is difficult, which will cause anomalies in the acceleration information. The project team did not artificially smooth these anomalies and suggests that future users pay attention to the anomalies and apply appropriate methods to process the acceleration or generate the acceleration based on the position and speed information provided in the processed CSVs.

- The data collected through this project is posted online as two separate datasets: single-vehicle and two-vehicle deployments. These datasets were separated because the two datasets have a different number of columns (the team could collect more data types using multiple vehicles).[2,3]

- If a future user of the data plots the processed trajectories, the trajectories will appear to be discontinuous due to missing data. However, the SV's position is only provided if an AdjV is detected. This decision was made to simplify how data was stored (every row is an instance where an AdjV is detected near the SV). If an AdjV is not detected, one can assume that the SV will continue to operate at its desired speed until a new AdjV is detected and impedes the SV's path.

- Explainable discrepancies exist from the distance, velocity, and acceleration in the processed dataset.

    - The distance_adjv and closest_distance_longitudinal in the processed dataset use slightly different references (vehicle centroids versus front to rear bumper, respectively). Please see table 7 and table 11. The use of different references will lead to a roughly fixed offset if comparing these two values.
    - Multiple velocities and accelerations can occur at the same time, the result of interpolating data from multiple AdjVs and running the data through the processing method detailed previously in the data processing pipeline section in this chapter. Because the SV's position, velocity, and acceleration are all outputs of the processing method, very minimal differences in these variables can occur. These differences can be considered negligible when using the data, or the average can be taken of the variables.

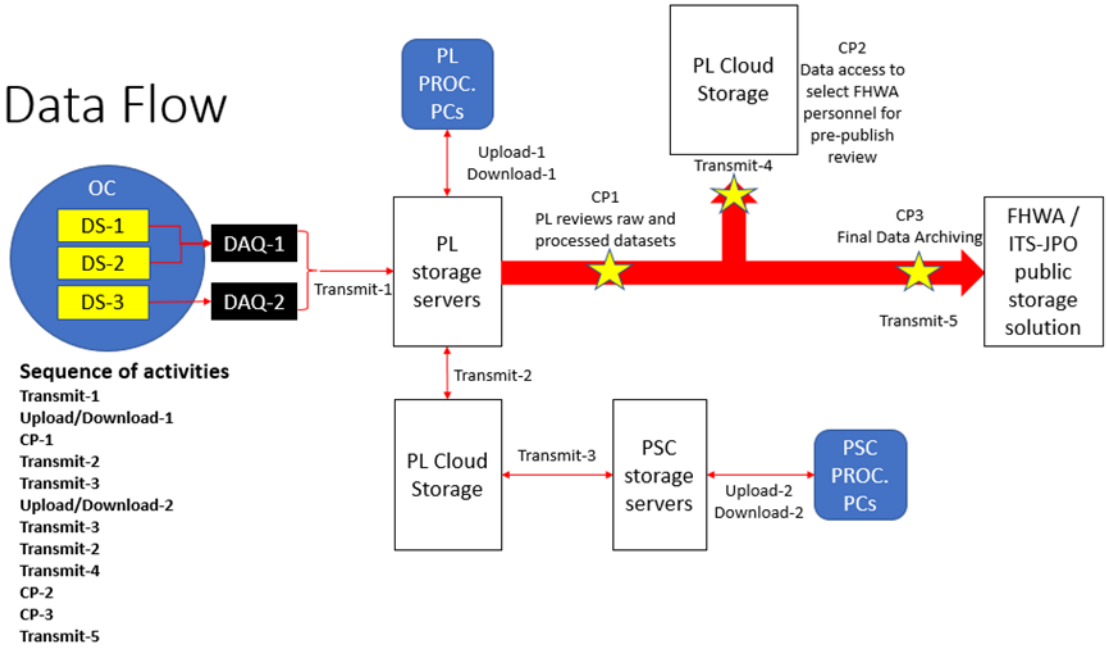
CHAPTER 4. DATA MANAGEMENT PLAN

This chapter outlines the project team's data management plan (DMP) for the project. The DMP outlines the technical and management approaches, including the following:

- Policies, technology, and mechanisms for capture, transmittal, storage, and archiving of data collected for the duration of the project.
- Appropriate organizations, tools, and mechanisms for preserving the integrity of collected datasets.

DATA FLOW

Figure 32 is graphical representation of the data flow overview that the project team used from data generation to final archiving. This process ensured that all appropriate data generators and recipients were able to access data in a timely and secure manner.



Source: FHWA.
CP = check point; DS = data source; OC = operating condition; PL = project lead; Proc. PCs = processing computers; PSC = project subcontractor; SFTP = secure file transfer protocol.

Figure 32. Flowchart: Data flow overviews.

From the data flow perspective, data sources and collection methods are generalized and technologically agnostic. Data flow started at data generation when a data source (DS) was captured through a DAQ device during a testing activity for any operating condition (OC); the DAQ devices and the testing activities during specified operating conditions are described in chapter 2. An in-field data check was performed to ensure that data was appropriately captured. This initial data checkpoint was an in-field data verification to ensure that data collection devices

were appropriately recording information and that data sources appeared reasonable in magnitude, frequency, and expected behavior.

Once data was captured through the appropriate DAQ device, the project team completed Transmit-1. Transmit-1 is the transfer of data from any DAQ device to the project lead's (PL) storage servers. Once on the PL storage servers, the information was downloaded (Download-1) onto PL's processing computers, where the data was checked to ensure consistency and initial exporting of the data to CSV began. The validation check, along with human spot check verification, was an automated data verification process to ensure that data was appropriately captured during any testing activity. When initial export activities were completed, the data in the form of raw CSVs and rosbags were uploaded back to the PL storage servers through Upload-1.

From the PL storage servers, data were transferred through Transmit-2 to the PL cloud storage. Once raw data was uploaded to the PL cloud storage, the raw data was transferred through Transmit-3 to the project subcontractor (PSC) storage server.

From this server, the PSC downloaded data (Download-2) for postprocessing according to the data-processing pipeline described in chapter 3. Once completed, postprocessed trajectory data was uploaded to the PSC storage servers (Upload-2), transferred back to the PL cloud storage through Transmit-3, and transferred from the PL cloud storage through Transmit-2 to the PL storage servers. At this point, PL completed the data validation checks described in chapter 3 and finalized data through checkpoint 1.

With the data finalized, information was transferred through Transmit-4 and Transmit-5. Transmit-4 is the transfer of data from PL to the Federal Highway Administration (FHWA) through a cloud storage service for interim data use during the project. Transmit-5 is the end of project transmission from PL to the final data archiving sites. Transmit-5 includes the transmission of the final processed data to the ITS Joint Program Office's ITS DataHub and transferring the raw and final processed data to the Office of Safety and Operation Research and Development's Multidisciplinary Data Management System (MDMS) for internal use by FHWA.[37]

The Data Transmission, Storage, and Backup section provides a technical description of the data transmission paths from data generation to final archiving. Additionally, this section covers the expected size of data generation efforts.

DATA MANAGEMENT APPROACH

The data management approach provides a high-level overview of how data was handled throughout the course of this project, including details of data standards and control policies that applied to data generated for this project. Furthermore, this section details the specific transmission paths and technical details of the data storage mediums planned throughout the project. Finally, the data management approach talks about retention policies both during and at the conclusion of the project.

Data Description

The primary outcome of this project is the development of raw sensor data, processed trajectory data, and appropriate metadata that enables researchers to better understand the impacts of ADAS-equipped vehicles on transportation system performance.

The published datasets contain processed vehicle trajectory information for both the data collection SV(s) (see table 8, table 12, and table 13) and surrounding AdjVs (see table 7 and table 11) as the primary DS. This primary DS is supplemented with additional information such as infrastructure conditions or environmental sensors (see table 9 and table 14) to further describe the operating conditions in which the raw data used to extract processed trajectories were collected. In addition, extra details on each run have been captured in metadata variables for the each of the two deployments (see table 10 and table 15).

There is a noted lack of standards that dictate the format of CAV trajectory datasets and variables. To inform this decision, the project team completed a review of selected publicly available datasets. The objective was to identify best practices and relevant standards for data variables within similar datasets. The project team reviewed the standards and segregated the data into two levels: the run-level information and day-level information. The run-level information contained direction of travel, run number, scenarios of interest, and following distance. The day-level information contained start and end point of the route, distance traveled, road condition, and speed limit. As part of this analysis, the project team also looked at past USDOT-published open-source data such as the Next-Generation Simulation (NGSIM) dataset to understand the nature (format, interoperability) of published data variables.[38]

Data Transmission, Storage, and Backup

This section provides a technical description of the data transmission paths from data generation to final archiving. Along the transmission paths, technical descriptions of the data storage mechanisms are presented, as well as data resiliency measures. As an inherent part of data storage considerations, the expected size of data generation efforts is also discussed.

As shown in figure 32, data transmission started at the beginning of the data flow process with the generation of data during a testing activity. Several data sources might flow to necessary DAQ devices. Once captured by the appropriate DAQ device, the first data checkpoint is reached. This initial data checkpoint was an in-field data verification to ensure that data collection devices were appropriately recording information and that data sources appeared reasonable in magnitude, frequency, and expected behavior. This data check is the lowest level check to ensure that data collection methods are meeting the DCP needs.

After completion of a testing activity, all collected data resided on local storage mediums that were deployed in-field. The storage medium will depend on the collection method; however, in nearly all cases, the mediums were either nonvolatile microsecure digital cards (720 Mbps) or a solid-state drive (SSD) with NOR-AND (NAND) flash nonvolatile memory (6 Gbps). Upon completion of a testing activity, data were transferred through Transmit-1 from the local storage medium to a combination of the PL's onsite storage servers and cloud storage servers. The transfer occurred onsite at the PL's facility. Transmit-1 occurred via one or a combination of

three transmission paths: Ethernet®/fiber (1 Gbps), USB 3.0 (5 Gbps), or SATA dII (6 Gbps). The transmission path was dependent on the storage medium. As a result of Transmit-1, data was transferred from local storage to the PL's local storage servers. The PL's local storage servers utilize three data storage rack-mounted server devices connected via 1-Gbps Ethernet connections. The data storage servers are a hybrid combination of SSDs and hard disk drives. In this manner, the storage arrays are designed to handle both primary and secondary flash workloads. The data storage server's storage devices are controlled by load-balancing software and load-balanced across the three blades to ensure even loading.

The local servers are located in a secured location with restricted access, thus providing physical security. Virtual access is controlled through permissions and right policies defined by the PL's information technology (IT) department. The servers feature end-point protection and change management notifications to provide additional data protection. The local storage server's operating system (OS) supports a Triple+ parity Redundant Array of Independent Disks (RAID) schema that tolerates three simultaneous drive failures per bank, thus achieving data resiliency through RAID in the face of potential storage medium failure. The data are also backed up to a cloud storage location daily with hourly incremental backups. Should any single drive fail, RAID and cloud storage backups ensure that data is not lost. Ethernet/fiber (1 Gbps) was used to support transmission to offsite cloud storage locations. The cloud storage servers have multiple backup methods that occur on data storage change with multiple server centers located throughout the United States. Cloud server centers are physically secured sites, and a security information and event management (SIEM) tool monitors the offsite storage to provide virtual access security.

Once data were securely on the onsite servers, project team members onsite or with remote-access privileges could access those servers to perform a second data validation check and begin data postprocessing. The validation check, along with human spot check verification, was an automated process of data verification to ensure that data was appropriately captured during any testing activity. The project team developed automated validation checks. The validation method used was dependent on the data stream being validated. This validation took on several methods to be both efficient and thorough. The data for SV1 were collected using a DAQ system and for SV2 using ROS node. The DAQ's diagnostic tools were used to check data as it was coming in to ensure that sensors were active and collecting at the expected frequency. When the data were exported from the DAQ, the project team inspected the size of the files to ensure the expected volume of data were received. Once on the processing computers, data were further validated by manually ensuring all sensor data came in at the expected rate and that all the sensor data were synchronized within the vehicle.

After validation, appropriate datasets were transferred to the PL's cloud storage site. Dataset transmission to the PL's cloud storage site is Transmit-2 in figure 32. Due to the large nature of this data collection effort and the dataset sizes, cloud storage was the most appropriate method of data transfer for Transmit-2. The PL hosted the cloud storage site. The server hosting the cloud storage site was a rack-mounted storage device with the same properties and protection enumerated previous in this section for the PL's storage servers (where the data was held after Transmit-1). Cloud storage access was controlled by the same IT control policies previously enumerated. Access was limited to PL approved parties, which required users to set up a controlled-access account through the IT department. This required users to submit a valid email

address/username for verification and required users to create a password. User read/write policies are controlled by the PL. Account information was stored onsite on the PL's storage server. Cloud storage access was enabled through a 1-Gbps Ethernet connection.

With the raw datasets on the cloud storage, the PSC could access data through Transmit-3. The PSC downloaded information onto one of its servers, which feature 1-Gpbs Ethernet connections. The PSC's servers currently utilize a RAID 0 construction with data redundancy through a backup to a cloud location. The backup process is automated and occurs daily. The PSC servers are physically onsite in a secured location with restricted access. Virtual access is controlled through the PSC's IT department that dictates controlled permissions and read/write access policies. The servers also feature end-point protection providing additional data security. With the raw datasets secured onto the PSC's storage servers, the PSC started postprocessing data according to the data-processing pipeline described in chapter 3. Once the data was postprocessed, the information was uploaded to the cloud storage through Transmit-3.

Next, the PL downloaded postprocessed data from the cloud storage onto the PL's storage servers through Transmit-2. Once the raw and processed datasets were on the PL's storage servers, the PL performed final postprocessing and validation checks. These checks were aimed at validating the processed data against the raw data and ensuring that consistent information was represented in both the datasets, as discussed in the data validation section of chapter 3.

With these checks completed, the team provided the information to FHWA and approved parties on an interim basis during the course of the project and final dataset transfer. Transmit-4 was the transmission path for FHWA's interim data access and was achieved via a secured web portal. A secured web portal is a cloud server offered commercially with appropriate data redundancies and data governance. The PL's IT department set up a restricted secured web portal that is limited to 100 Gb of storage and no more than 30,000 files. The PL has a SIEM tool for internet security, a next-generation firewall, server end-point protection, and server change management notifications.

Transmit-5 is the end of project transmission where raw and processed datasets were archived. Two transmissions occurred through Transmit-5:

- The first data transfer through Transmit-5 contained all raw and processed data. Descriptive metadata with title, keywords, and context about the raw and final processed datasets was uploaded with each dataset. These datasets contain personally identifiable information (PII) and require the appropriate protections of data containing sensitive information. These data are being stored as part of the MDMS in the Office of Safety and Operations Research and Development at the Turner-Fairbank Highway Research Center for internal use.

- The second transfer of data was from the PL's storage servers to the ITS DataHub.[37] This second transmission contained metadata and the processed datasets. This does not contain raw datasets nor any information containing PII. Providing these data allows the easiest and widest access of data to all potential users.

The final checkpoint verifies successful data transmission to the ITS DataHub and MDMS.[37]

Data Rights and Controlled Access

This section describes the project team's policies for ownership, rights, and controlled access of data collected during the project. Providing appropriate data access to safeguard data is a key aspect of data stewardship. In accordance with the OPEN Government Data Act, datasets must be made publicly accessible unless specific concerns require the data to have controlled access.(39)

Creative Commons Zero License for Published Datasets

The project team assigned an open license to datasets collected under this Intelligent Transportation Systems Joint Program Office (ITS/JPO)-funded data collection effort. The open license encourages reuse of datasets in the public domain without any restrictions concerning their reuse, attribution, or copyrights. All final published datasets uploaded to ITS/JPO Public Repository are governed under the Creative Commons Zero (CC0) license.(40)

By using a CC0 License, the project team and the USDOT/ITS/JPO/FHWA organization choose to opt out of any copyright and related or neighboring rights over these datasets. This choice enables the project team to explicitly declare our "No Rights Reserved" stance over these datasets. This approach allows the creators of the database to waive all copyright and related rights for these databases to the fullest extent allowed by law.(40)

Controlled Access and Ownership of Data and Datasets

Controlled access is defined as restricting access to certain groups of persons due to data containing PII, information that threatens the privacy of an individual or group, information that threatens the confidentiality of a person or group, or information that contains confidential business information. Data containing PII must be handled securely. Because video data was recorded outside of the vehicle, no guarantee exists that PII was not collected and therefore raw data with the video is considered PII.

The project team, in collaboration with FHWA, determined controlled-access policies applicable to the data acquired during this project. Table 18 outlines the ownership of data, controlled-access policies, and reasons for controlled-access restrictions on data at various stages of the lifecycle.

Table 18. Data controlled access and ownership overview.

Step	Storage Location	Type of Data	Access	Controlled Access	Reasons for Controlled Access
1	DAQ	Raw data	Project lead	Yes	Raw data, PII
2	Storage server	Raw data	Project lead	Yes	Raw data, PII
3	Processing PC	Sanity check routines, raw data	Project lead	Yes	Raw data, PII

Step	Storage Location	Type of Data	Access	Controlled Access	Reasons for Controlled Access
4	Storage server	Raw data	Project lead	Yes	Raw data, PII
5	Cloud storage	Validated raw data	Project team, FHWA, FHWA-approved auditor	Yes	Raw data, PII
6	Subcontractor storage server	Validated raw data	Project subcontractor	Yes	Raw data, PII
7	Subcontractor processing PC	Validated raw and processed data	Project lead	Yes	Shared space with raw data
8	Cloud storage	Processed Data for Review	PL, FHWA, FHWA-approved auditor	Yes	Shared space with raw data
9	ITS DataHub	Processed Datasets with no PII	FHWA	No	—
10	Turner-Fairbank Highway Research Center	Processed Data Raw Data with PII	FHWA-approved users	Yes	PII

— No data.

FHWA/ITS/JPO Public Storage Solution

The project team and FHWA decided to archive the final processed data on the ITS DataHub. Data published on the ITS DataHub do not contain sensitive information. The raw data, which contains sensitive information, will be stored as part of the MDMS in the Office of Safety and Operations Research and Development at the Turner-Fairbank Highway Research Center. This secondary storage location will contain the entirety of datasets generated during this project, including both raw and processed data. The authorized access to the cloud storage was granted based on request from FHWA to different organizations.

Data Size

Table 19 shows data size by the data collection rate. Modification to use LiDAR on both vehicles means that the data collected per hour is nearly the same between SV1 and SV2. A total of around 10 TB of data was collected.

Table 19. Hourly data capture rate.

Data Capture	Unit	Recorded Data Rate
Data collection vehicle Vehicle location/orientation, motion, control input, and other parameters	kB/s	380 MB/h
Three video feeds at 720 p 10 Hz, H.264 compression	GB/h	21 GB/h
LiDAR point cloud	GB/h	48 GB/h
Total data/h	GB/h	69.38 GB/h

At project completion, the team uploaded two CSV files to the ITS/JPO DataHub containing the final processed datasets available for public use in accordance with Creative Commons Zero:[37,40]

- Single-SV operation:[2]
  - CSV size: 3.3 GB.
  - Number of instances (rows): 4,169,447
  - Number of data collection runs: 215.
  - Number of hours of collected data: 120.

- Two-SV operation:[3]
  - CSV size: 2.2 GB.
  - Number of instances (rows): 2,530,154.
  - Number of data collection runs: 68.
  - Number of hours of collected data: 24.

Note that an instance is an observation of an SV and an AdjV at a specific time step.

Data Retention, Archiving, and Maintenance

Data retention, maintenance, and archiving strategy is divided into two distinct periods: during the project and after the project concluded.

For the duration of the project, data existed in numerous places at once depending on the current stage of the project (see figure 32); however, the PL maintains copies of the original raw data stored and backed up on the onsite servers. This ensures that data will always be available should any storage medium fail along the defined transmission paths. After the project concluded, data was transferred to the final data storage locations depending on whether it is protected or publicly available data.

The onsite data storage servers have data resiliency features that attempt to minimize data loss. The data storage servers' data resiliency features are specified in the Data Transmission, Storage, and Backup section of this report. In addition to the built-in features of the data storage servers, the PL utilized cloud services to back up data daily with incremental hourly backups.

CHAPTER 5: KEY CONCLUSIONS FROM THE PROJECT

This project successfully collected data on ADAS-equipped vehicles with SAE Level 2 partial driving automation capabilities in naturalistic traffic. The project's goal was to create a dataset containing trajectories for an instrumented ADAS-equipped vehicle with SAE Level 2 partial driving automation capabilities (SV) and all AdjVs that were perceived by the sensors on the ADAS-equipped vehicles in naturalistic traffic. This data collection effort used both a discreet production ADAS-equipped vehicle and readily identifiable ADAS-equipped vehicles where the sensor stack is visible to AdjVs. The team collected the dataset under diverse road and traffic conditions with a broad spectrum of roadway types, roadway conditions, traffic densities, and speed limits at locations around Columbus, OH. The project team hopes that this dataset will enable researchers to characterize and model the interaction between ADAS-equipped vehicles and their driving environment, which includes other road users. Ultimately, this effort will support an improved understanding of how ADAS-equipped vehicles and how driver perception of a sensor stack can impact transportation system performance.

Lessons Learned

Several lessons were learned over the course of the project that could enable improved data collection and processing on future projects.

Data Collection Lessons

First, the research team knew that challenges would exist for detecting and tracking objects more than 45 m away from the LiDAR unit, given the sparseness of the LiDAR point cloud produced for those objects. The team did not anticipate that adding a second data collection vehicle (originally part of the test plan to evaluate how drivers responded to two ADAS-equipped vehicles instead of just a single vehicle in the traffic stream) would significantly improve the data quality and AdjV detection range by providing a denser PCD. In addition to the improved range, knowing the precise location of a second vehicle provided a straightforward method for validating the processed data against ground truth, raw data. Thus, in future data collection efforts, research teams may wish to consider multiple vehicle deployments to increase the amount of data they are collecting for processing.

Second, the research team collected forward- and rear-facing camera footage but found the footage was not a reliable data source due to the poor quality of some of the video data, particularly overexposed video data. The team adjusted the camera after the initial data sample was collected to improve the quality of the data produced. However, the research team ultimately decided to develop a data-processing pipeline that did not use the camera data as an input because of the challenges in working with the over exposure of the video data. In addition, the team needed to frequently extrinsically calibrate cameras and other sensors, such as LiDAR, when the orientation of the camera changed, which is another challenge when collecting real-world camera data and fusing it with other sensors. Thus, processing any data collected during rain events was challenging. The team found that the LiDAR point cloud data was significantly impacted by the rain, making the collected data virtually impossible to process and forced the team to abandon efforts to process any raw collected during inclement weather. Choosing a

combination of sensors that perform well under all the conditions in data collection, such as under different lighting and weather conditions, would improve future collected dataset.

Finally, the research team used two video file formats for the data collection. This decision required the team to take additional time to convert the raw data files to a consistent format with further processing to acquire the required parameters. The team was able to modify the settings in one of the DAQ systems to eventually collect all video data in the same format, saving a lot of time. Therefore, ensuring the data recoding file formats are consistent throughout the deployments enables more efficiency.

Data Processing Lessons

Exposure-related data collection issues aside, another way the camera data could have been more useful is if more were known about its characteristics, specifically in relation to the LiDAR at the start of the project, enabling sensor fusion algorithms to be available. More thorough documentation of the relationship between sensors (i.e., extrinsic and intrinsic sensor calibration) would have enabled the team to apply more diverse algorithms to the data. Some of these data were collected partway through the project to enable the postprocessing described in chapter 3. The project team recommends both calibrating each sensor and documenting the sensor extrinsic and intrinsic characteristics and the sensor's relationship to the entire sensor stack (even if the information is not initially anticipated to be used in the algorithm), particularly at the start of the project. This documentation will enable more fluid changes to the processing pipeline once data starts coming in and a more algorithm agnostic dataset that can enable more research in the future.

Beta Users of the Data

To increase the usability of the data, a few internal teams accessed this dataset before it was posted online to help document potential pain points in understanding the processed data, which significantly improved the data's documentation.[2,3] This section summarizes some ongoing and recently completed research efforts using the central Ohio ADAS datasets.

The methodology developed for data collection and processing has been used in two projects by the data collection and processing team. First, the lessons learned by this project improved future data collection efforts on an ADS grant project, funded by the Federal Motor Carrier Safety Administration. In earlier stages of that project, the project team was able to build on the principles of this project by enabling SV2 to use HD maps in conjunction with the automation software installed on the vehicle to fully navigate on the road (with safety driver supervision). This project aided in providing a better upfront collection of sensor characterization and calibration from a system level and DCP. The success of the multiple vehicle deployment has led to the team prioritizing using multiple (two or more) vehicles on the road at the same time to enable improved data analysis and verification (a lesson learned from this project).

Second, the project team has taken the raw data from this project and annotated the LiDAR data with different types of road users such as vehicles, pedestrians, and lane markings for machine learning model training. The team also benchmarked several cooperative LiDAR-based object-detection algorithms from early fusion to late fusion.[41]

Additionally, a separate FHWA project team used the data collected by this project to develop a multivariate piecewise linear model (MPL) to capture the ADAS-equipped vehicle's ACC behaviors. The project team compared the performance of the MPL model with a traditional linear model and a nonlinear model. The velocity profiles of the real SV and simulation data indicate that the MPL model has the best performance, particularly in the crest and trough regions.[42]

Future Uses of the Data

Researchers can use the data generated in this study in numerous ways including, but not limited to, the following ideas proposed in chapter 1:

- Are adjacent drivers (i.e., drivers that are interacting with the instrumented ADAS-equipped vehicles whose behavior is captured through the instrumented vehicle's sensor stack) altering their driving behavior (e.g., following distance, gap acceptance) when interacting with ADAS-equipped vehicle compared to their baseline behavior interacting with other manually driven vehicles in traffic? The database file contains trajectories for both the SVs and AdjVs (e.g., time series position, speed, and acceleration data). Thus, researchers can use these data to study the effect of the ADAS-equipped SV on the driving behavior of the nonautomated AdjVs.

- Are the behavioral changes of nonautomated AdjVs due to the behavior changes associated with the ADAS-equipped vehicle behavior (e.g., more consistent following distance; larger headways), or are drivers altering their behavior due to the appearance of the ADAS-equipped vehicle (e.g., visibility of sensor suite)? The type_of_vehicle metadata column records whether the SV in a specific row is being operated as a D-ADAS or an RI-ADAS-equipped vehicle. Thus, future users of this dataset can compare how traffic flow is affected when drivers may perceive that they are near vehicles with advanced capabilities.

- How does SV operation impact traffic flow differently than multivehicle strings? The data are stored in separate single- and two-vehicle data collection files, enabling future research into how multiple vehicles with conspicuous sensor stacks may affect traffic flow. Please remember that although SV2 was an RI-ADAS, it was operated as an SAE Level 0 vehicle with no ADAS features used for driving assistance due to limitations with the map data.

- What is the impact of driving environment (e.g., freeway versus arterial; clear roads versus wet roads) on ADAS performance? The data were collected on both dry and wet roadways, allowing for analysis of traffic patterns based on environmental conditions. As discussed in the lessons learned section, the only adverse weather conditions that could be processed were wet pavements (after rain had ceased) due to challenges with processing LiDAR data during rain events. The metadata columns collected for each instance of the data should enable researchers to filter the data according to their specific research questions regarding the impact of driving environment on ADAS-equipped vehicle performance and AdjV behavior.

Future Research

This study's conclusion leaves several areas to be further explored. First, this study was limited by the technology available at the time of data collection, which is why the project team collected all data with SAE Level 2 ADAS-equipped vehicles. With advancing technology, data collection efforts in the future may be performed using ADS (SAE Levels 3–5).[1]

In addition, this project was unable to successfully capture data using CVs, despite efforts by the team to collect data in central Ohio, which has an increased penetration rate of CV and infrastructure technology.[43,44] In future data collection efforts, teams should consider designing data collection efforts to capture the impact of CV, connected ADAS-equipped vehicles, and connected ADS-equipped vehicles on traffic flow and AdjV behavior.

ACKNOWLEDGMENTS

The original map in figure 7 was modified to show the start and end points of the data collection routes. The original maps are the copyrighted property of Google Earth and can be accessed at https://www.google.com/earth.

The original maps in figure 22-A and figure 23-A were modified to show the routes followed by the vehicles during data collection. The original maps are the copyrighted property of Google Earth and can be accessed at https://www.google.com/earth.

REFERENCES

1. SAE International. 2020. *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*. SAE J3016_202104. Warrendale, PA: SAE International. https://www.sae.org/standards/content/j3016_202104/, last accessed June 30, 2023.

2. USDOT. 2023. "Advanced Driver Assistance System (ADAS)-Equipped Single-Vehicle Data for Central Ohio" (web page). https://data.transportation.gov/Automobiles/Advanced-Driver-Assistance-System-ADAS-Equipped-Si/iie8-uenj, last accessed September 26, 2023.

3. USDOT. 2023. "Advanced Driver Assistance System (ADAS)-Equipped Two-Vehicle Data for Central Ohio" (web page). https://data.transportation.gov/Automobiles/Advanced-Driver-Assistance-System-ADAS-Equipped-Tw/vhz2-exyi, last accessed October 11, 2023.

4. *Fixing America's Surface Transportation (FAST) Act*. 2015. Pub. L. No. 114-94. U.S. Government Publishing Office, December 3, 2015. https://www.govinfo.gov/app/details/PLAW-114publ94, last accessed August 24, 2023.

5. AAA. 2019. *Advanced Driver Assistance Technology Names*. Heathrow, FL: American Automobile Association. https://www.aaa.com/AAA/common/AAR/files/ADAS-Technology-Names-Research-Report.pdf, last accessed November 1, 2022.

6. Mahmassani, H. S., A. Elfar, S. Shladover, and Z. Huang. 2019. *Development of an Analysis/Modeling/Simulation (AMS) Framework for V2I and Connected/Automated Vehicle Environment*. Report No. FHWA-JPO-18-725 Washington, DC: U.S. Department of Transportation. https://rosap.ntl.bts.gov/view/dot/39965, last accessed June 30, 2023.

7. FHWA. 2013. *Highway Functional Classification Concepts, Criteria and Procedures*. Publication No. FHWA-PL-13-026. Washington, DC: Federal Highway Administration.

8. Open Robotics. 2020. *Robot Operating System* (software). Version 1 Melodic.

9. DEWEsoft. 2022. "Dewesoft Downloads" (web page). https://dewesoft.com/download, last accessed June 30, 2023.

10. Claims MS GmbH. n.d. "AccidentSketch.com" (web page). https://accidentsketch.com/, last accessed June 30, 2023.

11. FHWA. 2014. "Highway Performance Monitoring System (HPMS). 5.0 Guidelines for Data Collection for High-Volume Routes" (web page). https://www.fhwa.dot.gov/policyinformation/hpms/volumeroutes/ch5.cfm, last accessed June 30, 2023.

12. FHWA. 2022. “Manual on Uniform Traffic Control Devices (MUTCD): Frequently Asked Questions - Part 5 - Traffic Control Devices for Low-Volume Roads.” (web page). Federal Highway Administration. https://mutcd.fhwa.dot.gov/knowledge/faqs/faq_part5.htm, last accessed June 30, 2023.

13. Mid-Ohio Regional Planning Commission. 2023. “Transportation Data Management System.” (web page). https://morpc.public.ms2soft.com/tcds/tsearch.asp?loc=Morpc&mod=, last accessed June 30, 2023.

14. FHWA. 2017. *User Guide for USLIMITS2*. Washington, DC: Federal Highway Administration. https://safety.fhwa.dot.gov/uslimits/documents/appendix-L-user-guide.pdf, last accessed June 30, 2023.

15. Google®. 2023. “Google Earth™” (web page). https://earth.google.com/web/, last accessed June 30, 2023.

16. National Geospatial-Intelligence Agency. 2023. “World Geodetic System 1984 (WGS 84)” (web page). https://earth-info.nga.mil/?dir=wgs84&action=wgs84, last accessed September 26, 2023.

17. Stevens, A., M. Dianati, K. Katsaros, C. Han, S. Fallah, C. Maple, F. McCullough, and A. Mouzakitis. 2017. “Cooperative Automation Through the Cloud: The CARMA Project.” In *Proceedings of 12th ITS European Congress*. Brussels, Belgium: ITS European Congress, 1–6.

18. USDOT. 2023. “carma-wm” (CARMA model software and configuration files in GitHub repository). https://github.com/usdot-fhwa-stol/carma-platform/tree/develop/carma_wm, last accessed October 10, 2023.

19. Open Robotics. 2021. “rosbag/Code API” (web page). http://wiki.ros.org/rosbag/Code%20API, last accessed Jun.25, 2023.

20. Baidu Apollo. 2022. “apollo” (software and configuration files in GitHub repository). https://github.com/ApolloAuto/apollo, last accessed June 30, 2023.

21. AutoLidarPerception. “tracking_lib” (LiDAR tracking library software and configuration files in GitHub repository). https://github.com/AutoLidarPerception/tracking_lib, last accessed Jun.25, 2023.

22. Himmelsbach, M., and H. J. Wuensche. 2012. “Tracking and Classification of Arbitrary Objects with Bottom-up/Top-down Detection.” In *2012 IEEE Intelligent Vehicles Symposium*. New York, NY: Institute of Electrical and Electronics Engineers, 577–582.

23. Xiong, L., X. Xia, Y. Lu, W. Liu, L. Gao, S. Song, and Z. Yu. 2020. “IMU-Based Automated Vehicle Body Sideslip Angle and Attitude Estimation Aided by GNSS Using Parallel Adaptive Kalman Filters.” *IEEE Transactions on Vehicular Technology* 69, no. 10: 10668–10680.

24. Werling, M., J. Ziegler, S. Kammel, and S. Thrun. 2010. “Optimal Trajectory Generation for Dynamic Street Scenarios in a Frenet Frame.” In *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*. New York, NY: Institute of Electrical and Electronics Engineers, 987–993.

25. Xia, X., Z. Meng, X. Han, H. Li, T. Tsukiji, R. Xu, Z. Zheng, and J. Ma. 2023. “An Automated Driving Systems Data Acquisition and Analytics Platform.” *Transportation Research Part C: Emerging Technologies* 151: 104120.

26. Akai, N., L. Y. Morales, E. Takeuchi, Y. Yoshihara, and Y. Ninomiya. 2017. “Robust Localization Using 3D NDT Scan Matching with Experimentally Determined Uncertainty and Road Marker Matching.” In *Proceedings of the 2017 IEEE Intelligent Vehicles Symposium IV*. New York, NY: Institute of Electrical and Electronics Engineers, 1356–1363.

27. Crescenzi, V., G. Mecca, and P. Merialdo. 2001. “RoadRunner: Towards Automatic Data Extraction from Large Web Sites.” *VLDB 1:* 109-118.

28. Althoff, M., S. Urban, and M. Koschi. 2018. “Automatic Conversion of Road Networks from Opendrive to Lanelets.” *Proceedings of the 2018 IEEE International Conference on Service Operations and Logistics, and Informatics*. New York, NY: Institute of Electrical and Electronics Engineers, 157–162.

29. USDOT. 2023. “opendrive2lanelet” (software and configuration files in GitHub repository). https://github.com/usdot-fhwa-stol/opendrive2lanelet, last accessed June 30, 2023.

30. Yao, Y., B. Deng, W. Xu, and J. Zhang. 2020. “Quasi-Newton Solver for Robust Non-Rigid Registration.” In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. New York, NY: Computer Vision Foundation, 7600–7609.

31. Poggenhans, F., J. H. Pauls, J. Janosovits, S. Orf, M. Naumann, F. Kuhnt, and M. Mayr. 2018. “Lanelet2: A High-Definition Map Framework for the Future Of Automated Driving.” In *Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems*. New York, NY: Institute of Electrical and Electronics Engineers, 1672–1679.

32. MathWorks. 2023. “FIR Filter Design” (web page). https://www.mathworks.com/help/signal/ug/fir-filter-design.html, last accessed June 30, 2023.

33. Xiong, L., X. Xia, Y. Lu, W. Liu, L. Gao, S. Song, Y. Han, and Z. Yu. 2019. “IMU-based Automated Vehicle Slip Angle and Attitude Estimation Aided by Vehicle Dynamics.” *Sensors* 19, no. 8: 1930.

34. Xu, R., H. Xiang, X. Xia, X. Han, J. Li, and J. Ma. 2022. “OPV2V: An Open Benchmark Dataset and Fusion Pipeline for Perception with Vehicle-to-Vehicle Communication.” In

Proceedings of the 2022 International Conference on Robotics and Automation (ICRA). New York, NY: Institute of Electrical and Electronics Engineers, 2583–2589.

35. Cheng, S., K. Zhao, and D. Zhang. 2019. "Abnormal Water Quality Monitoring Based on Visual Sensing of Three-Dimensional Motion Behavior of Fish." *Symmetry* 11, no. 9: 1179.

36. Meng, Z., X. Xia, R. Xu, W. Liu, and J. Ma. Forthcoming. "HYDRO-3D: Hybrid Object Detection and Tracking for Cooperative Perception Using 3D LiDAR." *IEEE Transactions on Intelligent Vehicles.*

37. USDOT. n.d. "ITS DataHub" (web page). https://www.its.dot.gov/data/index.htm, last accessed June 30, 2023.

38. FHWA. 2020. "Next Generation Simulation (NGSIM)" (web page). https://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm, last accessed June 30, 2023.

39. U.S. Congress. Senate. *Open Government Data Act*. 115th Cong., 1st sess. 2017. S. Rep. 115-134. https://www.govinfo.gov/app/details/CRPT-115srpt134/CRPT-115srpt134.

40. Creative Commons. n.d. "CC0" (web page). https://creativecommons.org/share-your-work/public-domain/cc0/, last accessed June 30, 2023.

41. Xu, R., X. Xia, J. Li, H. Li, S. Zhang, Z. Tu, Z. Men, H. Xiang et al. "V2v4Real: A Real-World Large-Scale Dataset for Vehicle-to-Vehicle Cooperative Perception." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. New York, NY: Institute of Electrical and Electronics Engineers, 13712–13722.

42. James, R. 2023. "Developing Improved Models for Assessing the Impacts of Automation on Transportation Operations" (workshop). Presented at the *2023 TRB Annual Automated Road Transportation Symposium*. Washington, DC: Transportation Research Board.

43. City of Columbus, OH. 2021. *Smart Columbus. Program Summary*. Columbus, OH: City of Columbus, OH. https://d2rfd3nxvhnf29.cloudfront.net/inline-files/Smart%20City%20Challenge-%20USDOT%20Executive%20Summary.pdf, last accessed June 30, 2023.

44. Connected Marysville. 2023. "Connected Marysville" (web page). https://connectedmarysville.com/, last accessed June 30, 2023.

Recommended citation: Federal Highway Administration,
Advanced Driver Assistance System-Equipped Vehicle Datasets Collected in Central Ohio: Final Report (Washington, DC: 2024) https://doi.org/10.21949/1521757

HRSO-50/03-24(WEB)E