



4-1-2023

Improving Future Vehicle Fuel Economy and Operational Design Domain Through Novel Data Pipelines

Kyle James Carow
Western Michigan University

Follow this and additional works at: https://scholarworks.wmich.edu/masters_theses



Part of the Automotive Engineering Commons, and the Mechanical Engineering Commons

Recommended Citation

Carow, Kyle James, "Improving Future Vehicle Fuel Economy and Operational Design Domain Through Novel Data Pipelines" (2023). *Masters Theses*. 5366.
https://scholarworks.wmich.edu/masters_theses/5366

This Masters Thesis-Open Access is brought to you for free and open access by the Graduate College at ScholarWorks at WMU. It has been accepted for inclusion in Masters Theses by an authorized administrator of ScholarWorks at WMU. For more information, please contact wmu-scholarworks@wmich.edu.



IMPROVING FUTURE VEHICLE FUEL ECONOMY AND OPERATIONAL DESIGN DOMAIN THROUGH NOVEL DATA PIPELINES

Kyle James Carow, M.S.E.

Western Michigan University, 2023

Modern automobiles have greatly advanced in recent years, with technological developments that enhance performance, safety, and comfort. However, there is still much room for improvement. Today's vehicles are heavily reliant on the combustion of fossil fuels, proven to be harmful for the environment on both a local and global scale. In addition, the safety benefits of autonomous vehicles and advanced driver assistance systems are not yet fully realized due to the limited operational design domain of these technologies. In this research, these needs are addressed through the development of two novel data pipelines. In the first study, a novel methodology is outlined that generates vehicle fuel economy models from real-world sparse fleet telematics data, enabling heavy-duty vehicle fleet operators to explore the minimization of fueling costs in new ways and to serve as an initial probe into a fully data-informed pipeline. In the second study, a data processing procedure is described that recovers lane-line geometry information from high-definition map data, expanding the operational design domain of automated vehicle technology to scenarios where lane lines are totally occluded by snow, leaves, or shadows. It is concluded that these methodologies have significant potential to improve the fuel efficiency and operational design domain of future vehicles.

IMPROVING FUTURE VEHICLE FUEL ECONOMY AND OPERATIONAL
DESIGN DOMAIN THROUGH NOVEL DATA PIPELINES

by

Kyle James Carow

A thesis submitted to the Graduate College
in partial fulfillment of the requirements
for the degree of Master of Science in Engineering
Mechanical Engineering
Western Michigan University
April 2023

Thesis Committee:

Zachary D. Asher, Ph.D., Chair
Richard T. Meyer, Ph.D.
Damon A. Miller, Ph.D.

Copyright by
Kyle James Carow
2023

ACKNOWLEDGMENTS

I would first like to recognize the friends and family who have supported me on my journey through graduate school by sitting through presentations, editing manuscripts late into the night, and offering innumerable helpful suggestions. Without their consistent, dedicated support, I would not be where I am today.

I would also like to acknowledge the support I have received from the EEAV lab, especially my research advisor Dr. Asher, through whom all of my graduate education and research was made possible. I want to thank the other lab members for the many wonderful memories we made both collaborating on research and socializing; I will always look back on my time at the lab with these memories held close to my heart.

Of course, this would not be possible without the assistance of my other committee members Dr. Meyer and Dr. Miller, who both have supported me immensely in this final push.

I thank you all from the bottom of my heart.

Kyle James Carow

TABLE OF CONTENTS

ACKNOWLEDGMENTS	ii
LIST OF TABLES	v
LIST OF FIGURES	vi
LIST OF ABBREVIATIONS.....	viii
CHAPTER	
I. INTRODUCTION	1
II. HIGH-FIDELITY HEAVY-DUTY VEHICLE MODELING USING SPARSE TELEMATICS DATA	3
Abstract.....	3
Introduction.....	4
Methodology	7
Telematics Data Overview.....	7
Model Derivation.....	11
Drive Cycle Derivation	11
Model Validation	19
Results and Discussion	20
Overall Results.....	20
Velocity Signal Processing Results	21
Conclusions.....	25
Chapter II References	26

Table of Contents—Continued

CHAPTER

III. PROJECTING LANE LINES FROM HIGH-DEFINITION MAPS FOR AUTOMATED VEHICLE PERCEPTION IN ROAD OCCLUSION SCENARIOS	29
Abstract	29
Introduction.....	30
Methods.....	33
Data Collection	33
Data Processing.....	36
Artificial Occlusion of Lane Lines	40
Metrics of Evaluation.....	42
Data Accuracy Metrics	42
Lane Line Overlay Metrics	43
Results.....	43
Data Accuracy Analysis.....	43
Lane Line Overlay Analysis	44
Eaton Proving Grounds Test Track.....	44
Campus Drive Loop.....	46
Results Summary	48
Conclusions.....	49
Chapter III References	50
IV. CONCLUSION.....	55

LIST OF TABLES

1. Sample counts for the 15 most common telematics measurement signals across entire fleet	8
2. Velocity signal processing summary table	12
3. Tabulated velocity signal processing results.....	24

LIST OF FIGURES

1. Sorted bar plot showing counts for the 15 most common telematics measurement signals. Note that the y-axis is in millions.....	9
2. Map of all telematics geospatial data.....	10
3. Velocity signal processing summary flowchart.....	13
4. Examples of an upward discontinuity in velocity (top) and a downward discontinuity in velocity (bottom) commonly found at signal gaps	15
5. Signal fusion demonstration for three consecutive time intervals. From left to right: fusion by interpolated averaging, fusion using signal A, fusion using signal B	17
6. Comparison of measured vs. simulated fuel consumption over time	20
7. Comparison of unprocessed measured and geospatially-derived velocity signals	21
8. Results of upward discontinuity correction. Shown is an uncorrected upward discontinuity (top) and the results of applying the correction algorithm (bottom).....	22
9. Results of downward discontinuity correction. Shown is an uncorrected upward discontinuity (top) and the results of applying the correction algorithm (bottom).....	22
10. Percent changes in crash rate due to inclement weather (directly from [16])	32
11. Trimble Catalyst DA2 GNSS receiver (directly from [25])	33
12. Collected lane line points at the Eaton Proving Grounds test track.....	34
13. Collected lane line points over the Campus Drive loop	34
14. WMU EEAV Lab research vehicles	35
15. Stereolabs ZED 2i stereo camera (directly from [26]).....	35
16. Swift Navigation Duro Inertial RTK GNSS receiver (directly from [27]).....	35

List of Figures—Continued

17. Frame from image data collected over the Eaton Proving Grounds test track	36
18. Frame from image data collected over the Campus Drive loop	36
19. Stereolabs ZED camera coordinate frame (directly from [28])	37
20. NED to XYZ coordinate frame rotation	38
21. OpenCV pinhole camera model coordinate frame (directly from [29])	39
22. Eaton Proving Grounds test track camera frame with lane lines artificially occluded, before and after	42
23. Campus Drive camera frame with lane lines artificially occluded, before and after.....	42
24. Two examples of Eaton Proving Grounds camera frames from straight road segments with high-definition map lane line data overlaid on artificially occluded lane lines. These show well-aligned lane lines	45
25. Eaton Proving Grounds camera frame from a curved road segment with high-definition map lane line data overlaid on artificially occluded lane lines. Deviation from true lane lines occurs in this curved road segment due to the GNSS operating in lower-precision RTK float mode and the insufficient sampling rate of the GNSS heading sensor	46
26. Two examples of Campus Drive camera frames from straight road segments with high-definition map lane line data overlaid on artificially occluded lane lines. These show well-aligned lane lines	47
27. Campus Drive camera frame from a curved road segment with high-definition map lane line data overlaid on artificially occluded lane lines. Deviation from true lane lines occurs in this curved road segment due to the insufficient sampling rate of the GNSS heading sensor.....	48

LIST OF ABBREVIATIONS

HDV	heavy-duty vehicle
EPA	Environmental Protection Agency
FASTSim	Future Automotive Systems Technology Simulator
NREL	National Renewable Energy Laboratory
NHTSA	National Highway Traffic Safety Administration
GNSS	global navigation satellite system
CDC	Centers for Disease Control and Prevention
ADS	Automated Driving Systems
AVs	Autonomous Vehicles
ADAS	Advanced Driver Assistance Systems
LDW	lane departure warning
LKA	lane keeping assistance
LCA	lane centering assistance
ODD	operational design domain
CV	computer vision
FHWA	Federal Highway Administration
WMU	Western Michigan University
NED	north, east, down
ROI	region of interest

CHAPTER I

INTRODUCTION

Today's vehicles are tremendous feats of engineering. From the invention of the automobile, a century of multi-disciplinary engineering has greatly improved performance, range, fuel efficiency, emissions, comfort, and safety. However, modern cars still leave much to be desired, and the vehicles of the future must conform to even stricter requirements. For example, the improvement of fuel economy and safety are two areas that will surely see ever-increasing engineering focus as pressure from consumers and governmental environmental regulation grows. This research is broken into two sections, with a study focusing on each topic individually.

The consumption of fossil fuels is directly tied to greenhouse gas emissions, which must be reduced as much as possible to mitigate impending climate disaster. Fuel is also a significant expense for heavy-duty vehicle fleet operators, who need to continually refuel vehicles that are innately less efficient due to large mass and poor aerodynamic characteristics. Likely the most rapid and inexpensive avenue of fuel economy improvement is through vehicle powertrain simulation, where software can simulate the required energy and fuel of a specific vehicle on a drive cycle. The first research question thus is "Can the sparse, telematics data available to fleet operators today be used to create and validate heavy-duty vehicle models?" To address this, the first study details a pipeline that creates heavy-duty vehicle models from real-world telematics data, enabling heavy-duty fleet operators to investigate ways to minimize fuel economy with reduced need for costly prototyping and testing. This work is found in Chapter II.

Automobile safety has always been a cornerstone of vehicle engineering, with inventions like seatbelts and airbags becoming mandatory and omnipresent in modern vehicles. Today,

advancements in the field of vehicle automation enable technology like advanced driver assistance systems, which will eventually develop into fully autonomous vehicles. This technology is currently limited to near-optimal scenarios, where the visibility of road features like lane lines is ensured. As such, when lane lines are occluded, as happens when snow, water, leaves, or shadows are present, the perception subsystem that enables automated driving loses necessary input and begins to break down. This limited operational design domain must be expanded to include lane line occlusion in order for autonomous vehicle technology to be viable anywhere. The second research question thus is “How can the operational design domain of automated vehicle technology be expanded to occluded lane-line scenarios?” The second study addresses this research question by describing a novel methodology for recovering occluded lane line geometry from high-definition maps that subverts the need for line-of-sight from perception sensors to the lane lines, thereby expanding the operational design domain of automated vehicle systems. This work is found in Chapter III.

To summarize the research questions of this thesis, Chapter II addresses the question “Can the sparse, telematics data available to fleet operators today be used to create and validate heavy-duty vehicle models?” by detailing a data processing pipeline that creates heavy-duty vehicle models from real-world telematics data, and Chapter III addresses the question “How can the operational design domain of automated vehicle technology be expanded to occluded lane-line scenarios?” by describing a novel methodology where occluded lane-lines geometry is recovered using high-definition maps.

CHAPTER II
HIGH-FIDELITY HEAVY-DUTY VEHICLE MODELING
USING SPARSE TELEMATICS DATA

This chapter consists of work developed in a project sponsored by Allison Transmission and presented in a conference publication at SAE World Congress Experience 2022, authored by Kyle Carow, Nathaniel Cantwell, Andrej Ivanco, Jacob Holden, Chad Baker, Eric Miller, and Zachary D. Asher. Please note that much of this chapter is presented verbatim from this publication. Kyle Carow led the development of the methodology and authorship of the publication, with the co-authors providing data, suggestions, and guidance. The citation for this work is below.

Carow, K., Cantwell, N., Ivanco, A., Holden, J. et al., “High-Fidelity Heavy-Duty Vehicle Modeling Using Sparse Telematics Data,” SAE Technical Paper 2022-01-0527, 2022, doi:10.4271/2022-01-0527.

Abstract

Heavy-duty commercial vehicles consume a significant amount of energy due to their large size and mass, directly leading to vehicle operators prioritizing energy efficiency to reduce operational costs and comply with environmental regulations. One tool that can be used for the evaluation of energy efficiency in heavy-duty vehicles is analysis of energy efficiency using vehicle modeling and simulation. Simulation provides a path for energy efficiency improvement by allowing rapid experimentation of different vehicle characteristics on fuel consumption without the need for costly physical prototyping. The research presented in this paper focuses on using real-world, sparsely sampled telematics data from a large fleet of heavy-duty vehicles to create high-fidelity models for simulation. Samples in the telematics dataset are collected

sporadically, resulting in sparse data with an infrequent and irregular sampling rate. Captured in the dataset was geospatial information, time-series measurements, and vehicle-specific metadata from a subset of 96 vehicles from varied geographic regions across North America. A series of custom algorithms was developed to process vehicle data and derive both vehicle model input parameters and representative drive cycles. Derived models provide a basis on which to simulate real-world vehicles and iterate on vehicle aerodynamics, auxiliary power loads, transmission shift schedules, and other parameters to achieve reduced fuel consumption and increase energy efficiency. Notably, these models were developed without the use of field data collection, using only data collected through fleet telematics. Processed representative drive cycles are used to validate the fuel economy of derived models. The models developed through this research allow for more representative vehicle simulations with increased flexibility regarding vehicle-to-vehicle variations.

Introduction

Heavy-duty vehicles (HDVs) consume a greater amount of fuel compared to light-duty vehicles because of their substantial mass, significant aerodynamic drag, and high auxiliary power loads. The greater consumption of fuel in HDVs leads to increased expenses for HDV fleet companies. Fueling accounts for 24% of motor carrier operating costs, second only to driver wages [1]. In addition, HDV fuel consumption contributes significantly to harmful greenhouse gas emissions. According to the U.S. Environmental Protection Agency (EPA), the combustion of fuel in freight trucks comprises the second largest source of CO₂ emissions within the transportation sector, at 23.6% [2]. Due to this, the EPA has set targets through the Clean Trucks Plan which sets increasingly strict limits for HDV greenhouse gas emissions [3]. These factors prompt HDV operators to explore optimization of fuel efficiency.

Energy efficiency optimization can be achieved through modeling and simulation of vehicles through software. While physical experimentation on vehicle parameters is expensive and time-consuming, simulation using validated models allows for rapid results through software alone. Modeling of HDVs presents a unique challenge, however, as unlike light-duty vehicles, HDVs possess a high degree of specialization as a result of vehicle vocation. Vehicle specialization causes great variations in mass, shape, and auxiliary power loads from vehicle to vehicle. In order to create sufficiently representative HDV models, model parameters must be tuned on a vehicle-to-vehicle basis, necessitating the use of a flexible and customizable vehicle simulation software, as well as an abundance of tuning data.

The Future Automotive Systems Technology Simulator (FASTSim), developed by the National Renewable Energy Laboratory (NREL), is a high-level advanced vehicle powertrain systems analysis tool that provides a simple yet extremely flexible vehicle simulation platform [4]. Due to FASTSim's high-level nature, simulations execute very quickly, allowing rapid iteration. In addition, as FASTSim is written in Python, it is highly transparent and customizable by nature. These features make FASTSim an excellent choice for highly customized vehicle modeling and simulation, as is necessary for HDVs.

Parameters must be tuned through the use of real-world vehicle performance data to generate a model with sufficient fidelity to be useful. Traditionally, the threshold for vehicle model validation has been an absolute error of 3% between measured and simulated fuel economy [5]. Real-world fleet telematics is one potential source of tuning data, containing a wide range of useful data such as geospatial information, time-series measurements, and vehicle-specific metadata such as vehicle make and model. However, it does not allow for straightforward model creation, as measurements are often sparsely sampled or sometimes

entirely absent. For this reason, a high degree of data processing, where input data is manipulated and adjusted through a series of steps, is often required to derive useful model parameters from fleet telematics data.

Telematics data has previously been used to characterize and optimize fuel consumption and vehicle emissions. A study of HDVs in the Houston–Galveston Area successfully utilized telematics measurements to characterize idling and found an average idling time of 185 minutes per day for the analyzed dataset [6]. Work by Mane, Djordjevic, and Ghosh shows that HDV telematics data can be leveraged to construct a framework for incentivizing HDV drivers to adopt more fuel efficient driving behavior [7]. Telematics data has also been used to identify potential for fuel economy improvement through HDV “platooning,” where multiple vehicles drive in close proximity in order to minimize aerodynamic drag energy losses [8]. This research demonstrates that telematics data is an extremely useful resource for the characterization and optimization of HDV fuel economy.

Other research extracted fuel economy information from telematics data without the use of vehicle modeling and simulation. One paper studied utilizes Kalman Filters created from telematics data to predict fuel economy for different hypothetical departure times over a predetermined route [9]. Another paper explores the use of geospatial data and CAN bus measurements to predict fuel economy using machine learning [10]. Both of these rely on the usage of collected data to calculate fuel economy measurements, rather than extrapolation to hypothetical drive cycles, as is possible with vehicle simulation. In general, little published work has been done to use telematics data as source of vehicle model derivation and validation, despite the potential for low-cost simulation and fuel cost savings.

This paper seeks to remedy this by providing a detailed and comprehensive methodology for using a large, sparse fleet telematics dataset to derive validated HDV fuel economy models. The hypothesis of this study is then that industry-typical quality telematics data provides sufficient detail to derive validated HDV models. Due to the preliminary nature of the research, the primary aim of this paper is to prove the hypothesis that HDV models can be generated and validated using sparse and poor-quality telematics data from a large and varied fleet of vehicles. In this paper, the phrase “high-fidelity” will refer to models achieving absolute errors below this threshold, as the phrase has no consistently used definition other than a model that simulates results to a subjectively acceptable degree. A dataset consisting of approximately 100 vehicles of many vocations from regions across North America was processed using a custom algorithm to derive model parameters. Representative drive cycles from the telematics dataset were used in FASTSim vehicle simulations to validate models against measured fuel economy.

Methodology

Telematics Data Overview

The telematics dataset used in this study consists of data from approximately 100 HDVs of multiple vocations (usage purposes) from across North America. The dataset includes time-series vehicle measurements, time-series geospatial data, and vehicle metadata. Time-series measurements encompass a majority of important signals such as velocity and fuel consumption. Time-series geospatial data includes GPS coordinates over time. Vehicle metadata provides information on the types of vehicles captured in the dataset, mostly regarding engine identification codes and vehicle vocation. Each type of data comes with its own set of unique challenges for adaptation to HDV modeling.

A challenge common to all time-series measurements in the dataset is the sparsity and uneven frequency of sample collection. Due to bandwidth limitations, the Ramer–Douglas–Peucker algorithm previously had been applied for data reduction in the sample collection process. Originally developed for image vector graphic optimization, this algorithm reduces the number of data points, maintaining overall curve detail at the expense of an acceptable level of error. The algorithm works by recursively analyzing several hypothetical linear segments between data points, selecting a piecewise linear approximation for any points that fall within a supplied perpendicular distance [11,12]. The reduction in sample count results in an uneven sampling rate and loss of temporal resolution, making the application of traditional data processing algorithms or any other process dependent on evenly spaced samples more complicated.

During data collection, certain signals were prioritized, resulting in great variations in sample counts measurement-to-measurement. The signal with the highest sample count was geospatial data, followed by measured vehicle velocity. Other signals with high sample counts included cumulative fuel consumption, engine speed, and miscellaneous engine fluid temperatures. Table 1 lists sample counts by measurement signal for the 15 most common time-series measurements. The same data is shown graphically in Figure 1, with signal names removed for simplicity.

Table 1. Sample counts for the 15 most common telematics measurement signals across entire fleet.

	Signal	Total Sample Count
1	Geospatial coordinates	1,150,367
2	Measured velocity	730,650

Table 1 - continued

	Signal	Total Sample Count
3	Fan drive state	477,016
4	Engine speed	360,669
5	Total cumulative fuel consumption	246,658
6	Ambient air temperature	205,106
7	Longitudinal acceleration	172,060
8	Acceleration (up-down)	161,135
9	Acceleration (side-to-side)	129,050
10	Transmission oil temperature	87,265
11	Acceleration (forward/braking)	78,714
12	Engine coolant temperature	76,173
13	Odometer	74,358
14	Cranking voltage	73,601
15	Torque converter lockup count	46,079

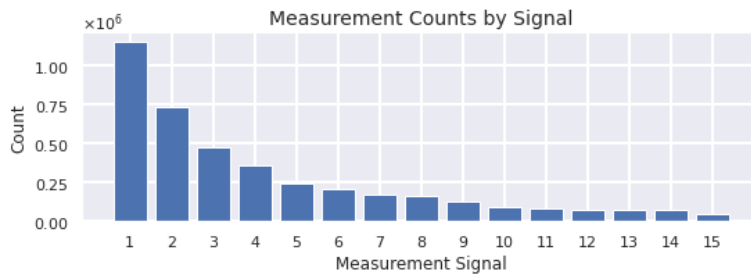


Figure 1. Sorted bar plot showing counts for the 15 most common telematics measurement signals. Note that the y-axis is in millions.

The presence and frequency of time-series measurement signals varies greatly between vehicles in the data, where some measurements are sampled very sparsely or not at all in certain vehicles. Low sample counts present a challenge for use in modeling, as for some vehicles

critical information like odometer readings can be too infrequently sampled to be usable. In this study, “sparse data” will refer to data that is sampled too infrequently and irregularly to be useful for straightforward vehicle modeling. In addition, some samples had gaps, where samples were uncollected for an extended period of time. These gaps are especially problematic for signals like vehicle velocity, where signal details critical to fuel economy may be missing. Moreover, information about power takeoff loads, ambient environment, auxiliary loads, and some other operating conditions influential to fuel economy were missing or insufficiently captured in the dataset for use in modeling.

The dataset consists of vehicles from across North America, with most geospatial data points located in the United States and some in southern Canada. Geospatial data consisted of time-series latitude and longitude measurements, and had the highest sample count of all time-series signals in the dataset. Due to this, the primary challenge presented by the geospatial data was the amount of processing required to derive useful information rather than sample sparsity.

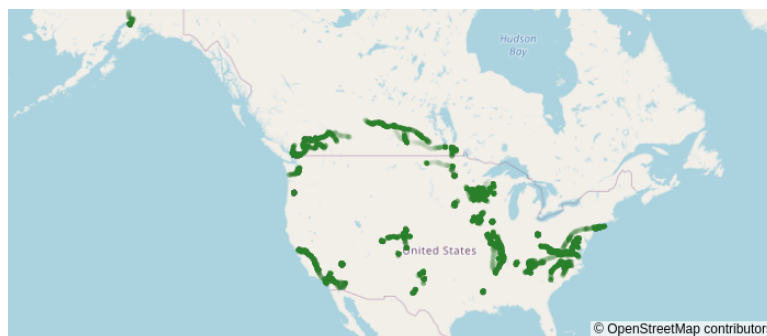


Figure 2. Map of all telematics geospatial data.

Obtained by augmenting available vehicle information with National Highway Traffic Safety Administration (NHTSA) data [13], the metadata includes details on the HDVs in the telematics dataset, providing engine parameters, transmission parameters, gross vehicle weight rating, and other useful information. Vehicles from a wide variety of vocations were captured in

the dataset, including coach and school buses, construction vehicles, refuse trucks, emergency service vehicles, and utility company service vehicles. This provided a diverse set of HDV telematics data to work with.

Model Derivation

As a starting point, the two pre-validated HDV models provided with FASTSim were used to create a base model. Many model parameters remained the same between the two and were reused in the base model. Some parameters were not the same for both models so the average value of the two parameters was used in the base model instead, where applicable. The result was the creation of a generic HDV model, but simulation results were poor. This was expected due to the wide variety of HDV characteristics.

Dynamic model variation was implemented using telematics data where possible to provide more representative parameter values. Vehicle mass, number of wheels, and other parameters were adjusted from the defaults using vehicle metadata, resulting in much higher fidelity simulation performance, as detailed in the ‘Results and Discussion’ section.

The use of vehicle metadata was enough to produce high-fidelity models. However, there are vehicles where metadata is incomplete or insufficient for deriving models. This will be the subject of future work, as described in the ‘Conclusions’ section.

Drive Cycle Derivation

Along with vehicle models, speed-by-time drive cycles are critical inputs to a backward-looking simulation, establishing the driving conditions and serving as the framework for iterative calculation. Drive cycles derived from vehicle speed measurements can be used to perform a simulation in order to validate vehicle models, allowing measured and simulated vehicle

performance to be compared directly. These vehicle speed measurements, however, must adequately represent the true driving behavior for use as a model validation drive cycle. This necessitates extensive drive cycle processing for sparse velocity signals. A summary of all velocity signal processing steps described in this section can be found in Table 2. A flowchart of velocity signal processing depicting all steps in detail is shown in Figure 3.

Table 2. Velocity signal processing summary table.

Velocity Signal Processing Summary		
Step		Step Description
0	Preprocessing	<ul style="list-style-type: none"> • Organize data for further processing • Derive secondary velocity signal from geospatial data
1	Stop Capturing	<ul style="list-style-type: none"> • Apply discontinuity correction algorithm and set small velocities to zero
2	Signal Fusion	<ul style="list-style-type: none"> • Fuse both velocity signals into one using custom algorithm
3	Upsampling	<ul style="list-style-type: none"> • Resample and interpolate signal to regular 1 Hz frequency
4	Smoothing	<ul style="list-style-type: none"> • Apply Savitzky–Golay filtering to smooth signal while retaining fine detail

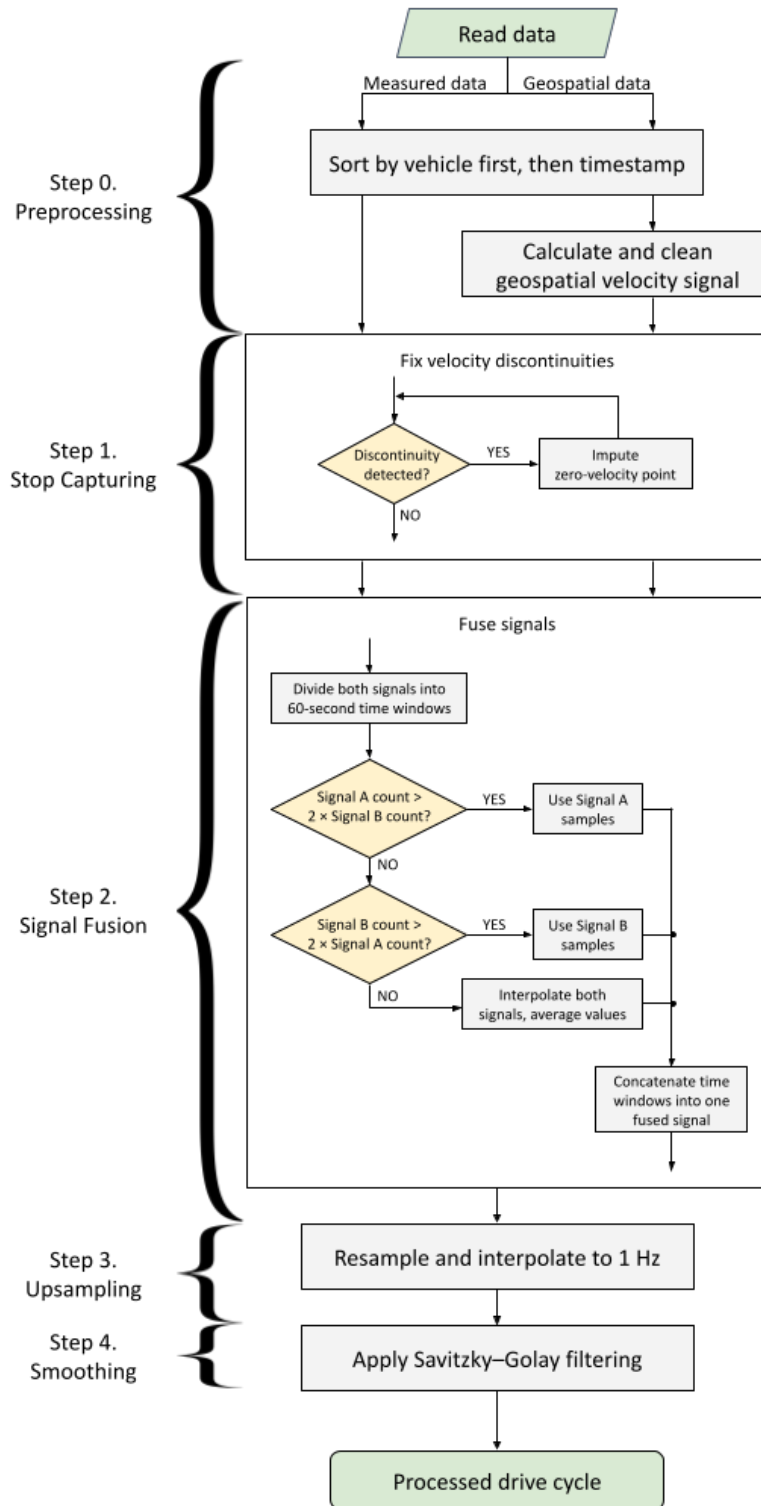


Figure 3. Velocity signal processing summary flowchart.

As the telematics measurements include many driving sessions over the course of multiple months, slices of data were selected for simulation. The two requirements to select data slices were a duration of around 30 to 60 minutes, and having at least 5 samples of each signal important to fuel economy derivation (odometer readings and cumulative fuel consumption).

Initially, the measured velocity signal was used as drive cycles for model validation. It quickly became apparent that large gaps in the velocity signal were causing issues with simulation results. While FASTSim allows unevenly spaced drive cycles, large time steps from one data point to another cause breakdowns in simulation fidelity. In addition, the use of the Ramer–Douglas–Peucker algorithm and other data loss led to loss of nuanced detail in the velocity signal. Velocity signal events and details such as accelerations, velocity peaks, stops, and overall signal smoothness are highly influential in the overall vehicle fuel economy [14]. If any of these events or details are not adequately captured due to missing data, simulation fidelity will be negatively impacted.

To remedy issues caused by gaps in data, a secondary velocity signal was derived using geospatial data. Iterating through the data, geodesic distances and timestamp differences were calculated between points. By dividing calculated distances by time differences, a new geospatial velocity signal was generated. The average timestamp between each data point was used as the timestamp for the newly created velocity signal. Points derived from data with large (over 60 second) timestamp differences or unrealistically high accelerations were removed from the derived signal. The presence of a secondary signal provides details not captured in the measured velocity signal and allows for a single velocity signal to be derived. Before this is possible, velocity signal features influential to fuel economy such as stops must be accurately represented. In both velocity signals, vehicle stops are occasionally captured inadequately due to signal gaps.

At many of these gaps, both velocity signals “jump” from a moving velocity to a stopped velocity of approximately zero, or from a stopped velocity to a moving velocity, leaving expansive straight lines across gaps. These features lead to derivation of inaccurate drive cycles and must be remedied.

To better capture vehicle stops, a custom algorithm was developed to detect discontinuities in velocity. The algorithm is applied on the entirety of a velocity signal, iteratively identifying samples that meet all of the following criteria: (1) there is a sufficient gap between analyzed sample and the previous sample, (2) the velocity at the analyzed sample is at least a threshold value, and (3) the velocity at the previous sample is at most a secondary threshold value. The custom discontinuity detection algorithm also detects downward velocity drops by looking forward at the next sample, with the third criterion satisfied if the velocity at the next sample is at most a secondary threshold value. Examples of both upward and downward velocity discontinuities are shown in Figure 4, outlined with dashed boxes.

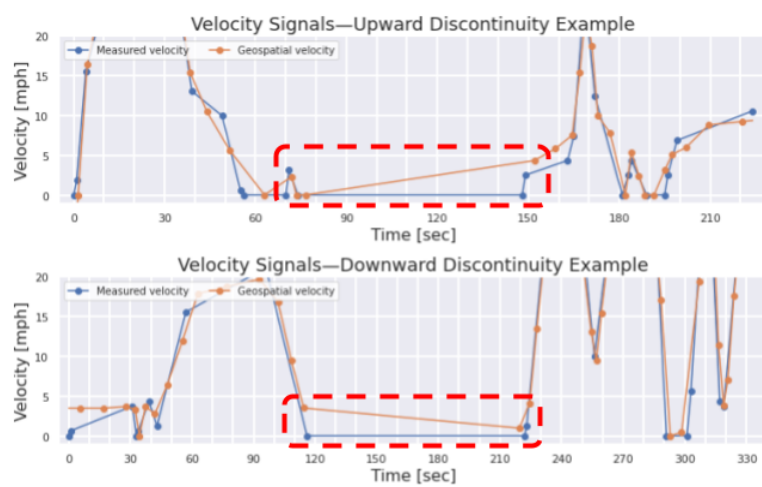


Figure 4. Examples of an upward discontinuity in velocity (top) and a downward discontinuity in velocity (bottom) commonly found at signal gaps.

After detection of velocity discontinuities, zero-velocity samples are imputed in the velocity signal. For upward velocity discontinuities, where a near-zero velocity is followed by a gap in samples before a nonzero velocity, a zero-velocity point is created in the signal just before the nonzero sample. The temporal location of this data point is found by using the acceleration value after the nonzero value if it is positive, otherwise a reasonable positive acceleration value is used. For downward velocity drops, where a near-zero velocity is preceded by a gap in samples after a nonzero velocity, this imputed zero-velocity point is placed just after the nonzero sample. The temporal location of the imputed data point is found in a similar way as in upward velocity discontinuities, but uses the acceleration before the nonzero velocity if it is negative. Otherwise, the timestamp of the imputed zero-velocity point is found using a reasonable negative acceleration value. The timestamp of imputed zero-velocity points is found by adding the time difference calculated with Equation 1 to the analyzed sample timestamp. In addition to discontinuity correction, velocities close to zero were set to zero to better capture vehicle stops.

$$\Delta t = -\frac{v}{a} \quad (1)$$

Where,

t = time relative to analyzed sample to place zero-velocity value

v = velocity of analyzed sample

a = acceleration, selected as detailed above

Once major problems with the velocity signals had been addressed, the two signals needed to be fused into a single velocity signal to be used as a drive cycle. Due to the existence of gaps, straightforward signal averaging and other simplistic methods could not be used. Instead, a custom algorithm was developed to iteratively fuse the measured and geospatial velocity signals into a single output velocity signal. The algorithm first splits the signals into 60-

second time intervals. For each time interval, the algorithm chooses one of two options to construct an output signal depending on interval sample counts. If one signal has at least twice the number of samples compared to the other signal, the signal with more samples is the output for that interval. If no signal has at least twice the number of samples than the other, the two signals are interpolated and averaged together. This algorithm fuses two signals together in a novel method robust to signal gaps, preserving detail from either signal where possible. A demonstration of the custom signal fusion methodology is shown in Figure 5.

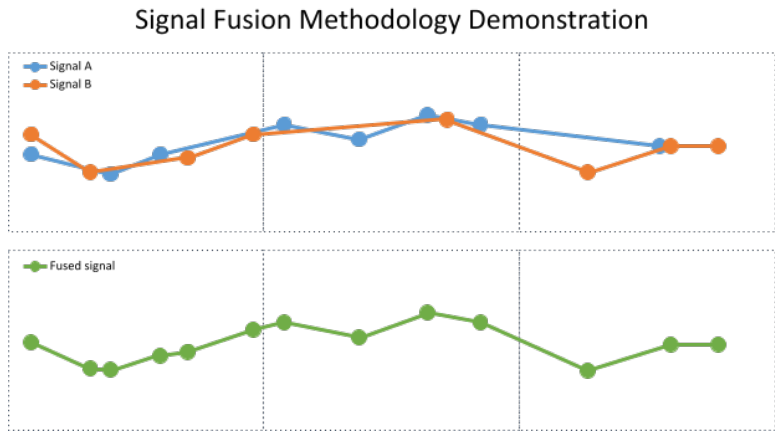


Figure 5. Signal fusion demonstration for three consecutive time intervals. From left to right: fusion by interpolated averaging, fusion using signal A, fusion using signal B.

For optimal simulation performance in FASTSim, drive cycles should not have large time steps between samples. Typically, a regular sampling rate of 1 Hz is most common for fuel economy evaluation. To implement this sampling rate, nonlinear interpolation was initially used to upsample the fused signal, but was found to produce undesirable features such as velocity peaks and valleys when applied over large gaps, so linear time-based interpolation was used instead. Upsampling to a regular frequency also enables the utilization of traditional signal processing techniques, such as Savitzky–Golay filtering.

Savitzky–Golay filtering is often used for signal smoothing due to its simplicity and appropriate handling of signal endpoints. However, it requires evenly spaced samples, like many other digital filters. After upsampling, the velocity signal could be smoothed using this algorithm. Smoothing parameters such as window size and polynomial order were carefully selected to be 15 seconds and 3rd order, respectively, to preserve as many fuel-economy-critical details (stops, peaks, and accelerations) as possible.

The end result of velocity signal processing was transformation from one measured velocity signal and geospatial data into a single, more reliable velocity signal, easily applied as a FASTSim simulation drive cycle. A summary of velocity signal processing at each step is shown in Table 2. A flowchart visualizing signal processing at each step is shown in Figure 5.

Model Validation

To validate derived models, simulated fuel economy over processed drive cycles must be compared to measured fuel economy, with a target absolute error between the two being within 3%. Because fuel economy is calculated over the entire drive cycle, rather than at one instantaneous point, measured fuel economy needed to be calculated from sparse odometer readings and fuel consumption measurements.

Measured fuel economy can be derived by comparing samples of odometer readings and cumulative fuel consumption from the start and end of data slices. Distance traveled can be calculated by subtracting the first odometer reading in the drive cycle from the last odometer reading. Fuel consumption over a data slice can be calculated similarly by subtracting the first value of the total fuel consumption from the last fuel consumption value. Dividing distance traveled by fuel consumption provides a measured fuel economy value. However, due to sparsity of data, the first and last signal values within the data slice available may not be temporally close to data slice endpoints, which could cause measured fuel economy to be inaccurate. That is, if the available odometer and fuel consumption values are far from drive cycle endpoints, it is possible the fuel economy derived could differ from its true value. To minimize this risk, before deriving measured fuel economy, the odometer and fuel consumption signals were both linearly interpolated and linearly extrapolated using available signal processing libraries to provide measurement samples close to data slice endpoints. While alternative methods of odometer and fuel consumption interpolation were considered, linear interpolation was selected as the data was essentially linear. A minimum sample count for both signals used to derive measured fuel economy was enforced to ensure measured data adequately captured real-world behavior.

As distance traveled is not an input to FASTSim drive cycles, simulated fuel economy is derived by dividing a distance derived from the integral of the velocity by the total simulated fuel consumption. This is useful for applications where total distance is unavailable, but the telematics data provides a better indication of the true total distance traveled. This measured distance traveled was divided by simulated fuel consumption to provide an alternative simulated fuel economy with error from FASTSim distance calculation minimized.

Results and Discussion

Overall Results

Due to the vast number of vehicles and trips in the dataset, a single vehicle was selected for result analysis, however, significant reductions in absolute errors occurred for the vast majority of vehicles tested. The application of model and drive cycle derivation described in this study resulted in a simulated fuel economy with an absolute error of 1.67% when compared to measured fuel economy. This is below the traditional fuel economy model validation threshold of 3%. This shows the methodology presented in this paper can be used to successfully derive and validate HDV models. The cumulative fuel consumption over time from measurements and simulation are compared in Figure 6.

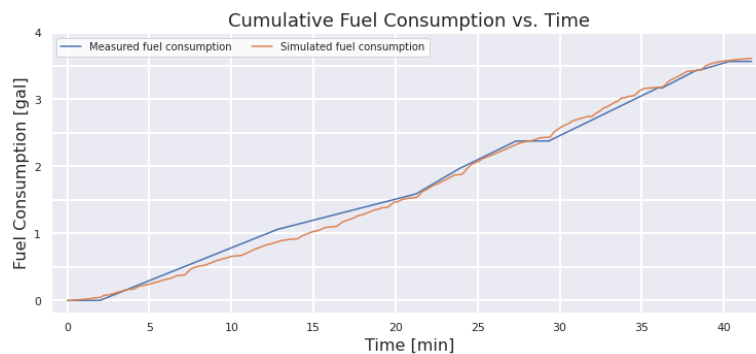


Figure 6. Comparison of measured vs. simulated fuel consumption over time.

Velocity Signal Processing Results

Derivation of a secondary geospatial velocity provided a secondary source of velocity data, however both signals had issues with sparsity and missing features. This is shown in Figure 7 through comparison of the two signals. In sections where the two signals follow each other closely, measured-versus-simulated fuel economy absolute errors are relatively low. In sections where a feature is missing from one signal due to inadequate sampling, errors are larger.

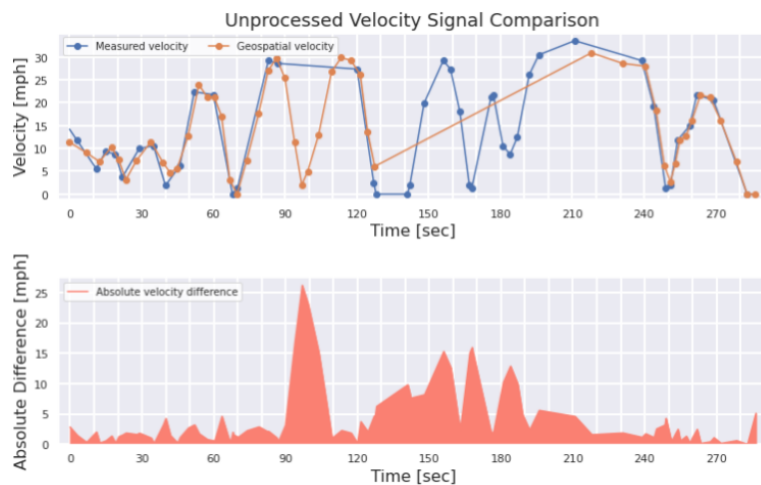


Figure 7. Comparison of unprocessed measured and geospatially-derived velocity signals.

Missing features are undesirable because they negatively affect drive cycle fidelity. This can be seen in the high absolute errors when unprocessed signals are used as simulation drive cycles. Use of unprocessed (but upsampled, for FASTSim performance) measured velocity resulted in an error of 10.55%. Similarly, use of geospatial velocity resulted in an error of 8.83%.

Discontinuities at vehicle stops originally missing from the velocity signals were recaptured using a custom algorithm. For the specific drive cycle tested, absolute errors did not noticeably decrease, likely due to lack of discontinuities. For a vehicle with more occurrences of discontinuities, application of the correction algorithm to measured velocity resulted in a reduction of absolute error by 0.22 percentage points. Application of the algorithm to geospatial

velocity resulted in a reduction of absolute error by 0.84 percentage points. Examples of stop capturing algorithm results are shown in Figure 8 and Figure 9.

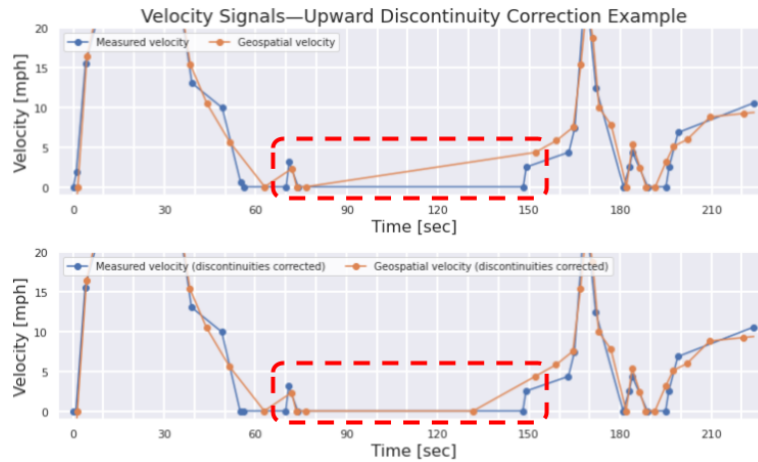


Figure 8. Results of upward discontinuity correction. Shown is an uncorrected upward discontinuity (top) and the results of applying the correction algorithm (bottom).

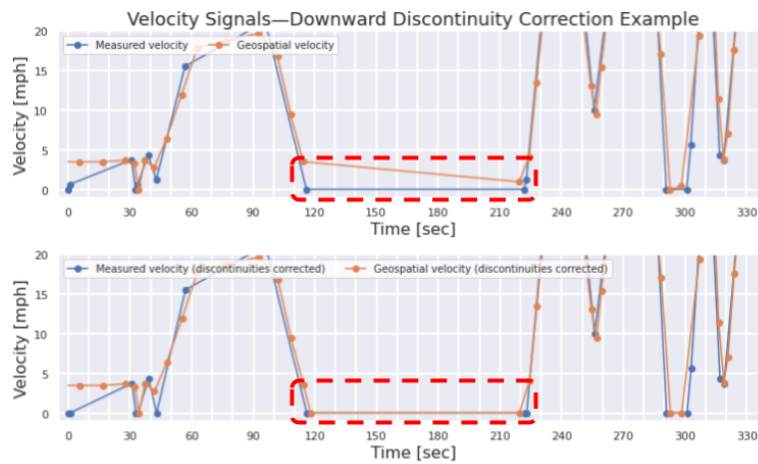


Figure 9. Results of downward discontinuity correction. Shown is an uncorrected upward discontinuity (top) and the results of applying the correction algorithm (bottom).

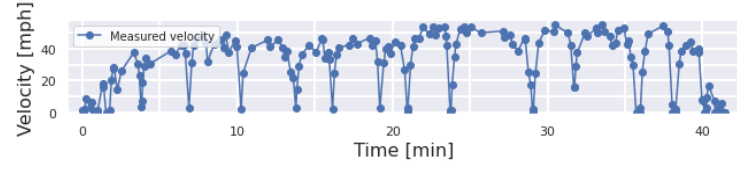
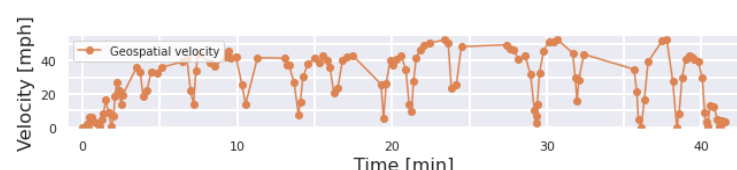
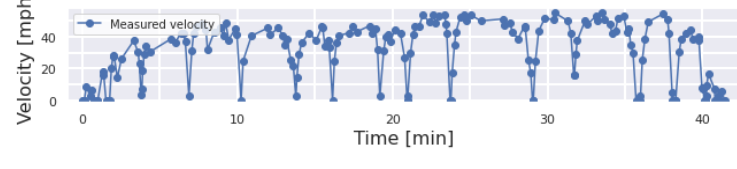
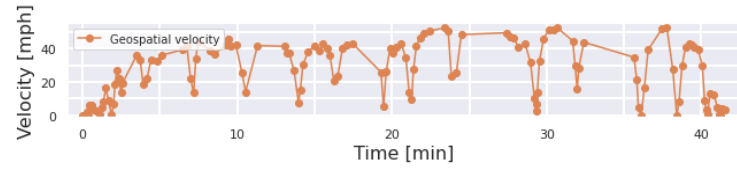
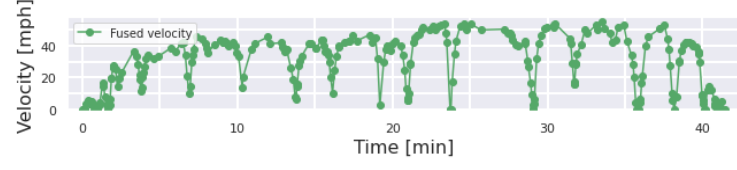
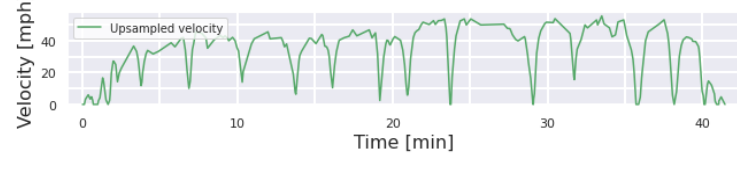
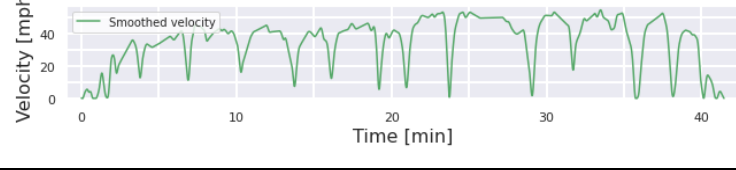
After stops were captured, a custom signal fusion algorithm was applied to transform the two velocity signals into one through consideration of sample counts in 60-second time intervals. Application of the signal fusion algorithm—resulted in an absolute error of 3.05%, a considerable

improvement from the absolute errors achieved in the previous signal processing step, 10.55% and 8.83%.

To achieve optimal simulation performance, signals must be upsampled before use in FASTSim. For this reason, all fuel economy results shown are simulated using upsampled velocity signals. This is also why fuel economy simulation fidelity does not increase due to upsampling.

As fuel economy results are impacted considerably by signal smoothness, Savitzky–Golay filtering was applied to the fused velocity signal. This resulted in a reduction in measured-versus-simulated fuel economy absolute error from 3.05% to 1.67%. A summary of velocity signal processing and associated simulation results is shown in Table 3.

Table 3. Tabulated velocity signal processing results.

Velocity Signal Processing Results				
Step	Signal	Plots	FE Results Measured FE: 7.57 MPG	
0	Preprocessing	Measured		Simulated FE: 6.77 MPG (10.55% error)
	Geospatial		Simulated FE: 8.24 MPG (8.83% error)	
1	Stop Capturing	Measured		Simulated FE: 6.77 MPG (10.57% error)
		Geospatial		Simulated FE: 8.24 MPG (8.83% error)
2	Signal Fusion		Simulated FE: 7.34 MPG (3.05% error)	
3	Upsampling		Simulated FE: 7.34 MPG (3.05% error)	
4	Smoothing		Simulated FE: 7.44 MPG (1.67% error)	

Conclusions

This study showcases the methodology and results from utilizing real-world telematics data for heavy-duty vehicle (HDV) modeling and simulation in the Future Automotive Systems Technology Simulator (FASTSim). Models were derived using FASTSim pre-validated HDV models as a starting point, from which a basic HDV model was constructed and dynamically varied using vehicle metadata available. Models were simulated using vehicle velocity signals from both velocity measurements and geospatial data. Simulated fuel economy was compared to a derived measured fuel economy value and was found to achieve an absolute error of 1.67%, within the traditional fuel economy model validation threshold of 3%. It has been shown that sparse telematics data can be used to generate validated HDV models that allow for the extrapolation of fuel economy performance to hypothetical drive cycles and the examination of the impact of different HDV characteristics on fuel economy.

When working with a fleet of vehicles across large regions under many different vocations, instrumentation of vehicles beyond simple telematics data recorders quickly grows prohibitively expensive. The methodology presented in this paper can be used to derive and validate fuel economy models when physical experimentation is unacceptably impractical, as is the case with fleets of diverse HDVs with widely varied vocations. However, before wide-scale application of this methodology, additional work must be performed to expand the methodology to more vehicles in the dataset and further improve performance.

Much of the work in this study focused around the restoration of data quality lost during data collection. The amount of data processing could be cut down substantially while increasing the fidelity of data by increasing sample count. While sample count seems to be prioritized by measurement, with important signals such as geospatial coordinate, vehicle velocity, and fuel

consumption sampled much more often than other signals—as shown in Figure 1—there often was not enough data within a slice to validate a vehicle model. Vehicle stops were often inadequately sampled, resulting in sudden velocity signal discontinuities, requiring rectification. Where possible, the sparsity of important measurements should be minimized.

To improve on the work done in this study, it is recommended that focus be placed on additional methods of deriving model parameters. Time-series measurements such as engine torque and speed could provide ways of deriving vehicle parameters through theoretical relationships. In addition, the use of sliding windows rather than static windows in the signal fusion data processing step could provide an improvement in performance. As this is a proof-of-concept study, future work should also include refinement and rigorous analysis of the application of this methodology with a wider set of vehicles.

Chapter II References

1. Williams, N. and Murray, D., “An Analysis of the Operational Costs of Trucking: 2020 Update,” American Transportation Research Institute, 2020.
2. U.S. Environmental Protection Agency, “Inventory of U.S. Greenhouse Gas Emissions and Sinks: 1990-2019”, EPA Report, 2021.
3. U.S. Environmental Protection Agency, “EPA Announces the ‘Clean Trucks Plan’”, EPA Fact Sheet, EPA-420-F-21-057, 2021.
4. Brooker, A., Gonder, J., Wang, L., Wood, E. et al., "FASTSim: A Model to Estimate Vehicle Efficiency, Cost and Performance," SAE Technical Paper 2015-01-0973, 2015, doi:10.4271/2015-01-0973.

5. Delgado, O.F., Clark, N.N., and Thompson, G.J., “Modeling transit bus fuel consumption on the basis of cycle properties,” *J. Air Waste Manag. Assoc.* 61(4):443–452, 2011, doi:10.3155/1047-3289.61.4.443
6. Farzaneh, R., Johnson, J., Jaikumar, R., Ramani, T., and Zietsman, J., “Use of Vehicle Telematics Data to Characterize Drayage Heavy-Duty Truck Idling,” *Transportation Research Record* 2674(11):542–553, 2020, doi:10.1177/0361198120945990
7. Mane, A., Djordjevic, B., and Ghosh, B., “A data-driven framework for incentivising fuel-efficient driving behaviour in heavy-duty vehicles,” *Transportation Research Part D: Transport and Environment* 95:102845, 2021, doi:10.1016/j.trd.2021.102845
8. Lammert, M.P., Bugbee, B., Hou, Y., Mack, A. et al., “Exploring Telematics Big Data for Truck Platooning Opportunities,” *SAE Technical Paper* 2018-01-1083, 2018, doi:10.4271/2018-01-1083.
9. Kolmanovsky, I., McDonough, K., and Gusikhin, O., “Estimation of Fuel Flow for Telematics-Enabled Adaptive Fuel and Time Efficient Vehicle Routing,” 2011 11th International Conference on ITS Telecommunications, 139–144, 2011, doi:10.1109/ITST.2011.6060041
10. Zeng, W., Miwa, T., and Morikawa, T., “Exploring Trip Fuel Consumption by Machine Learning from GPS and CAN Bus Data,” *Journal of the Eastern Asia Society for Transportation Studies* 11:906–921, 2015, doi:10.11175/easts.11.906
11. Ramer, U., “An Iterative Procedure for the Polygonal Approximation of Plane Curves,” *Computer Graphics and Image Processing* 1(3), 1972, doi:10.1016/S0146-664X(72)80017-0

12. Douglas, D.H. and Peucker, T.K., “Algorithms for the reduction of the number of points required to represent a digitized line or its caricature,” *Cartographica* 10(2):112–122, 1973, doi:10.3138/fm57-6770-u75u-7727
13. National Highway Traffic Safety Administration, <https://www.nhtsa.gov/>
14. Asher, Z.D., Trinko, D.A., and Bradley, T.H., “Increasing the Fuel Economy of Connected and Autonomous Lithium-Ion Electrified Vehicles,” in: Pistoia, G. and Liaw, B., eds., *Behaviour of Lithium-Ion Batteries in Electric Vehicles: Battery Health, Performance, Safety, and Cost*, Springer International Publishing, Cham, ISBN 9783319699509: 129–151, 2018, doi:10.1007/978-3-319-69950-9_6

CHAPTER III

PROJECTING LANE LINES FROM HIGH-DEFINITION MAPS FOR AUTOMATED VEHICLE PERCEPTION IN ROAD OCCLUSION SCENARIOS

This chapter consists of work developed in a project sponsored by the Michigan Translational Research & Commercialization (MTRAC) program and submitted for review as a conference publication at SAE World Congress Experience 2023 authored by Kyle Carow, Parth Kadav, Johan Fañas Rohas, and Zachary D. Asher. Please note that much of this chapter is presented verbatim from this publication. Kyle Carow led the data collection, methodology development, and publication authorship with essential assistance from all co-authors.

Abstract

Contemporary ADS and ADAS localization technology utilizes real-time perception sensors such as visible light cameras, radar sensors, and lidar sensors, greatly improving transportation safety in sufficiently clear environmental conditions. However, when lane lines are completely occluded, the reliability of on-board automated perception systems breaks down, and vehicle control must be returned to the human driver. This limits the operational design domain of automated vehicles significantly, as occlusion can be caused by shadows, leaves, or snow, which all occur in many regions. High-definition map data, which contains a high level of detail about road features, is an alternative source of the required lane line information. This study details a novel method where high-definition map data are processed to locate fully occluded lane lines, allowing for automated path planning in scenarios where it would otherwise be impossible. A proxy high-definition map dataset with high-accuracy lane line geospatial positions was generated for routes at both the Eaton Proving Grounds and Campus Drive at Western Michigan University (WMU). Once map data was collected for both routes, the WMU Energy Efficient and Autonomous Vehicles Laboratory research vehicles were used to collect

video and high-accuracy global navigation satellite system (GNSS) data. The map data and GNSS data were fused together using a sequence of data processing and transformation techniques to provide occluded lane line geometry from the perspective of the ego vehicle camera system. The recovered geometry is then overlaid on the video feed to provide lane lines, even when they are completely occluded and invisible to the camera. This enables the control system to utilize the projected lane lines for path planning, rather than failing due to undetected, occluded lane lines. This initial study shows that utilization of technology outside of the norms of automated vehicle perception successfully expands the operational design domain to include occluded lane lines, a necessary and critical step for the achievement of complete vehicle autonomy.

Introduction

The US Centers for Disease Control and Prevention (CDC) and the US National Highway Traffic Safety Administration (NHTSA) report that motor vehicle accidents account for nearly 40,000 US deaths in 2019 and comprised the 13th leading cause of death in the US in 2016 and 2017 [1-3]. CDC data indicates the estimated cost of US motor vehicle fatalities to be about \$390 billion in 2019 when accounting for both medical costs and economic productivity losses [4]. Additionally, motor vehicle traffic crashes consistently rank the 7th greatest contributor to years of life lost, as they disproportionately cause more deaths to younger people [3]. In response to the great cost of motor vehicle accidents, the emerging technological field of vehicle automation seeks to mitigate motor vehicle accidents and injuries. While in the future, high levels of autonomy through Automated Driving Systems (ADS) and Autonomous Vehicles (AVs) will be available, some vehicles available today offer Advanced Driver Assistance Systems (ADAS) features to improve safety.

ADAS features such as lane departure warning (LDW), lane keeping assistance (LKA), and lane centering assistance (LCA) can dramatically improve motor vehicle safety. LDW reduces single-vehicle, sideswipe, and head-on injury crashes by 21% [5]. It is estimated by Benson et al. that LDW and LKA could have prevented about 520,000 crashes in 2016, associated with about 190,000 injuries. They also report that ADAS technology at large has the potential to mitigate 40% of all passenger vehicle crashes, and about 30% of all crash-related deaths [6]. In terms of the cost of motor vehicle accidents, Harper, Hendrickson, and Samaras estimate that incorporating ADAS features into the entire US light-duty vehicle fleet would lead to an annual net savings of between \$4 billion to \$215 billion when considering the cost of the technology and the cost of motor vehicle crash injuries and deaths [7]. ADAS has also been demonstrated to be useful for enabling energy efficiency improvements for individual vehicles [8-11]. But, these ADAS features have an operational design domain (ODD) limited to unoccluded lane lines, as they work by using real-time perceptive sensors such as computer vision (CV) to detect road features, primarily lane lines, which are normally visible in clear driving environments [12-14]. However, many vehicle crashes occur in inclement weather, where road features can be completely occluded by snow or ice.

In fact, while vehicles regularly travel less in winter seasons as evidenced by seasonal vehicle-miles-traveled trends, the inclement driving environment conditions associated with the colder seasons leads to increased vehicle accidents and fatalities [15-16], as seen in Figure 10. According to the Federal Highway Administration (FHWA), approximately 21% of all vehicle crashes in the US from 2007 to 2016 were weather-related [17]. The ODD of vehicle autonomy does not effectively include inclement weather conditions where lane lines are occluded, as perception systems lack the necessary input to determine road geometry [18-19]. It is estimated

that approximately 70% of US roads are located in snowy regions, meaning for higher levels of automation throughout the continental US, the ODD of automated vehicles must be expanded to handle roads occluded by snow and ice [20]. How can the ODD be expanded to occluded lane-line scenarios, thus enabling higher levels of driving autonomy?

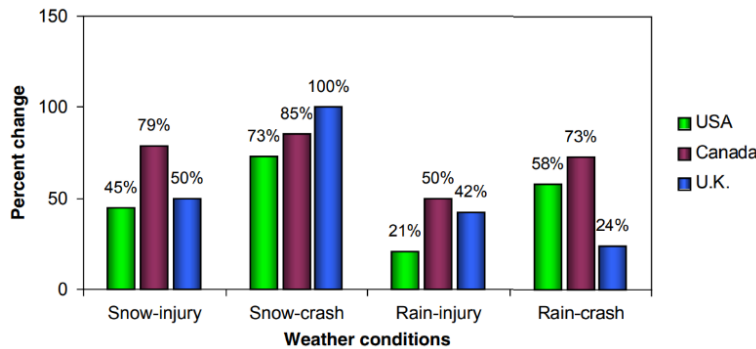


Figure 10. Percent changes in crash rate due to inclement weather (directly from [16]).

Greater connectivity of the vehicle to the internet of things has great potential to aid automated vehicle technologies [21]. Specifically, the utilization of high-definition maps for localization of the ego vehicle has been shown to have great potential. High-definition maps are datasets that contain highly detailed regional data, far surpassing the minimum level of detail required for road network route planning as is available in standard maps. Notably, they can include the positions of road features such as traffic signs, road shape, and lane lines [22]. Little research has focused on occluded lane line scenarios, despite the potential for safety improvements. Poggenhans, Salscheider, and Stiller present a method of localization where high-definition maps are queried with estimated lane line locations to determine the true ego vehicle position. This method begins to lose accuracy in inclement weather and relies on real-time perception of road markings, thus would likely break down when lane lines are occluded, similar to the current state-of-the-art [23]. VSI Labs described use of high-definition maps in their study, where lane lines were used directly as an input to the ego vehicle control system, but their exact

methodology appears to be proprietary and confidential [24]. High-definition maps have the potential to increase the level of automation of ADS technology, but no method exists with sufficient ability to operate in occluded lane line scenarios. This paper describes a novel methodology for utilizing high-definition map data to expand the ODD of ADS and ADAS technologies to occluded lane line scenarios.

Methods

Data Collection

To avoid the cost of a high-definition map service, data was collected manually over two distinct routes, and proxy high-definition maps were created. The first route was selected to be a portion of the Eaton Proving Grounds main test track in Marshall, Michigan, shown in Figure 12. This route contained a straight section, followed by a curve to the left surrounded by trees. The second route was selected to be the Campus Drive loop near Western Michigan University's Parkview Campus, shown in Figure 13. This route consisted of a winding, continuously curving road segment. Several hundred extremely high-precision geospatial points were collected along the right and left lane lines for each route using a Trimble Catalyst DA2 GNSS receiver, as shown in Figure 11, which advertises an accuracy of 2 centimeters when using the Catalyst 1 subscription level. High accuracy geospatial data was especially important in this study, as the lane lines are relatively small and any substantial displacement degrades the results significantly.



Figure 11. Trimble Catalyst DA2 GNSS receiver (directly from [25]).

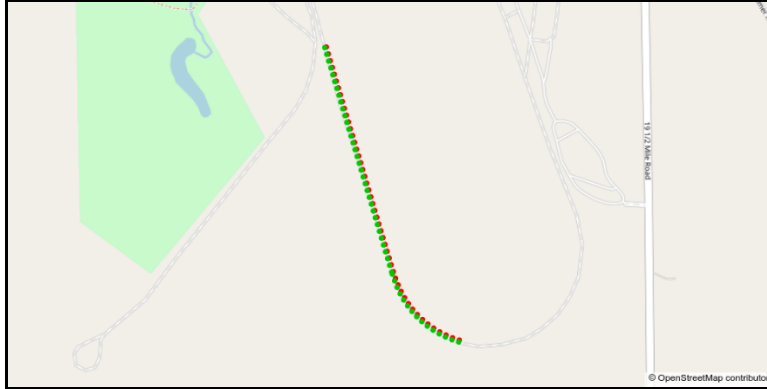


Figure 12. Collected lane line points at the Eaton Proving Grounds test track.

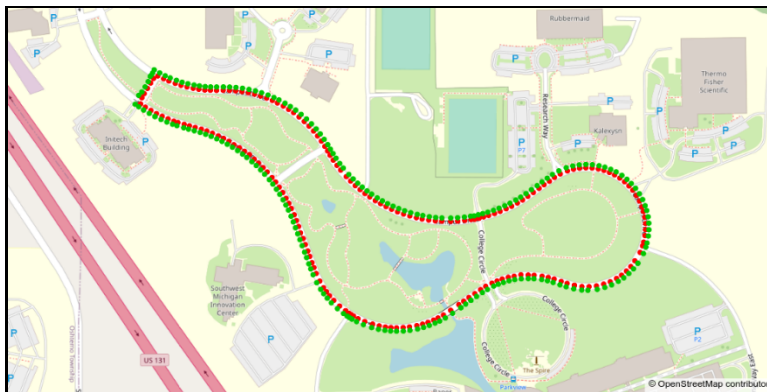


Figure 13. Collected lane line points over the Campus Drive loop.

To achieve smoothed lane lines over both routes, all lane lines points were put in order along the route and cubically interpolated, resulting in many equidistant geospatial points. This interpolation of geospatial points should theoretically be mapped to true distances before interpolation, as the length of latitude and longitude increments is not constant over Earth, but at this relatively small scale the error is negligible. After this step, the proxy high-definition maps for both routes were complete.

The Western Michigan University (WMU) Energy Efficient and Autonomous Vehicles Laboratory research vehicles, shown in Figure 14, were then used to collect data over both routes. The Kia Niro Hybrid was used at the Eaton Proving Grounds, and the Kia Soul Electric

Vehicle was used to collect data over the Campus Drive route. Both vehicles were equipped with the same sensor suite, including but not limited to a Stereolabs ZED 2i stereo camera (Figure 15) and a Swift Navigation Duro Inertial RTK GNSS receiver (Figure 16), which advertises 4 centimeter accuracy when using the Skylark Precise Positioning service.



Figure 14. WMU EEA V Lab research vehicles.



Figure 15. Stereolabs ZED 2i stereo camera (directly from [26]).



Figure 16. Swift Navigation Duro Inertial RTK GNSS receiver (directly from [27]).

The ego vehicles were driven over the selected routes and data was collected from the camera, IMU, and GNSS receiver using the Robot Operating System (ROS). The ego vehicle was operated by a human driver, driving no more than 25 miles per hour. A data processing

methodology was developed to project lane lines over the camera feed utilizing the sensor data in various ways. Figures 16 and 17 show unprocessed sample camera frames from the Eaton Proving Grounds route and the Campus Drive Route, respectively.



Figure 17. Frame from image data collected over the Eaton Proving Grounds test track.



Figure 18. Frame from image data collected over the Campus Drive loop.

Data Processing

There are several coordinate frames relevant to this study. The lane line points are in the WGS 1984 coordinate system, consisting of latitude, longitude, and altitude. The cartesian coordinate frame of the vehicle is chosen to be z-forward, x-right, and y-down to be consistent with the coordinate systems of both the ZED stereo camera and the freely-available OpenCV

Python/C++ library. The overarching goal of data processing was to transform the geospatial points of both the lane lines and the ego vehicles into the local coordinate frame aligned with the camera, where the data could be overlaid and given to camera-based path planning systems. To do this, several coordinate transformations were necessary. All points were transformed from the WGS 1984 coordinate system to the north, east, down (NED) coordinate system, with the origin specified as the ego vehicle GNSS base station. This transformation brings the data from a global coordinate system into a right-handed local coordinate system. The NED coordinate system was chosen over the east, north, up (ENU) coordinate system in order to best match the coordinate system of the camera, shown in Figure 19, as well as the pinhole camera model coordinate system used by OpenCV, where the x-axis is right, the y-axis is down, and the z-axis is forward. The coordinate transformation of the lane line points to the NED system was handled by the pymap3d Python library.

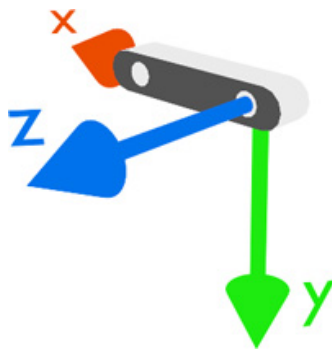


Figure 19. Stereolabs ZED camera coordinate frame (directly from [28]).

Next, the local NED coordinate frame needed to be rotated to align with the ego vehicle heading. However, the sampling rate of the vehicle heading was insufficient for real-time alignment of the lane line points, especially for curved road segments. To better align with the vehicle over time, IMU data were then used to interpolate the vehicle heading. Measurements of the angular velocity were multiplied by elapsed time to provide angular adjustments to the

heading as shown in Equation 2. The angle θ_0 is the heading reported by the sensor, $\dot{\theta}$ is the angular velocity about the IMU (up) z-axis, Δt is the time elapsed from the heading sample, and θ is the adjusted heading. In summary, the coordinate frame was rotated by the adjusted heading θ about the down axis. This can be represented as shown in Equation 3, where the x-axis is right, the y-axis is down, and the z-axis is forward. This rotation is shown graphically in Figure 20.

The equations for the transformation are given as

$$\theta = \theta_0 - \dot{\theta} \Delta t \quad (2)$$

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \\ \cos \theta & \sin \theta & 0 \end{bmatrix} \begin{bmatrix} N \\ E \\ D \end{bmatrix} \quad (3)$$

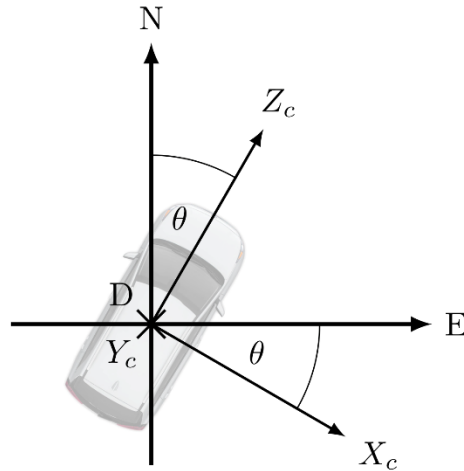


Figure 20. NED to XYZ coordinate frame rotation.

Once all points were rotated to the new XYZ coordinate system, offsets in each direction were used as appropriate to translate from the base station GNSS receiver to the left camera. Once this was complete, all lane line points had been transformed from the global WGS 1984 coordinate system to the local XYZ coordinate system used by the camera. Next, the points needed to be projected onto the camera image feed, applying a maximum-distance threshold if desired. To do this, the pixel location (u, v) of each point must be found, which can be done

given the XYZ coordinates and intrinsic camera properties retrieved from the camera metadata. The point projection method equations below are summarized from the OpenCV documentation, following the notation in Figure 21 [29].

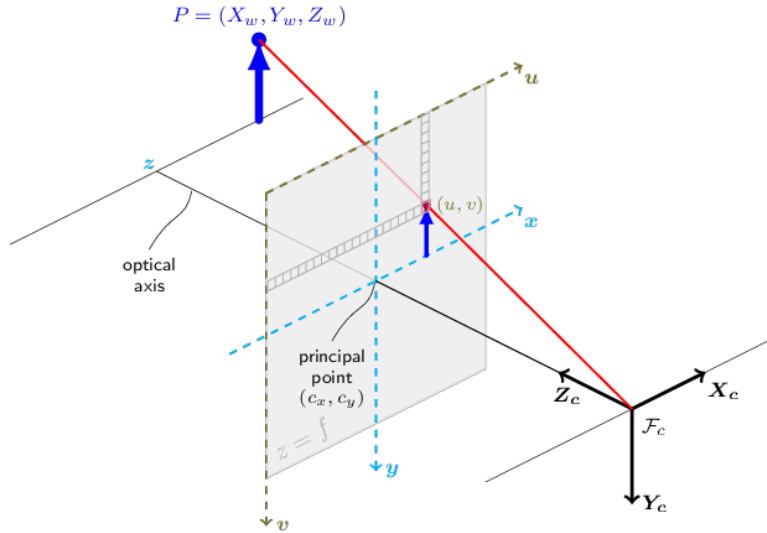


Figure 21. OpenCV pinhole camera model coordinate frame (directly from [29]).

First, the ratios of the x- and y-coordinates to the z-coordinate of the point are calculated, and the hypotenuse of the x- and y-coordinates r is found following

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} X_c/Z_c \\ Y_c/Z_c \end{bmatrix} \quad (4)$$

$$r^2 = x'^2 + y'^2. \quad (5)$$

The relevant camera properties were the radial distortion coefficients $k_1, k_2,$ and $k_3,$ the tangential distortion coefficients p_1 and $p_2,$ the principal point coordinates c_x and $c_y,$ and the focal lengths f_x and $f_y.$ These properties can be used to find an intermediate result in both the x- and y-directions. The additional radial distortion coefficients $k_4, k_5,$ and k_6 and thin prism distortion coefficients $s_1, s_2, s_3,$ and s_4 were not necessary and are excluded here. This gives Equation 6 as

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} x'(1 + k_1r^2 + k_2r^4 + k_3r^6) + 2p_1x'y' + p_2(r^2 + 2x'^2) \\ y'(1 + k_1r^2 + k_2r^4 + k_3r^6) + p_1(r^2 + 2y'^2) + 2p_2x'y' \end{bmatrix} \quad (6)$$

Finally, the projected pixel location (u, v) of the point is found using the focal lengths f_x and f_y and the principal point x- and y-coordinates c_x and c_y by Equation 7 by

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} f_x x'' + c_x \\ f_y y'' + c_y \end{bmatrix} \quad (7)$$

Sequential application of Equation 3 through Equation 6 results in the appropriate pixel locations of all 3-dimensional points projected onto the 2-dimensional camera image. This is the basis for augmentation of a camera-frame path planning algorithm with high-definition map lane line geometry, without the need for line-of-sight to the lane line.

Additionally, for a sense of distance, a radius or diameter r_{point} for each point to be drawn can be calculated by dividing a size scaling factor s by the distance of the point from the origin, and rounding the result to the nearest integer. This is shown in Equation 7 and Equation 8, defined as

$$dist = \sqrt{X_c^2 + Y_c^2 + Z_c^2} \quad (8)$$

$$r_{point} = \text{round}(s/dist) \quad (9)$$

Alternatively, as selected for this study, a polygonal chain can be drawn connecting all points for a result that most closely imitates lane lines. This is simply done by connecting the points with line segments.

Artificial Occlusion of Lane Lines

For result comparison, lane lines were extracted from the data using CV techniques. Hue/saturation/lightness thresholding masks were created to isolate yellow and white lane line pixels. While this did successfully identify the appropriate pixels, many pixels from the sky and

background passed through the threshold. To remedy this, a second processing step was taken where a region of interest (ROI) mask crops the image to a trapezoidal shape. The values of the threshold masks and the shape of the ROI mask differed between the two datasets due to different camera orientation and different lighting conditions. This provided a basic methodology for extraction of lane line pixels.

Data along both routes could not be collected when lane lines were occluded, e.g. by collected snow or leaves. For this reason, lane lines in the camera feed were artificially occluded using further video postprocessing, thereby simulating road coverage conditions. The concept of postprocessing the camera feed in order to simulate different conditions is not entirely novel – Rubaiyat, Qin, and Alemzadeh utilized a similar method in their study in order to analyze resilience of autonomous vehicles to disturbed camera input [30]. The specific postprocessing methodology chosen was to use extracted lane line pixels and draw circles at each, using approximately the same color as the road surface. This resulted in the camera feed now having fully occluding lane lines, making normal detection of lane lines through CV or machine learning techniques impossible. Indeed, if one applies the same lane detection CV technique to detect lane lines on these processed images, no detections would be possible. The result of this CV image processing is shown for the Eaton Proving Grounds test track route in Figure 22, and for the Campus Drive loop route in Figure 23. Lane line geometry extracted from the proxy high-definition map was drawn on top of these images to demonstrate that the methodology is entirely independent of camera-based lane line detections. The occluded lane lines shown in subsequent figures does not perform perfectly, but as this was just for demonstration purposes current results are satisfactory.



Figure 22. Eaton Proving Grounds test track camera frame with lane lines artificially occluded, before and after.



Figure 23. Campus Drive camera frame with lane lines artificially occluded, before and after.

Metrics of Evaluation

Data Accuracy Metrics

The most important metric for ensuring the quality of collected geospatial data was horizontal (tangential to ellipsoid surface) and vertical (perpendicular to ellipsoid surface) accuracy. These metrics are reported by both the Trimble Catalyst DA2 handheld sensor as well as the Swift Navigation Duro Inertial sensor. The Swift Navigation Duro Inertial sensor also operates in several accuracy modes, depending on satellite visibility and mobile network conditions. The highest accuracy mode is RTK fixed, determined by the number of visible satellites and whether the rover sensor is receiving corrections from the base station sensor. The

goal of data collection was to keep collected lane line points accurate to 2 centimeters, and to keep the ego vehicle GNSS system operating in RTK fixed mode, accurate to 4 centimeters.

Lane Line Overlay Metrics

At present, no metrics exist for quantitative analysis of projected lane line accuracy. The scope of this study did not include development of such a quantitative metric, instead a qualitative analysis showcasing the current strengths and areas of improvement was selected. The exploration and development of a quantitative metric is proposed as a topic for future study.

Results

Data Accuracy Analysis

In the lane line data collected with the Trimble Catalyst DA2 sensor over the Campus Drive route, the horizontal accuracy of the collected lane line points never exceeded 2 centimeters, and the vertical accuracy never exceeded 5 centimeters. Weather conditions and mobile network strength were poorer at the Eaton Proving Grounds test track, so despite multiple attempts, the maximum horizontal accuracy reached 7 centimeters, and the maximum horizontal accuracy reached 25 centimeters. In general, most collected points had a horizontal accuracy between 1 to 3 centimeters and a vertical accuracy between 6 to 9 centimeters. The lane line data collected along both routes were more than sufficient to create the proxy high-definition map.

The Swift Navigation Duro Inertial sensor consistently operated in the highest accuracy fixed RTK mode for the duration of the Campus Drive route. As such, the Campus Drive route data has very good accuracy; the horizontal accuracy of the ego vehicle GNSS receiver was at most 5.0 centimeters, and the vertical accuracy was at most 7.2 centimeters. The ego vehicle GNSS receiver faced more accuracy challenges at the Eaton Proving Grounds test track,

switching from fixed RTK to float RTK mode near the end of the route, when the vehicle was surrounded by trees.

Lane Line Overlay Analysis

Eaton Proving Grounds Test Track

The projected lane lines in the straight road segment of the Eaton Proving Grounds route align very well with the true lane line locations, and little deviation is observed. This shows the methodology holds promise for determination of occluded lane lines, especially for straight roadways. Two example results from this segment are shown in Figure 24.



Figure 24. Two examples of Eaton Proving Grounds camera frames from straight road segments with high-definition map lane line data overlaid on artificially occluded lane lines. These show well-aligned lane lines.

When the ego vehicle entered the more wooded area, the Swift Navigation Duro Inertial sensor switched from the higher-accuracy RTK fixed mode to the lower-accuracy RTK float mode. This resulted in a divergence of the projected lane lines away from the true lane lines. Curves present a challenge to this method, even in RTK fixed mode, as discussed further in the following Campus Drive route results. This divergence is shown in Figure 25.

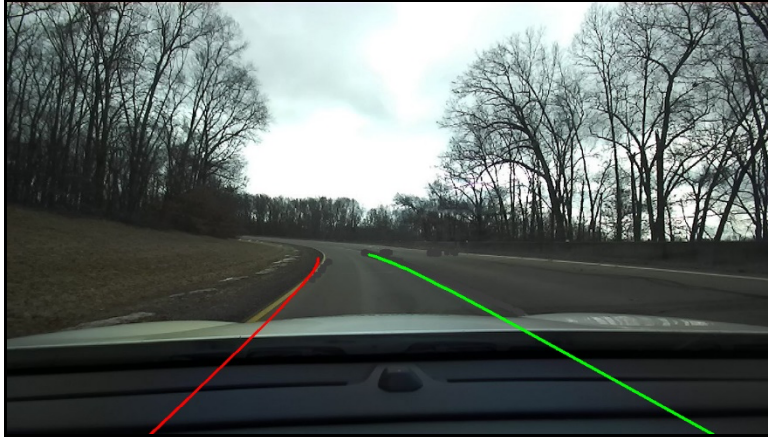


Figure 25. Eaton Proving Grounds camera frame from a curved road segment with high-definition map lane line data overlaid on artificially occluded lane lines. Deviation from true lane lines occurs in this curved road segment due to the GNSS operating in lower-precision RTK float mode and the insufficient sampling rate of the GNSS heading sensor.

Campus Drive Loop

The transformed and projected lane line geometry aligns well with the lane line pixels in the Campus Drive dataset in straight road segments as well, as shown in two examples in Figure 26. The text overlaid on the images shows the instantaneous Swift Navigation Duro Inertial sensor accuracy and operating mode. Note that the Eaton Proving Grounds dataset did not include this accuracy data due to technical limitations at the time of collection.

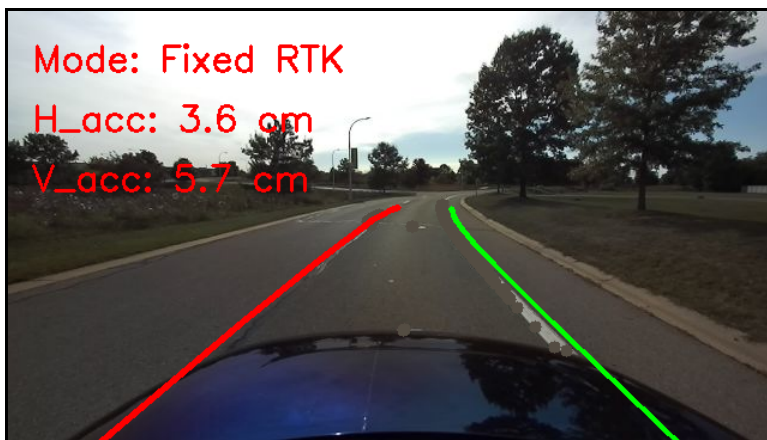
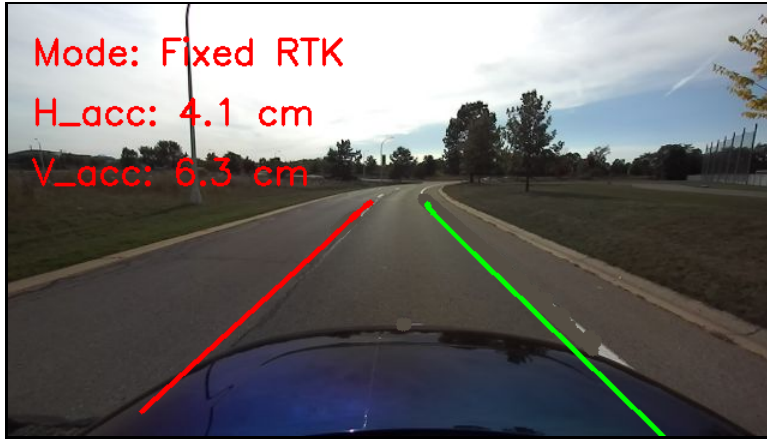


Figure 26. Two examples of Campus Drive camera frames from straight road segments with high-definition map lane line data overlaid on artificially occluded lane lines. These show well-aligned lane lines.

The largest challenge in this dataset was caused by heading sensor sampling rate and the constantly curving road. As the road curves, any delay in the heading data will cause lateral misalignment of the projected lane lines. This is why IMU data was utilized for corrections, as described in Equation 1. This is shown in Figure 27. These results show that in order to most accurately overlay lane line geometry using high-definition map data, in addition to very high sensor accuracy, the ego vehicle IMU and GNSS sensors must sample at a very high rate, otherwise the projected lane lines will become misaligned with the true lane lines in curves. This

problem is less apparent for straight road segments, where the sensed ego vehicle heading does not change much.



Figure 27. Campus Drive camera frame from a curved road segment with high-definition map lane line data overlaid on artificially occluded lane lines. Deviation from true lane lines occurs in this curved road segment due to the insufficient sampling rate of the GNSS heading sensor.

Results Summary

Overall, the results of this proof-of-concept study are encouraging. Lane line geometry was successfully transformed from a proxy high-definition map into a local coordinate frame, then projected onto the camera feed. This allows for the control system of the ego vehicle to utilize high-definition map data in the same way that it would have used camera-detected lane line data, thus solving the problem of lane line occlusion. These results show that this methodology, with sufficient further development, can be used to assist the ego vehicle controller when lane lines become occluded, such as by shadows, leaves, or snow on the road surface.

One challenge with the development of this technology is alignment of the lane lines in curved road segments. Any rotation of the ego vehicle not captured frequently enough will effectively cause a drift of the projected lane lines away from the true lane lines. A very

frequently sampled heading sensor could address this challenge. Further development of quantitative metrics is also suggested.

Conclusions

This paper describes and demonstrates a novel methodology to extract lane line geometry through high-definition maps without the use of real-time camera perception. The lane lines from the high-definition maps were transformed from a global coordinate frame to a local coordinate frame aligned with the camera, and then projected onto the image. This results in the lane lines effectively becoming visible once again and able to be used by the path planning process that would otherwise be inoperable due to lack of input. Overall the results show that this technology concept can be used for augmenting vehicle automation in occluded lane line scenarios. The reconstructed lane lines align very well with the true lane lines in straight road segments, but challenges presently exist with overlay accuracy in curved roads and when GNSS accuracy degrades due to obstruction by trees.

This methodology provides a foundation from which to build an automated navigation procedure robust to lane line occlusion. Perception technology that breaks out of the established norms of camera, radar, and lidar sensing is needed to address the problems of resilient operation and operation in inclement weather. This initial study can be expanded through development of a quantitative measurement in order to rigorously define the accuracy of projected lane lines and enable optimization. Among the suggested metrics was quantification of the mean intersection over union over the drivable region, that is, the area bounded between the two lane-lines. Another metric hypothesized involves using the transformed high-definition map lane line data to construct mathematical equations for the lines on the image, and quantifying the geometric distance of detected lane line pixel locations as the error. This could also be compared with the

current standard of lane line detection. Furthermore, this methodology can be applied to winter driving, where lane lines may be occluded by snow collected on the road surface and line-of-sight to satellites can be disturbed by cloudy weather or precipitation. This methodology could also prove useful for mapping lane line locations by reversing the data processing pipeline, where lane line locations detected through AI or other methods can be transformed into high-precision geospatial points. To improve the issues surrounding heading sampling rates, the data pipeline could be altered to incorporate Kalman filters or other data prediction algorithms.

Chapter III References

1. National Center for Statistics and Analysis, “Traffic Safety Facts 2019: A Compilation of Motor Vehicle Crash Data,” Report No. DOT HS 813 141, US National Highway Traffic Safety Administration, 2021
2. Xu, J., Murphy, S., Arias, E., and Kochanek, K., “Deaths: Final Data for 2019,” Volume 70, Number 8, US Centers for Disease Control and Prevention, 2021, doi:10.15620/cdc:106058
3. National Center for Statistics and Analysis, “Motor Vehicle Traffic Crashes as a Leading Cause of Death in the United States, 2016 and 2017,” Report No. DOT HS 812 927, US National Highway Traffic Safety Administration, 2020
4. US Centers for Disease Control and Prevention, “Web-based Injury Statistics Query and Reporting System (WISQARS) Cost of Injury,” <https://wisqars.cdc.gov/cost>, accessed Oct. 2022.
5. Insurance Institute for Highway Safety, Highway Loss Data Institute, “Real-world benefits of crash avoidance technologies,” 2020

6. Benson, Tefft, Svancara, et al., “Potential reductions in crashes, injuries, and deaths from large-scale deployment of advanced driver assistance systems,” AAA Foundation for Traffic Safety, 2018
7. Harper, C.D., Hendrickson, C.T., and Samaras, C., “Cost and benefit estimates of partially-automated vehicle collision avoidance technologies,” *Accident Analysis & Prevention* 95(Pt A):104–115, 2016, doi:10.1016/j.aap.2016.06.017
8. Gaikwad, T., Rabinowitz, A., Motallebi Araghi, F., Bradley, T., Asher, Z., Fong, A., and Meyer, R., *Vehicle Velocity Prediction Using Artificial Neural Network and Effect of Real World Signals on Prediction Window*, SAE Technical Paper Series, 2020, doi:10.4271/2020-01-0729
9. Rabinowitz, A., Motallebi Araghi, F., Gaikwad, T., Asher, Z.D., and Bradley, T.H., “Development and Evaluation of Velocity Predictive Optimal Energy Management Strategies in Intelligent and Connected Hybrid Electric Vehicles,” *Energies* 14(18):5713, 2021, doi:10.3390/en14185713
10. Motallebi Araghi, F., Yao, K., Rabinowitz, A., Hoehne, C., Garikapati, V., Holden, J., Wood, E., Chen, S., Asher, Z., and Bradley, T., “Mobility energy productivity evaluation of prediction-based vehicle powertrain control combined with optimal traffic management,” SAE Technical Paper Series, SAE International, 400 Commonwealth Drive, Warrendale, PA, United States, 2022, doi:10.4271/2022-01-0141
11. Tunnell, J., Asher, Z.D., Pasricha, S., and Bradley, T.H., “Toward improving vehicle fuel economy with ADAS,” *SAE International Journal of Connected and Automated Vehicles* 1(12-01-02-0005):81–92, 2018

12. Haque, M.R., Islam, M.M., Alam, K.S. et al., “A computer vision based lane detection approach,” *International Journal of Image, Graphics and Signal Processing* 10(3):27, 2019
13. Janai, J., Güney, F., Behl, A. et al., “Computer Vision for Autonomous Vehicles: Problems, Datasets and State of the Art,” *Foundations and Trends in Computer Graphics and Vision* 12(1–3):1–308, 2020, doi:10.1561/06000000079
14. Bounini, F., Gingras, D., Lapointe, V., and Pollart, H., “Autonomous Vehicle and Real Time Road Lanes Detection and Tracking,” *2015 IEEE Vehicle Power and Propulsion Conference (VPPC)*, 1–6, 2015, doi:10.1109/VPPC.2015.7352903
15. US Bureau of Transportation Statistics, “Vehicle Miles Traveled [VMT],” retrieved from FRED, Federal Reserve Bank of St. Louis; <https://fred.stlouisfed.org/series/VMT>, accessed Oct. 2022.
16. Qiu, L. and Nixon, W.A., “Effects of Adverse Weather on Traffic Crashes: Systematic Review and Meta-Analysis,” *Transp. Res. Rec.* 2055(1):139–146, 2008, doi:10.3141/2055-16
17. US Federal Highway Administration, “How Do Weather Events Impact Roads?,” https://ops.fhwa.dot.gov/weather/q1_roadimpact.htm, accessed Oct. 2022.
18. Goberville, N.A., Kadav, P., and Asher, Z.D., “Tire Track Identification: A Method for Drivable Region Detection in Conditions of Snow-Occluded Lane Lines,” *SAE Technical Paper 2022-01-0083*, 2022, doi:10.4271/2022-01-0083.
19. Kadav, Parth, Nicholas Goberville, Farhang Motallebi Araghi, Alvis Fong, and Zachary D. Asher. n.d. “Tire Track Identification: Application of U-Net Deep Learning Model for

- Drivable Region Detection in Snow Occluded Conditions.” In Intelligent Transportation Systems World Congress.
20. US Federal Highway Administration, “Snow and Ice,” https://ops.fhwa.dot.gov/weather/weather_events/snow_ice.htm, accessed Oct. 2022.
 21. Shetty, A., Yu, M., Kurzhanskiy, A., Grembek, O., Tavafoghi, H., and Varaiya, P., “Safety challenges for autonomous vehicles in the absence of connectivity,” *Transp. Res. Part C: Emerg. Technol.* 128:103133, 2021, doi:10.1016/j.trc.2021.103133
 22. Seif, H.G. and Hu, X., “Autonomous Driving in the iCity—HD Maps as a Key Challenge of the Automotive Industry,” *Proc. Est. Acad. Sci. Eng.* 2(2):159–162, 2016, doi:10.1016/J.ENG.2016.02.010
 23. Poggenhans, F., Salscheider, N.O., and Stiller, C., “Precise Localization in High-Definition Road Maps for Urban Regions,” 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2167–2174, 2018, doi:10.1109/IROS.2018.8594414
 24. VSI Labs, “Map-based Lane Keeping with HERE HD Live Map,” Vision Systems Intelligence, LLC, 2018
 25. Trimble DA2 | Catalyst GNSS Systems, <https://geospatial.trimble.com/products-and-solutions/trimble-da2>, accessed Oct. 2022.
 26. Stereolabs, ZED 2i - Industrial AI Stereo Camera, <https://www.stereolabs.com/zed-2i/>, accessed Oct. 2022.
 27. Swift Navigation, Duro Inertial Ruggedized GNSS Receiver Ideal for Outdoor Deployments, <https://www.swiftnav.com/duro-inertial>, accessed Oct. 2022.

28. Stereolabs, Coordinate Frames, <https://www.stereolabs.com/docs/positional-tracking/coordinate-frames/>, accessed Oct. 2022.
29. OpenCV Documentation, Camera Calibration and 3D Reconstruction, https://docs.opencv.org/4.6.0/d9/d0c/group__calib3d.html, accessed Oct. 2022.
30. Rubaiyat, A.H.M., Qin, Y., and Alemzadeh, H., “Experimental Resilience Assessment of an Open-Source Driving Agent,” 2018 IEEE 23rd Pacific Rim International Symposium on Dependable Computing (PRDC), 54–63, 2018, doi:10.1109/PRDC.2018.00016

CHAPTER IV

CONCLUSION

The methodologies presented in this document demonstrate novel expansions of the field of vehicle engineering. The topics of focus were the improvement of fuel economy, and increasing the range of applicable scenarios in which automated vehicle systems can operate.

The first study details a data pipeline that allows heavy-duty vehicle operators to analyze the fuel economy and performance of their fleets through modeling and simulation alone, reducing the cost of prototyping and experimentation. By sequentially applying the outlined procedure of preprocessing, stop capturing, signal fusion, upsampling, and signal fusion to existing fleet telematics data, vehicle models and drive cycles can be generated, validated to within 3% absolute error (as seen by the 1.67% error achieved), and then used to evaluate real-world fuel economy in order to reduce fueling costs and greenhouse gas emissions. This study addressed the stated research question, proving that the industry-typical sparse telematics data can indeed be used for creation and validation of heavy-duty vehicle models and drive cycles.

The second study shows how the limited operational design domain of the developing field of vehicle automation is remedied through recovery of lane line geometry via high-definition maps. A proxy high-definition map was created using several high-precision geospatial sensors. The data was then interpolated and transformed in a data pipeline from the global WGS 1984 coordinate frame to exactly align with the local coordinate frame of the ego vehicle camera. This allows the perception subsystem of automated vehicle systems to operate even without line-of-sight to lane lines, showing that the operational design domain can be expanded while only depending on network connectivity and availability of high-definition map data. This second study successfully addressed the associated research question by describing

and demonstrating a novel methodology that expands the operational design domain of automated vehicle technology to occluded lane-lines.

Together, these studies form a basis for the future improvement of vehicle fuel economy and the expansion of the operational design domain of automated vehicle systems to lane line occlusion scenarios, paving the way for the vehicle of the future.