IDEA
Innovations Deserving
Exploratory Analysis Programs

NCHRP IDEA Program

Turning Movement Counts on Shared Lanes: Prototype Development and Analysis Procedures

Final Report for
NCHRP IDEA Project 198

Prepared by:
David A. Noyce
Andrea R. Bill
Madhav V. Chitturi
Kelvin R. Santiago-Chaparro
University of Wisconsin-Madison

August 2019

TRB TRANSPORTATION RESEARCH BOARD
The National Academies of
SCIENCES · ENGINEERING · MEDICINE

Innovations Deserving Exploratory Analysis (IDEA) Programs Managed by the Transportation Research Board

This IDEA project was funded by the NCHRP IDEA Program.

The TRB currently manages the following three IDEA programs:

- The NCHRP IDEA Program, which focuses on advances in the design, construction, and maintenance of highway systems, is funded by American Association of State Highway and Transportation Officials (AASHTO) as part of the National Cooperative Highway Research Program (NCHRP).
- The Safety IDEA Program currently focuses on innovative approaches for improving railroad safety or performance. The program is currently funded by the Federal Railroad Administration (FRA). The program was previously jointly funded by the Federal Motor Carrier Safety Administration (FMCSA) and the FRA.
- The Transit IDEA Program, which supports development and testing of innovative concepts and methods for advancing transit practice, is funded by the Federal Transit Administration (FTA) as part of the Transit Cooperative Research Program (TCRP).

Management of the three IDEA programs is coordinated to promote the development and testing of innovative concepts, methods, and technologies.

For information on the IDEA programs, check the IDEA website (www.trb.org/idea). For questions, contact the IDEA programs office by telephone at (202) 334-3310.

IDEA Programs
Transportation Research Board
500 Fifth Street, NW
Washington, DC 20001

The project that is the subject of this contractor-authored report was a part of the Innovations Deserving Exploratory Analysis (IDEA) Programs, which are managed by the Transportation Research Board (TRB) with the approval of the National Academies of Sciences, Engineering, and Medicine. The members of the oversight committee that monitored the project and reviewed the report were chosen for their special competencies and with regard for appropriate balance. The views expressed in this report are those of the contractor who conducted the investigation documented in this report and do not necessarily reflect those of the Transportation Research Board; the National Academies of Sciences, Engineering, and Medicine; or the sponsors of the IDEA Programs.

The Transportation Research Board; the National Academies of Sciences, Engineering, and Medicine; and the organizations that sponsor the IDEA Programs do not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the object of the investigation.

Turning Movement Counts on Shared Lanes: Prototype Development and Analysis Procedures

IDEA Program Final Report

Project 198

Prepared for the IDEA Program
Transportation Research Board
The National Academies

David A. Noyce
Andrea R. Bill
Madhav V. Chitturi
Kelvin R. Santiago-Chaparro

University of Wisconsin-Madison
Department of Civil & Environmental Engineering

Submitted on June 4, 2019
Submitted with Revisions on August 14, 2019

ACKNOWLEDGEMENTS

The project is funded by the NCHRP IDEA Program. Special thanks to Rebecca Szymkowski from the Wisconsin DOT, Michael Hardy from the City of Appleton, Yang Tao from the City of Madison, and Leroy Wignot from MS Sedco which were part of our advisory panel and provided their expertise in setting the direction of the project. Special thanks to additional staff from MS Sedco (commercialization partner), including Jonathon Brack, who supported the project by providing technical guidance and answering technical questions at key stages of the project.

EXPERT REVIEW PANEL

Rebecca Szymkowski, *Wisconsin DOT*
Michael Hardy, *City of Appleton*
Yang Tao, *City of Madison*
Leroy Wignot, *MS Sedco, Inc.*
Jonathon Brack, *MS Sedco, Inc.*

TABLE OF CONTENTS

LIST OF FIGURES

EXECUTIVE SUMMARY

Turning movement count data is key to evaluating the performance of signalized intersections and is also a crucial component of data-driven decision-making processes of transportation agencies. Unfortunately, the availability of quality turning movement count data is not the norm for transportation agencies. In fact, the 2012 National Traffic Signal Report Card conducted by the National Transportation Operations Coalition identified traffic monitoring and data collection practices in the U.S. as weak and thus assigning the practices an “F” grade, a grade that did not change after a similar assessment was conducted in 2007.

Even as technology advances, some of the current methods used for turning movement counts that are considered standard practice are based on manual procedures. These procedures, while accurate if performed correctly, limit the amount of data available to transportation agencies due to the reliance on human observers. Therefore, it is not a surprise that automated methods have been developed as alternatives to manual procedures. Temporary automated counters are often installed by an intersection to obtain volume data for periods longer than what is possible with human observers but these are still deployed for a limited period of time, typically less than a week, and as such are unable to capture cyclical changes in traffic.

The use of radar-based vehicle detection systems as an alternative to loop detectors has grown over the years and is arguably an underutilized vehicle detection technology. The underutilization argument is based on radar-based vehicle detection systems being capable of continuously tracking the position of vehicles but only reporting to the controller the presence of vehicles over a detection zone that emulates the location of an inductive loop. If vehicle trajectory data from an intersection approach is continuously logged, monitoring vehicle volumes over long periods of times is possible, including breaking down the volume into movements by analyzing the paths of the vehicle trajectories regardless of lane configurations.

Project Results

Results from the project include the development of a data collection device capable of logging vehicle trajectories from intersections instrumented with a commercially available radar-based vehicle detection system. The device can be installed inside a signal cabinet, is independent of the controller platform, and was the result of work by the commercialization partner (MS Sedco) in coordination with the research team. One of the advantages of the data collection device is that it serves as a platform for algorithms that make performance measures monitoring possible. An example of these algorithms is the one developed as part of the Type 1 IDEA project which was streamlined and improved as part of the Type 2 project described in this report. The data collection device implements some of the key noise removal techniques described in this report thus making it possible to improve the quality of turning movement counts generated based on the trajectory data collected. Improvements made to the noise removal and summary procedures made commercializing the product possible and open the doors for future improvements and the introduction of analysis procedures beyond turning movement counts.

Previous Accuracy Findings and Improvement Opportunities

A classification algorithm developed as part of the Type 1 IDEA project can generate turning movement counts that were over 90% accurate based on the type of data obtained using the data collection device described. Accuracy was measured by comparing ground-truth vehicle volume with volume reported by a classification algorithm. This accuracy was deemed sufficient since it is similar to that claimed by existing products with less functionality and capabilities. However, a review of the results and a detailed side-by-side comparison of trajectory data and video from signalized intersection revealed that the best approach to improve the performance of the previously developed classification algorithm was to improve the accuracy of the underlying data.

To improve the accuracy of the underlying data, noise removal procedures were developed that achieve similar levels of volume accuracy by focusing on accurately representing trajectories and eliminating noise. The noise removal procedure keeps track of vehicles that enter the intersection and implementation of key aspects of the procedure was possible by relying on the data collection device commercialized by the commercialization partner. Development of the noise removal procedure required a characterization of the typical noise found in the dataset to gain a detailed understanding of the noise found in a typical dataset of vehicle trajectories. Finally, a procedure for assigning lane values to vehicle trajectories was also identified and is documented in this report.

Commercialization Status and Next Steps

An initial version of a data collection device capable of implementing algorithms to obtain performance measures from signalized intersections has been commercialized by the commercialization partner (MS Sedco). The commercialization partner released an initial version of the data collection and analysis product in the Summer of 2019. The data collection device implements the key noise removal techniques described in this report and can be a platform for future performance monitoring techniques. Furthermore, it implements procedures to breakdown vehicle volume at an intersection approach by lane. As a result, adding classification by movement based on a streamlined version of procedures developed as part of the Type 1 IDEA project can be accomplished via a software update.

Once the necessary commercialization steps with the licensing arm of the University of Wisconsin-Madison are cleared, the commercialization partner plans to integrate a streamlined version of the classification algorithm into their product via a software update. Adding the classification algorithm to the commercialized data collection product as a software update will be possible because of the joint effort between the commercialization partner and the research team to define a strong and flexible underlying software architecture and data storage techniques.

CHAPTER 1. INTRODUCTION

Turning movement count data, i.e., vehicle volume classified by time periods into specific movements, approaches, is key to evaluating the performance of signalized intersections both from the safety and operational perspective. Turning movement counts are also a crucial component of data-driven decision-making processes used by transportation agencies. Unfortunately, having quality turning movement count data is not the norm for transportation agencies. In fact, the 2012 National Traffic Signal Report Card conducted by the National Transportation Operations Coalition identified traffic monitoring and data collection practices in the U.S. as weak and thus assigning the practices an "F" grade [1], a grade that did not change after a similar assessment was conducted in 2007 [2].

Even as technology advances, some of the currents methods that are used to collect turning movement count data and that are considered standard practice are still based on manual procedures. These manual procedures, while accurate if performed correctly, limit the amount of data available to agencies due to the reliance on human observers. Automated methods that are often based on image processing techniques are often installed by an intersection to obtain volume data for periods longer than what is possible with human observers but these are still deployed for a limited period of time, typically less than a week, and as such are a temporary solution unable to capture cyclical changes in traffic.

1.1 IDEA PRODUCT

Radar-based vehicle detection technologies have grown in use over time as an alternative to traditional loop-based vehicle detection at signalized intersections. In collaboration with a commercialization partner (MS Sedco), the research team developed data collection and analysis procedures capable of analyzing the trajectories of vehicles on an intersection approach and classifying those trajectories into vehicle movements irrespective of the lane used by the vehicle. In other words, the classification of vehicle trajectories into movements was demonstrated as feasible even for vehicles using a shared lane. The feasibility was demonstrated as part of a Type 1 project [3]. As part of a Type 2 project, the quality of the data available for analysis was improved with new noise removal techniques; thus, simplifying the analysis procedures needed to classify vehicles into movements. The research team also worked closely with the commercialization partner on the development of a data collection device commercialized in the Summer of 2019. Finally, a framework that will make it possible to obtain additional performance measures beyond what was thought initially possible was also established.

1.1.1 Concept and Innovation

A limitation of existing vehicle detection systems is that, in general, even when the underlying technology is significantly more advanced than loop detectors, vehicle detection systems continue to emulate the behavior of inductive loop detectors by communicating the presence of vehicles on an intersection approach using zone-based position. While there are reasons for communicating the presence of vehicles at specific zones within an intersection approach, such as algorithms that rely on this type of data to generate performance measures, such a narrow view of performance monitoring leads to ignoring valuable information about the position of vehicles on an intersection approach. For example, radar-based vehicle detection can continuously track the position of vehicles on an intersection approach, but the underlying trajectory data is only used for communicating the presence of a vehicle to the signal controller once a vehicle is within a detection zone. The remaining underlying data is ignored.

Data collection procedures capable of monitoring the underlying trajectory data of a commercially available radar-based vehicle detection system were developed. The procedures can log vehicle trajectories without interfering with the required interactions between the radar sensor and the signal controller that makes emulating loop detectors possible. By storing vehicle trajectory data, advanced performance measures can be obtained. Examples of performance measures that can be obtained include turning movement counts even at intersections with shared lanes, something that can’t be obtained by analyzing loop-based data unless supplemental loops are installed.

In the research project described in this report, noise removal and filtering techniques were also developed to improve the quality of the existing data collection and analysis procedures. This innovation will make it possible to obtain turning movement counts and will also enable the monitoring of other performance measures. All of this will be possible by deploying a small data collection device installed inside the signal cabinet that is independent of the signal controller, thus facilitating the adoption of the technology. Connectivity and configuration of the device are enabled through a web server thus providing for easy data analysis and configuration via a web browser. Interacting with the device is possible over the network or via a wireless network activated by the device near the signal cabinet.

1.2 PREVIOUS WORK AND OPPORTUNITIES FOR IMPROVEMENT

A classification algorithm was previously implemented using the R programming language and by relying on other software tools that typically run on desktop computers. The algorithm was found to provide an accurate representation of the vehicle volume at signalized intersections. Through discussion with our commercialization partner, based on additional feedback received, and based on the experience of the research team, numerous changes to the underlying procedures and supporting platform were identified as recommended or needed to position the research as a commercially viable product successfully. The sections ahead summarize some of these recommended changes. Final implementation of these changes will depend on final decisions made by the commercialization partner but as will be described throughout this report. A number of these changes have already made it to a data collection system commercialized in the Summer of 2019 by the commercialization partner.

1.2.1 Changes to Supporting Platform

One of the limitations of the previous research is the reliance on software tools, and an operating system, typically associated with desktop platforms. Changes should be made to move the data collection and analysis procedures to a Linux-based operating system. A move towards a Linux-based operating system will make it possible to create a data collection and analysis platform that can run on devices such as the Raspberry Pi as well as on custom hardware, thus ensuring the technical portability of the platform in the future. For example, the code base that supports the data collection and analysis processes should be portable to other processor architectures thus making it possible to be deployed as part of other hardware products.

1.2.2 Changes to Data Collection Procedure

The previous data collection approach was focused on obtaining vehicle trajectory data with limited vehicle summary information. This approach forced the analysis procedures to focus on removing noise in the dataset and filtering unnecessary vehicle trajectories. An updated version of the data collection procedure should summarize vehicle trajectory data and provide additional information that can be used by a simplified version of the movement classification algorithm as well as other performance monitoring procedures. Better noise removal techniques should be developed as well as lane classification techniques to support better decision making by classification algorithms. Summary trajectory data should be exposed over the network to allow a simplified version of the analysis procedures to query the data and generate performance measures.

1.2.3 Implementation of Web-Based Client-Side Analysis Procedures

The previously developed analysis procedures rely on software tools that are not well-suited for commercialization. For example, the classification algorithm was implemented in the R programming language and relied on supplemental packages to analyze vehicle trajectory data. And while the R programming language provides an excellent tool for research and prototyping, using the language on a commercial product introduces unnecessary complexities into the commercialization process. Therefore, it is recommended that a client-based approach to data analysis be implemented. In the recommended client-based approach, the purpose of the data collection system installed inside a signal cabinet will be to collect and summarize vehicle trajectory data but not conducting any analysis, i.e., implement the changes mentioned in Section 1.2.2. A web page should be served by the data collection system over the network containing analysis code written in JavaScript which is executed on the web browser of the user accessing the web page. The code served should take advantage of existing visualization libraries that make the creation of dynamic charts and analysis of data possible.

1.3 OBJECTIVES

Several of the recommended changes, to prepare the research for commercialization, were implemented directly by the commercialization partner (MS Sedco) or will be implemented in the future. The focus of the research team was on identifying changes that improve the quality of the final product from the data collection and analysis perspective. Therefore, the following were the research objectives of the project:

- Understand the nature of the noise that is found in vehicle trajectory datasets to develop better noise removal procedures.
- Based on an understanding of the noise in trajectory datasets, develop filtering procedures that improve the quality of the data to make the results more reliable.
- Develop lane classification procedures that can be used to support a more constrained vehicle movement prediction procedure to prevent misclassification due to confusions in the trajectory dataset downstream of the stop bar.

1.4 COMMERCIALIZATION STATUS AND NEXT STEPS

An initial version of a data collection device capable of implementing algorithms to obtain performance measures from signalized intersections has been commercialized by the commercialization partner. The commercialization partner released an initial version of the device in the Summer of 2019 that implements key noise removal techniques described in this report, and that serves as a platform for implementing additional performance monitoring techniques in the future. Furthermore, it implements procedures to breakdown vehicle volume at an intersection approach by lane. Lessons learned from test deployments on 3 cities (Appleton, WI, Bloomington, IL, and, Ames, IA) and multiple intersections were key to developing the commercialized device. Examples of locations were the device was tested include the intersection of East Northland Avenue & North Meade Street (City of Appleton) and the intersection of University Boulevard & Highway 30 Westbound Offramp (City of Ames).

As a result of the underlying device software architecture, adding classification by movement based on a streamlined version of procedures developed as part of the Type 1 IDEA project can be achieved via a software update. Once the necessary commercialization steps with the licensing arm of the University of Wisconsin-Madison are cleared, the commercialization partner plans to integrate a streamlined version of the classification algorithm into their product via a software update. This improved version of the classification algorithm that will be deployed as a software update will rely on summarized trajectory data generated after the application of the key filtering procedures described in this report and will include an additional layer of classification by relying on a lane value assigned to each vehicle trajectory. Appendix C provides details technical details about the improved version of the classification algorithm and observed performance.

CHAPTER 2. NOISE REMOVAL AND CHARACTERIZATION PROCEDURES

From the transportation engineering perspective, existing scientific research has focused on comparing the performance of radar-based detection system with other types of detection systems that rely on alternative technologies such as video and thermal imaging. That is in addition to comparing the performance against traditional detection systems based on inductive loops. The comparisons have primarily focused on monitoring the activation of detection zones across different systems due to the presence of vehicles. When the data is compared, statistics about missed calls, false calls, stuck-on calls, dropped calls are usually produced. In the literature, the percentage of missed calls represent the portion of confirmed vehicles that traveled through a zone and that were not detected by the system. Similarly, the percentage of false calls represent the portion of calls reported by a detection system that was not the result of a vehicle traveling through a zone. The percentage of stuck-on calls represent the portion of calls that remain active after vehicles leave the detection zone. Finally, the percentage of dropped calls represent the portion of calls associated with the presence of a vehicle that was canceled while the vehicle was still within a zone.

The evaluation of device performance by comparing missed, false, stuck-on, and dropped calls across systems is a valid one for understanding how vehicle detection devices function as a replacement for loop-based detection. These types of evaluations have shown that radar-based vehicle detection is a reliable replacement for traditional loop detection. As a result, radar-based technology was used as the foundation for generating automated turning movement counts at signalized intersections using a classification algorithm developed as part of the previously completed NCHRP IDEA Type 1 project.

As previously mentioned, one of the objectives of the Type 2 project described in this report was making the results from the classification algorithm more reliable. One of the areas identified for improvements was the nature of the dataset used by the classification algorithm. A key step required to improve the dataset used by the classification algorithm is to understand the noise in the dataset and identify better filtering procedures. The sections ahead describe the type of noise and recommended procedures for the removal of what will be described as ghost vehicle trajectories. It should be noted that the level of noise removal and characterization outlined in the sections ahead is not recommended for all potential users of the data but instead describe the most detailed approach identified by the research team. In the Plans for Implementation chapter, the recommended noise removal approach for implementation is outlined and balances what is possible from the research perspective with what is possible (by considering user needs) for a final product.

2.1 THE NEED FOR BETTER NOISE CHARACTERIZATION

Understanding the typical behaviors observed in a dataset of radar-based vehicle trajectories can help eliminate ghost trajectories and clean incorrect portions of trajectories. The term ghost trajectory is used to describe duplicate trajectories usually caused by large and segmented vehicles. Incorrect portions of trajectories are often the result of the radar system continuing to assume that a vehicle is stopped upon arriving at the intersection while recognizing the eventual departure from the intersection as a new vehicle. If ghost and incorrect trajectories are removed from the dataset, the computation of better traffic counts will be possible. Therefore, the effort to establish filtering techniques to clean-up vehicle trajectories datasets and to characterize the trajectories in the dataset is highlighted. Filtering involves the removal (or trimming) of trajectories, while characterization involves the identification of typical behaviors that can be used to group trajectories into groups useful for detailed evaluations. No research effort has been found that provides a detailed discussion of vehicle trajectory data at the individual vehicle level described in the sections ahead. Therefore, the procedures presented are key not only to the completion of the Type 2 project described but also to advancing the field of performance measures monitoring.

2.2 TRAJECTORY SCENARIOS REQUIRING FURTHER FILTERING

A review of different video and trajectory dataset shows that most trajectories reported by the radar accurately represent a vehicle. This is consistent with previous research, which found that radar-based vehicle detection systems have a lower percentage of false calls when compared with other detection technologies. However, the reviews also revealed that there are instances in which the radar correctly tracks a vehicle but generates duplicate trajectories and instances in which trajectories are shown in a false collision path. These instances are limited; however, the presence alone demonstrates there are potential dataset cleanup procedures that could rely on the characterization of typical errors. One common trend found for errors in trajectory reporting was that errors were associated with scenarios that violate the laws of physics, i.e., trajectory errors could be easily identified because two different vehicles are reported as occupying the same space at the same time. Five types of errors found in trajectory datasets are discussed ahead as Type A, B, C, D, or D trajectories.

2.2.1 Type A: Converging Trajectories

Figure 1 shows an example of a trajectory error caused by the radar incorrectly converging the path of vehicles into a single point. Figure 1a shows vehicles detected by the radar, while Figure 1b shows the radar representation of the dataset. Radar data for the timestamp shown in Figure 1a was accurate. However, when the vehicles started moving vehicle 1 and 4 were reported by the radar as been on the same spot at the same time (Figure 1c). Such a pattern would be characteristic of a crash; however, the data shows vehicle 1 disappearing while vehicle 4 continued moving without interruptions.

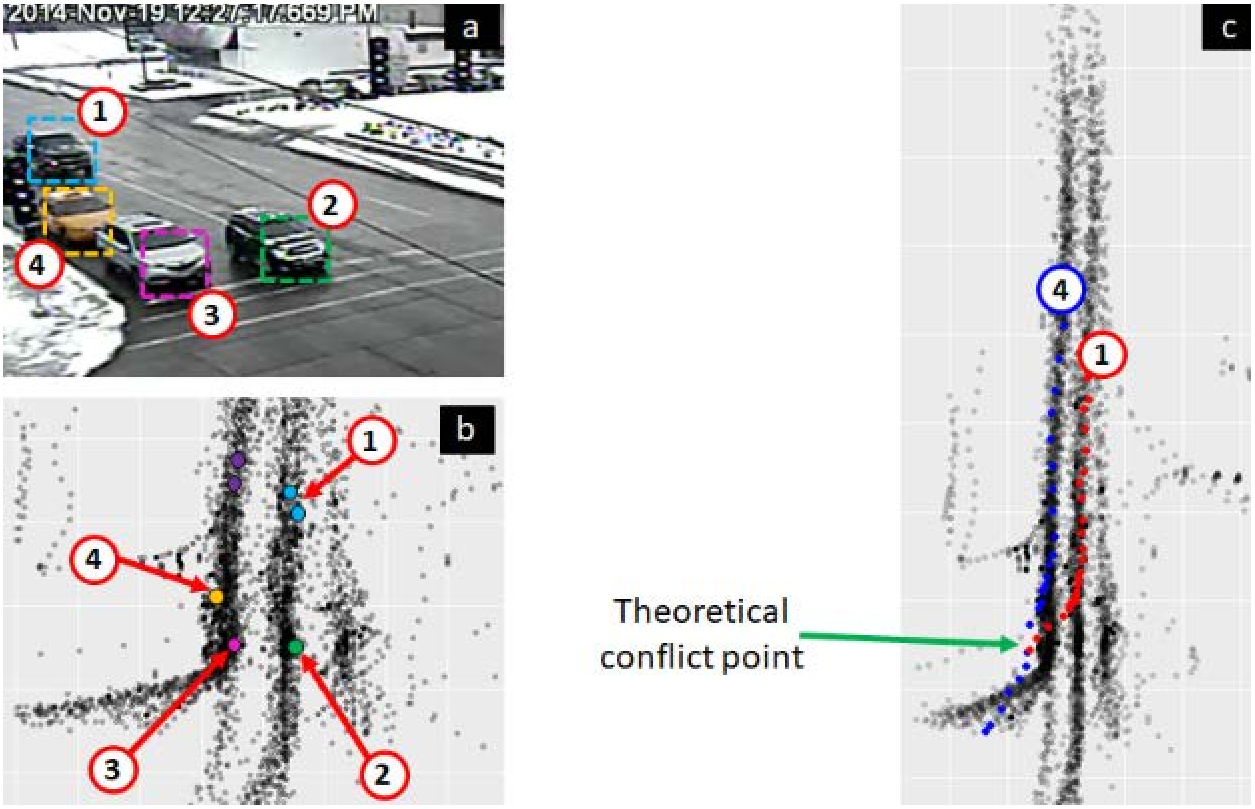


Figure 1. Type A - Convergence of Trajectories

2.2.2 Type B: Ghost Trajectories

Ghost trajectories are a type of trajectory error in which two different vehicle trajectories are reported for the same vehicle by the radar. These are often temporary, but there are instances in which the duration (start and end based on time) is similar to that of a full trajectory. Figure 2 and Figure 3 show examples of ghost trajectories. In both figures, vehicle 1 in the video screenshot is represented by two different sets of trajectory points shown in blue/red.

2.2.3 Type C: Segmented Trajectories

A segmented trajectory is a vehicle trajectory that has been split into two by the radar. A segmented trajectory is the result of a vehicle been dropped by the radar and then detected again as a new vehicle. When detected again, a shift in the position reported can change, thus creating a theoretical conflict, i.e., different vehicles on the same location at the same time. Figure 4a shows two vehicles detected by the radar while Figure 4b shows that for the most upstream vehicle, two different unique trajectories are associated with the vehicle. These two unique trajectories overlap in space-time temporarily and when the overlap is ignored, and the two trajectories are combined, the path followed by the vehicle is clearly defined as shown in Figure 4c.

Not all segmented trajectory types can be detected by identifying trajectory points that appear to occupy the same space at the same time. A trajectory segmented into two different ones often have endpoints and start points that are sufficiently spaced in terms of time and distance to not trigger and space-time conflict. An example of the aforementioned scenario is shown in Figure 5. As the figure shows, two sets of trajectory points (blue/red) are used to represent the position of the vehicle. The Type C trajectory scenario is often the result of a vehicle that comes to a stop and is then detected as a new vehicle when it starts moving again.

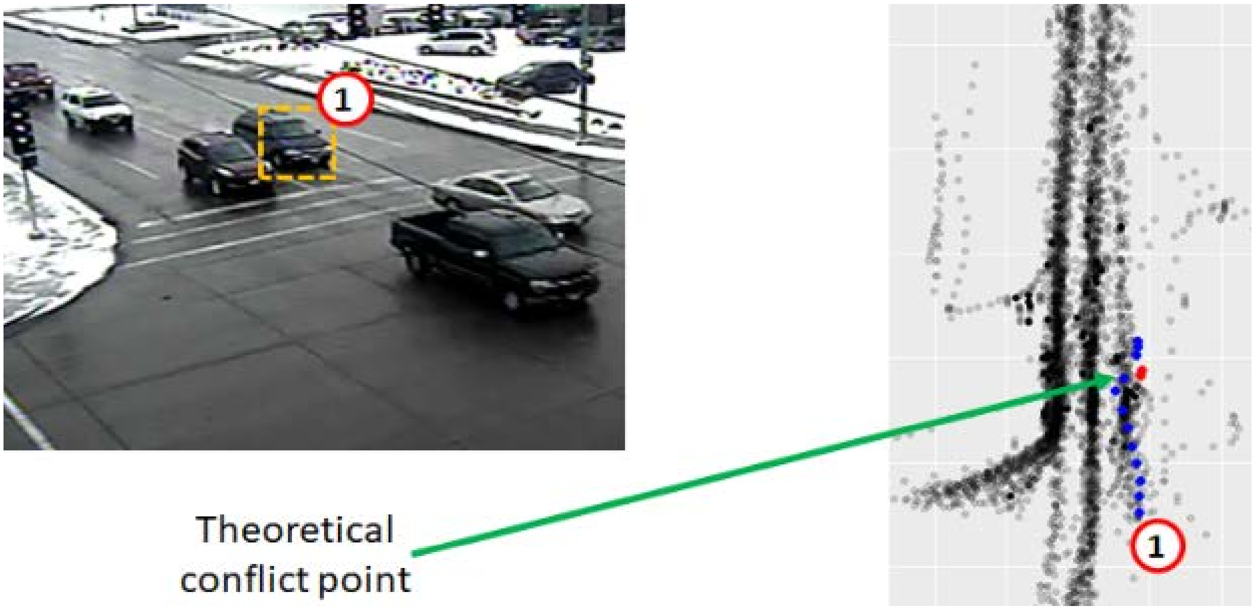


Figure 2. Type B - Ghost Trajectory (Example 01)

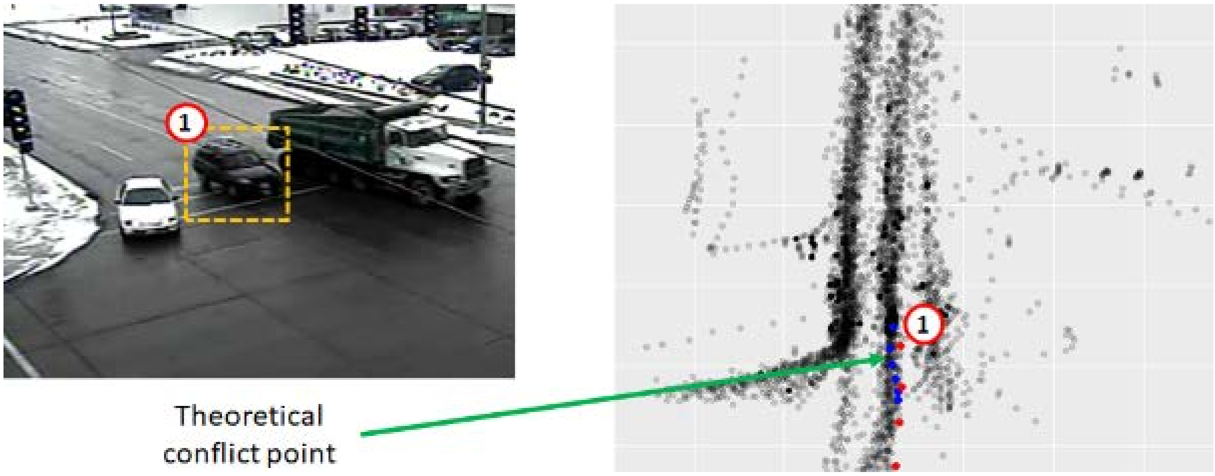


Figure 3. Type B - Ghost Trajectory (Example 02)

2.2.4 Type D: Static Trajectories

A static vehicle trajectory error is a special type of error caused by the false detection of a vehicle by the radar-based vehicle detection system. One characteristic of this type of error is that trajectory points logged have a vehicle speed that is always zero, and as a result, there is no change in the position over time. Points associated with a Type D trajectory are often found along areas of the approach where vehicles stop, e.g., the stop bar or by driveways. From an analysis perspective, these errors are often the result of ghost trajectories (Type B) that are too far from the main trajectory to be detected by identifying space-time conflicts. These static trajectories could also result from pedestrian activity near the intersection or vehicles within a parking lot.

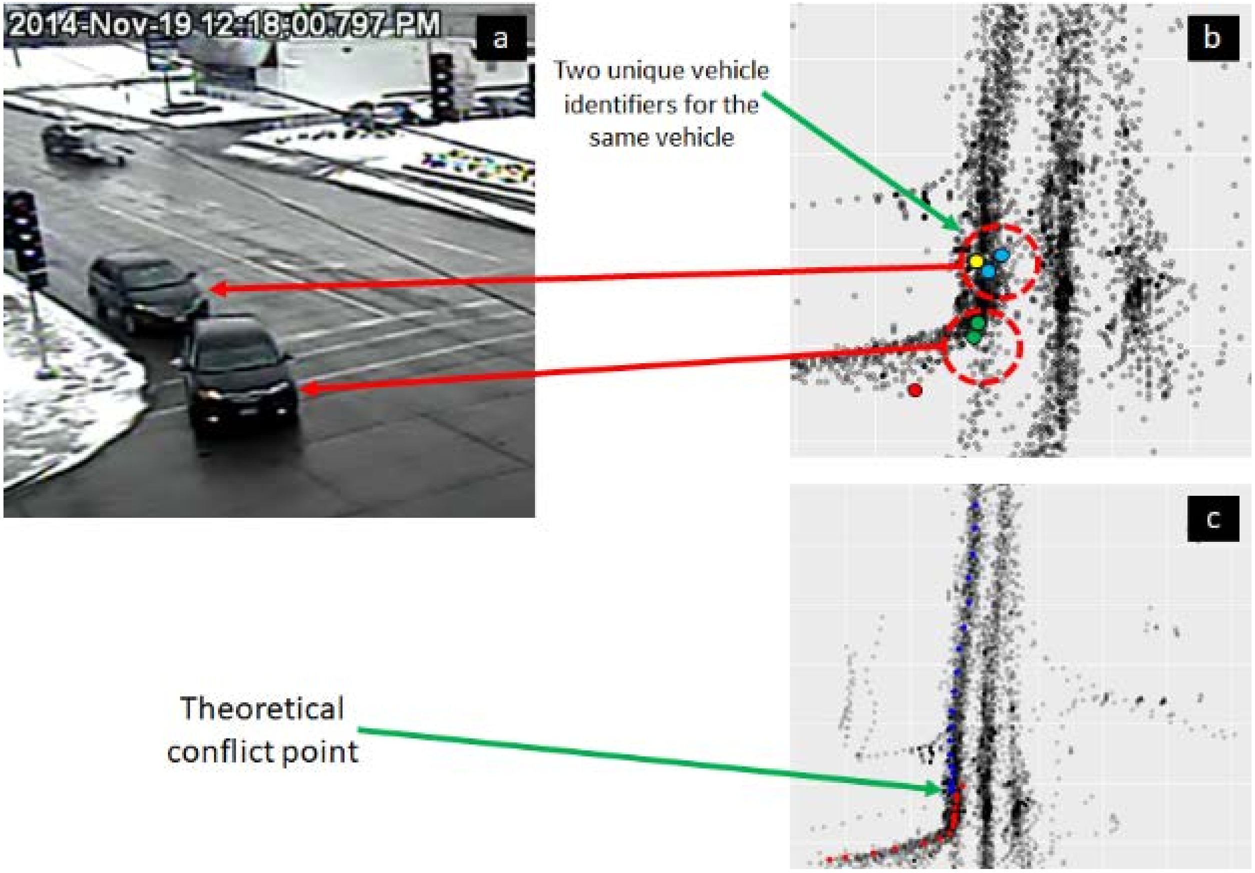


Figure 4. Type C - Segmented Trajectory (Space-Time Conflict)

2.2.5 Type E: Conflicting Approach Trajectories

Conflicting approach trajectories refer to noise caused by a radar device temporarily reporting the presence of vehicles traveling on a conflicting approach. For example, a radar pointed towards the southbound approach often tracks vehicles exiting the westbound approach while traveling in front of the southbound approach stop bar. Points associated with a Type E trajectory are easily identifiable since the first and last

Y coordinate are downstream of the stop bar and the total distance traveled on the Y-axis is smaller than the distance traveled on the X-axis.

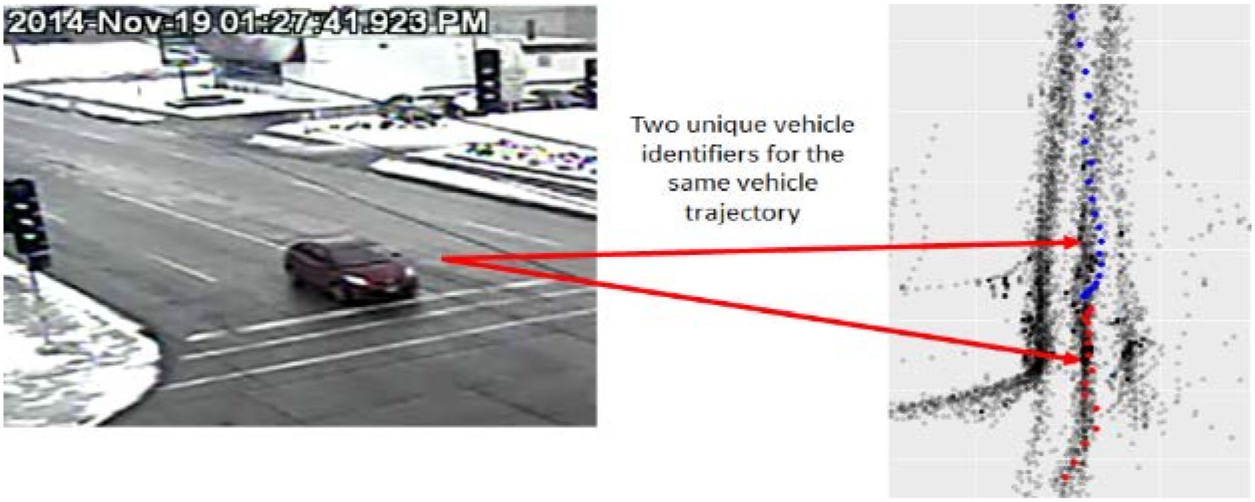


Figure 5. Type C - Segmented Trajectory (No Space-Time Conflict)

2.2.6 Filtering Trajectory Data Based on Scenario Type

The typical dataset obtained for an intersection approach is visualized in Figure 6a. Therefore, a typical dataset such as the one shown in the figure requires the application of filters to remove trajectory points associated with the Type D and E trajectories discussed in Sections 2.2.4 and 2.2.5. Similarly, filtering techniques can be used to detect scenarios with Type A, B, and C pairs of trajectories. Once detected, trajectories part of the pairs can be removed, joined, or trimmed depending on the type of scenario. Steps outlined ahead describe the procedures that can be used to clean the trajectories dataset for an approach and prepare the dataset for further analysis.

2.2.7 Step No. 1: Removal of Type E Trajectories

Each vehicle trajectory is associated with a unique vehicle identifier. Therefore, for each identifier, the associated trajectory points can be sorted by timestamp, thus allowing the identification of the first and last Y coordinates reported for the vehicle. Type E trajectories can then be removed from the initial dataset if the first Y coordinate is lower than Y_T, a user-provided threshold selected to reflect a point along the centerline of the approach between the Y_{B1} and Y_{B2} asymptotes in Figure 7. In the case of Figure 6, the Y_T parameter is set to 95 feet, and it represents a position immediately upstream of the Y_{B1} asymptote shown in Figure 7. Y_T can vary and does not necessarily have to match the Y_{B1} asymptote and is key for the filtering and analysis process. Figure 6b shows an example of a dataset for which E trajectory points were removed using Y_T.

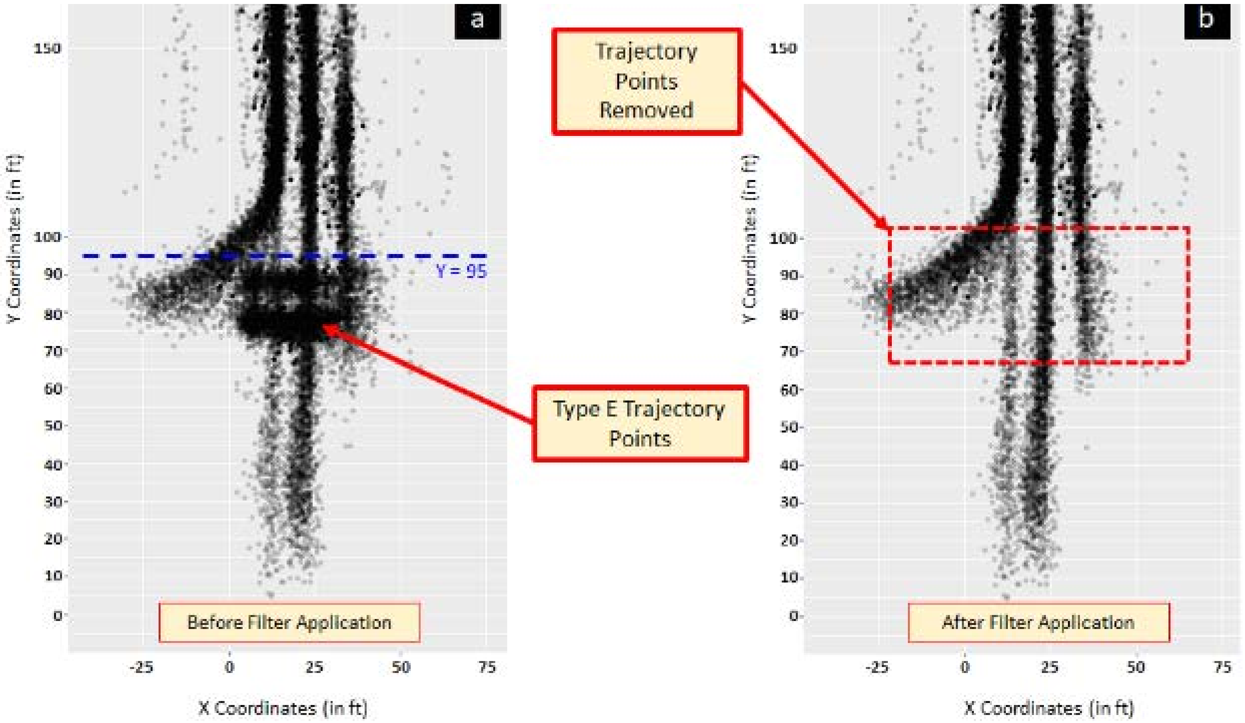


Figure 6. Illustrated Removal of Type E Trajectories

Typically, the number of Type E trajectories in a dataset is of significant magnitude when compared to the total number of trajectories obtained for an intersection approach. For example, in the "Wisconsin Avenue and North Meade Street" southbound approach, Type E trajectories represent approximately 48% of the trajectories reported by the radar. While the percentage mentioned is significantly dependent on traffic and varies by approach, it illustrates the importance of removing Type E trajectories.

2.2.8 Step No. 2: Removal of Type D Trajectories

As in the case of removing Type E trajectories, the first and last Y as well as the first and last X coordinate associated with each vehicle trajectory can be identified. Therefore, if no change in the vehicle position is detected during the analysis process, then that classifies the points as Type D trajectory. As such, the removal of the points from the dataset is warranted. Figure 8 shows the location of Type D trajectory points along the approach. On the southbound approach of the "Wisconsin Avenue and North Meade Intersection," Type D trajectories represent approximately 4% of all trajectories reported by the radar. The reported percentage, while certainly not as significant as the percentage of Type E trajectories, does highlight the need for removal of the trajectories prior to analyzing the content of a dataset of vehicle trajectories.

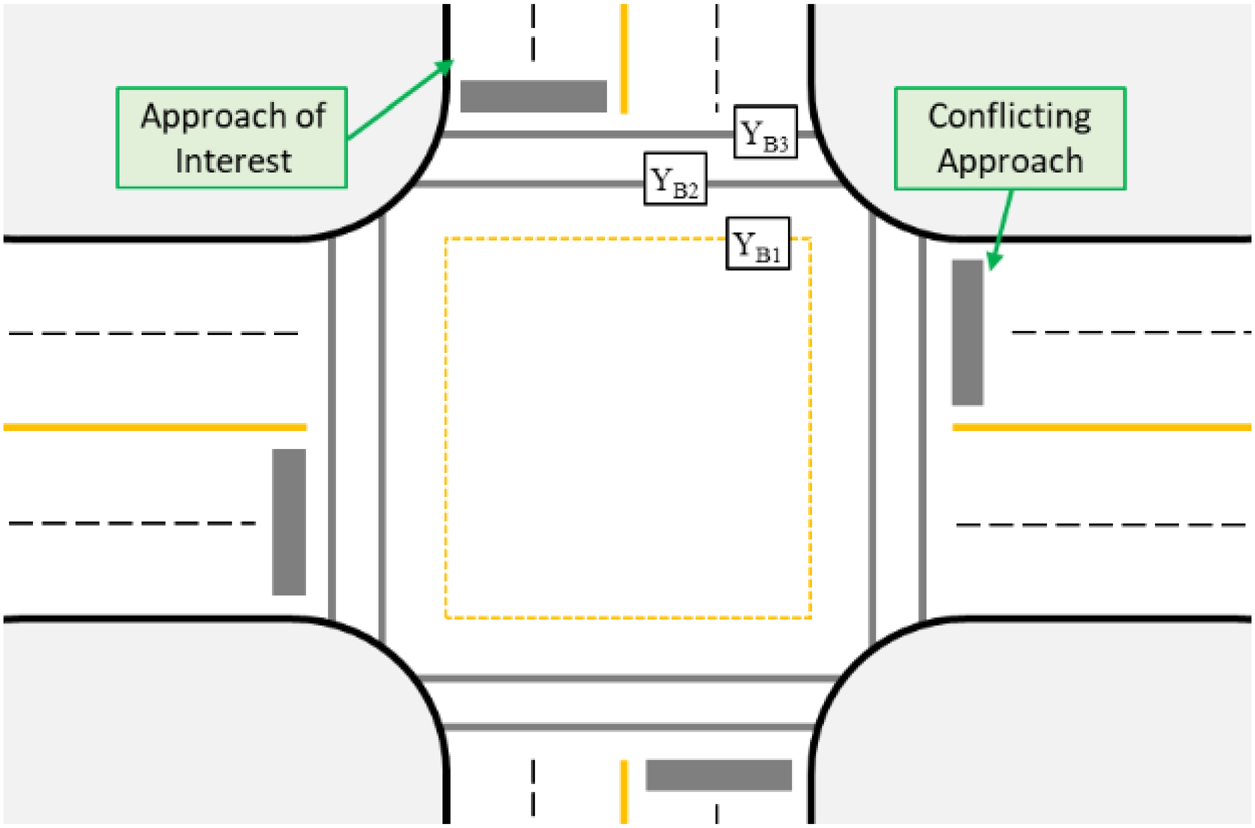


Figure 7. Critical Points and Typical Intersection Schematics

2.2.9 Step No. 3: Identifying Potentially Conflicting Trajectories

Once Type D and E trajectories are removed, pair of trajectories that appear to violate the laws of physics (having a space-time conflict) need to be identified and classified into Type A, B, or C pairs. The identification of potentially conflicting trajectories requires information about each vehicle trajectory that goes beyond the known start points and endpoints of the trajectory. A filter, based on the concept of space-time conflicts, is used to identify conflicting pairs of trajectories. The filter detects two trajectories that appear to be trying to occupy the same space at the same time. A key input value required for the aforementioned filter is the threshold distance (D_T) used to determine if two trajectories are reported as occupying the same space at the same time. A D_T value is required since the position reported by the radar device is that of the front vehicle bumper.

D_T is a user-defined value based on knowledge about the site and traffic conditions. For example, in the case of a standard sedan, if two trajectories are reported as being 6 feet apart, that would indicate a potential space-time conflict thus suggesting that in such a scenario a D_T value equal to 6 feet is appropriate. The procedure is based on a moving time window that evaluates trajectory points at every unique timestamp record available in the trajectory dataset.

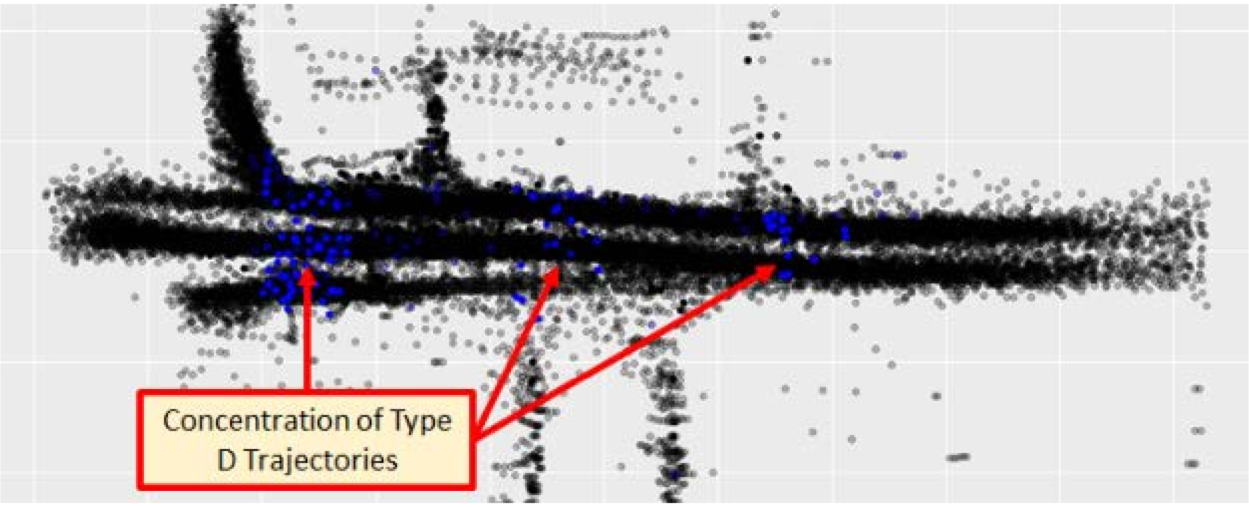


Figure 8. Illustration of Type D Trajectories to Remove

The result of the conflict detection process described is a trajectory pairs dataset (C_P) that appear to occupy the same point at the same time. For each pair of trajectories, a second filter determines if the scenario is one that reported a conflict due to incorrect tracking of a stopped vehicle by the radar. Errors due to incorrect tracking of a stopped vehicle is the most common cause of erroneous conflict detection. Under such a scenario, an erroneous conflict is detected because instead of tracking a vehicle that stopped and then started moving again via the same unique identifier, the radar assigns a new unique identifier to the same vehicle once it starts moving. At the same time, a "ghost" vehicle associated with the initial unique identifier is still reported for the last known stopped position. Therefore, when a properly tracked vehicle is detected as traveling over the "ghost" vehicle by the filtering procedure then an erroneous conflict is detected.

Each pair of trajectories identified as conflicting after the application of the filtering procedures is analyzed to detect erroneous conflicts. The longest trajectory, measured in terms of distance traveled along the Y-axis, is identified and referred to as P_L. The shortest trajectory is identified as P_S. Stopped observations are then removed from each of the trajectories, thus resulting in a collection of points known as P_{LM} and P_{SM}. For each trajectory point in P_{LM}, the corresponding point (based on the timestamp) in P_{SM} is found and the distance between the points is computed. If all the distances computed for a pair of trajectories are greater than the specified D_T value, the reported conflict is deemed erroneous. Figure 9 shows examples of erroneous conflicts identified due to the application of the space-time violation filter. In the figure, trajectory points associated with stopped vehicles are shown in green. As shown, once green points are removed, the trajectories no longer overlap.

For analysis purposes, once a pair of trajectory points is identified as representing an erroneous conflict, trajectories in the pair are removed from the C_P dataset and no further filtering is applied. Once these pairs are removed, the remaining pairs of trajectories (C_{PR}) is ready for analysis. Each pair of trajectories included in the C_{PR} dataset can be analyzed to determine if the scenario is composed of Type A, B, or C trajectories. In a pair of trajectories representing a Type B scenario, one of the trajectories is removed since the removed trajectories are considered noise. In Type A or Type C scenarios, trajectories are trimmed by removing points from the end or the start of trajectories to comply with laws of physics that prevent multiple vehicles from occupying the same space at the same time. Steps 4 - 6 describe procedures used to remove noise or to trim trajectories.

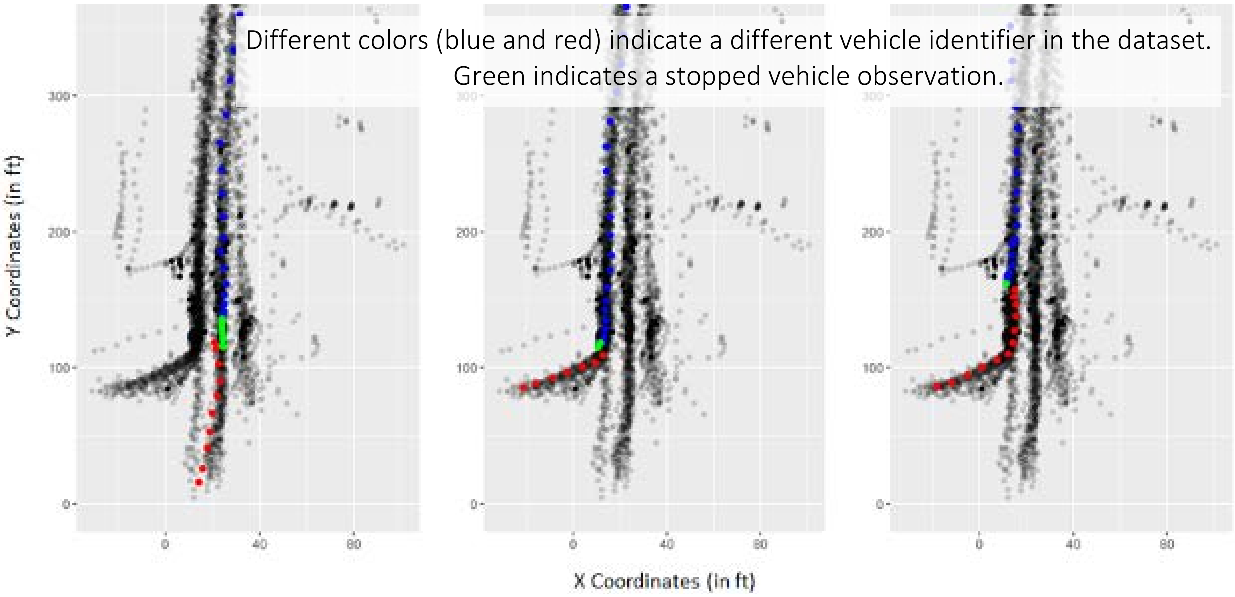


Figure 9. Example of Erroneous Conflicts Identified

2.2.10 Step No. 4: Removal of Noise Part of a Type B Trajectory Pair

Each trajectory pair part of C_{PR} can be analyzed to determine if the pair of trajectories represents a Type B scenario. In a Type B scenario, one of the trajectories is considered a ghost trajectory and therefore should be removed from the dataset. Determining if a trajectory pair represents a Type B scenario, requires the calculation of the values shown in Table 1. In the table, vehicle 1 represents the vehicle with the highest number of trajectory points with a speed value greater than zero. Using the values in Table 1, a pair of trajectories is considered Type B if any of the conditions listed ahead are satisfied in addition to the requirement that at least one of the trajectories in the pair crosses the stop bar.

- $R_{DY} <= 0.25$, $U_{Y1} > U_{Y2}$, and $L_{Y2} > L_{Y1}$, or
- $M_D < D_T$, $U_{Y1} > U_{Y2}$, and $L_{Y2} > L_{Y1}$, or
- $R_{DY} > 0.25$, $DX_L < 12$ feet, and $DX_S < 12$ feet.

Examples of pairs of trajectories classified as Type B are shown in Figure 10. Once classified as Type B, the trajectory identified as vehicle 2 within the pairs is removed from the full dataset. If none of the conditions are satisfied, a pair of trajectories is not considered a Type B thus prompting further evaluations to determine if the scenario represented is a Type C or a Type A.

Table 1. Values Computed to Evaluate if Trajectories Represents a Type B Scenario

Trajectory	**Description and Details**	**Notation**
Vehicle 1	Number of points with a speed value greater than zero	M_{O1}
Vehicle 1	Change in X coordinate between first and last point	DX_L
Vehicle 1	Distance traveled along the Y-axis	DY_L
Vehicle 1	Maximum Y Coordinate	U_{Y1}
Vehicle 1	Minimum Y Coordinate	L_{Y1}
Vehicle 2	Number of points with a value speed greater than zero	M_{O2}
Vehicle 2	Change in X coordinate between first and last point	DX_S
Vehicle 2	Ratio of Vehicle 2 to Vehicle 1 moving points, i.e., M_{O2} / M_{O1}	R_{DY}
Vehicle 2	Minimum distance between vehicle 2 and vehicle 1 trajectory points	D_M
Vehicle 2	Average distance between vehicle 2 and vehicle 1 trajectory points	M_D
Vehicle 2	Maximum Y Coordinate	U_{Y2}
Vehicle 2	Minimum Y Coordinate	L_{Y2}

2.2.11 Step No. 5: Identifying and Combining Type C Trajectory Pairs

Pairs of trajectories in the C_{PR} dataset not identified as representing a Type B scenario are analyzed to determine if the pair represents a Type C scenario. Figure 11 shows examples of pairs of trajectories identified as Type C. As shown in the figure, both trajectories appear to represent the path of a unique vehicle. The reason the two trajectories were marked for further analysis and included in the C_{PR} dataset is because of an overlap between the trajectories triggered a space-time conflict during the conflict detection process.

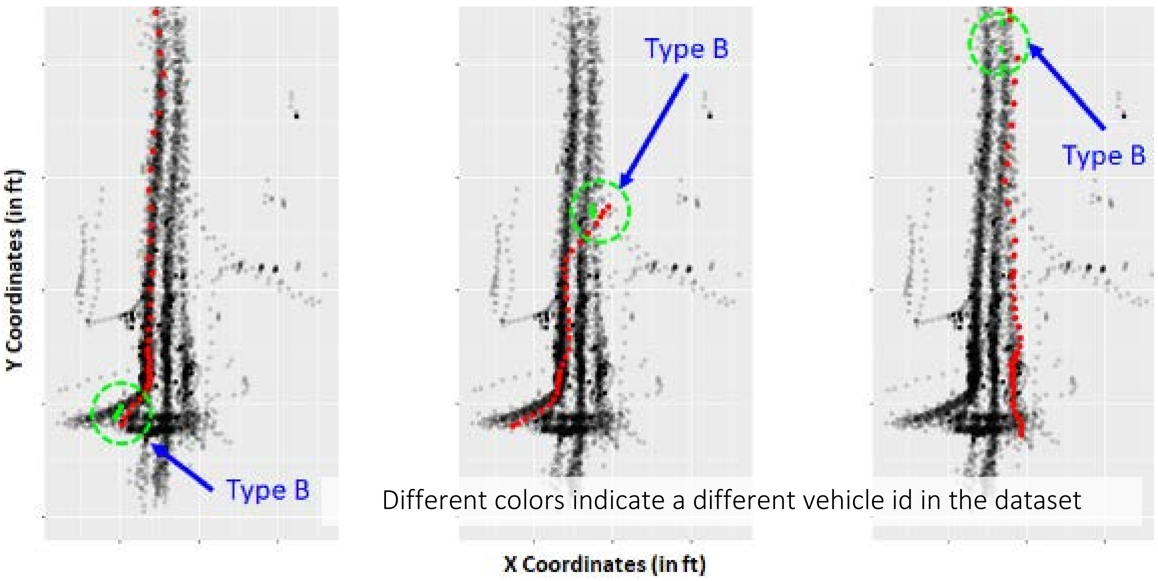


Figure 10. Example of Type B Trajectories Identified for Removal

If a pair of trajectories represents a Type C scenario, the two trajectories can be combined into one. If an overlap between the trajectories exists, combining the trajectories require trimming the end of the Vehicle 1 trajectory, as well as the start of the Vehicle 2 trajectory, could require trimming until no space-time conflict exists in the dataset. If no overlap exists, both trajectories can then be combined into one without further manipulation.

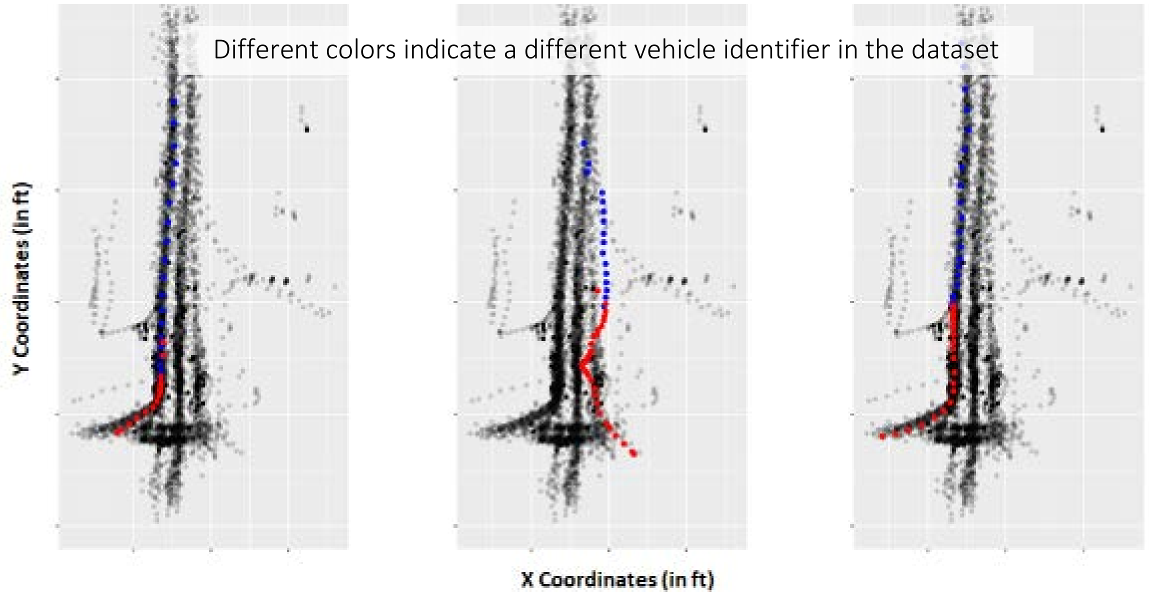


Figure 11. Example of Type C Trajectories Identified

In order to identify if a pair of trajectories represents a Type C scenario, the values shown in Table 2 are computed for each trajectory. In the table, vehicle 1 refers to the trajectory found as starting the furthest upstream on the approach, while vehicle 2 refers to the trajectory found to end the furthest downstream. Therefore, the first step in determining if a pair of trajectories represents a Type C scenario is for the $L_{Y2} < L_{Y1}$ and $U_{Y2} < U_{Y1}$ conditions to be satisfied.

Once the conditions are satisfied, the overlap between the trajectories is computed. P_{O2} and P_{O1} values are divided by the maximum of the N_{O1} and N_{O2} values. The smallest value resulting from the division is considered the overlap value. If there is no overlap, the pair of trajectories is considered a Type C. Furthermore, if trajectories overlap, and the overlap value is less than 0.25 the pair of trajectories is also considered a Type C. Once identified as Type C, the pair of trajectories is joined into one in the dataset. Prior to joining, any point that results in the overlap of trajectories is removed.

Table 2. Values Calculated to Identify, Trim, and Join Type C Scenario Trajectories

Trajectory	**Description and Details**	**Notation**
Vehicle 1	Number of points in trajectory with a speed greater than zero	N_{O1}
Vehicle 1	First Y coordinate in the trajectory	U_{Y1}
Vehicle 1	Last Y coordinate in the trajectory	L_{Y1}
Vehicle 1	Number of points with a speed value > 0 and with a Y coordinate < U_{Y2}	P_{O1}
Vehicle 2	Number of points in trajectory with a speed greater than zero	N_{O2}
Vehicle 2	First Y coordinate in the trajectory	U_{Y2}
Vehicle 2	Last Y coordinate in the trajectory	L_{Y2}
Vehicle 2	Number of points with a speed value > 0 and with a Y coordinate > L_{Y1}	P_{O2}

2.2.12 Step No. 6: Cleanup of Type A Pairs of Trajectories

If a trajectory pair in C_{PR} is not Type E, D, C, or B, the trajectory is treated as a possible Type A. Pairs of Type A trajectories appear to converge at some point in time while exhibiting the behavior of two different vehicles most of the time. To determine if convergence takes place upstream or downstream P_{DU} and P_{DD} values need to be calculated. P_{DU} is the distance between trajectories upstream while P_{DD} is the distance downstream. Both distance values are measured along the X-axis, i.e., the distance is measured perpendicularly along the Y-axis. If $P_{DU} < P_{DD}$ trajectories are considered to converge upstream; otherwise, trajectories are considered to converge downstream. In addition to the convergence upstream or downstream, the R_{DY} value (previously computed in Step 4) needs to be higher than 0.25 for a pair of trajectories to be considered Type A.

Once a pair of trajectories meet the conditions, the pair can be treated as a Type A scenario. Trimming the end or start of the trajectories requires computing the Table 3 values. In the table, vehicle 1 refers to the vehicle that appears to join the path, or diverge from the path, of vehicle 2. If the converge of points is upstream, trajectory points with Y coordinates higher than M_{Y1} are isolated for analysis. Points from the start of the trajectory are programmatically removed from the dataset until the change in M_{XU} caused by the removal of the points is less than 0.5 feet. Similarly, if the converge of points is downstream, trajectory points with Y coordinates lower than M_{Y1} are isolated for analysis. Points from the end of the trajectory are programmatically removed from the dataset until the change in M_{XD} caused by the removal of the points is less than 0.5 feet.

Table 3. Values Calculated to Identify and Trim Type A Scenario Trajectories

Trajectory	Description and Details	Notation
Vehicle 1	Average Y coordinate	M_{Y1}
Vehicle 1	Average X coordinate for trajectory points with a Y coordinate > M_{Y1}	M_{XU}
Vehicle 1	Average X coordinate for trajectory points with a Y coordinate < M_{Y1}	M_{XD}

After completion of Step 6, a clean and streamlined vehicle trajectories dataset (C_{VT}) is available for analysis. If the cleaning procedures described in the previous sections are successful, C_{VT} should result in a dataset that preserves actual vehicle trajectories associated with vehicles while eliminating noise. Section 2.3 assesses the accuracy of the C_{VT} dataset produced after the application of the filtering procedures described in the previous sections.

2.3 ACCURACY EVALUATION

The number of vehicles reported by the radar as crossing a known point on the approach can be easily computed. Due to the nature of the dataset, the time at which the crossing takes places can also be computed using an interpolation process. Vehicle volumes by 5-minute intervals can be computed from the C_{VT} dataset, and obtained after the application of the filtering techniques described in Section 2.2.6, by using the timestamp of the crossing.

2.3.1 Obtaining Ground Truth Volume

Using video from the Wisconsin Avenue and Meade Street intersection in Appleton, WI ground truth volume counts were obtained. A total of 6 hours of video were processed to obtain ground truth volume. Due to the focus on comparing vehicle volumes, the timestamp associated with each vehicle recorded using the video was the time when the vehicle crossed the downstream edge of the crosswalk. Therefore, the timestamp recorded for each vehicle acts as a surrogate of the time when vehicles crossed the Y_T point. No movement classification by vehicle took place since the timestamp recorded for each vehicle was obtained prior to a point where the movement of each vehicle was clear.

2.3.2 Side-by-Side Volume Comparison

Vehicle volumes obtained from the video were summarized in 5-minute intervals. Ground truth volume for each 5-minute period was compared with the volume for the same period obtained by analyzing the C_{VT} dataset using a Y_T value of 105 feet. For each interval in the dataset, the difference between vehicle volume from video and vehicle volume from an analysis of the C_{VT} dataset were computed and prepared for analysis. After processing 6 hours of video, a total of 70 intervals were available for analysis. The reason for 70 intervals over a 6-hour video period instead of 72 is that two video recordings were used and for each video recording the first 5-minute interval was not included in the analysis due to the time offset between the video and the radar dataset which yields an incomplete ground truth dataset.

A visual summary of the comparison of the vehicle volumes for each interval is shown in Figure 12. In the figure, a histogram describing the volume differences observed over 70 intervals analyzed is shown. As suggested by the figure, filtering techniques described are capable of removing noise from the raw data reported by the radar while having a minimal impact on the number of vehicles reported as crossing a line defined by Y_T. In fact, for 72.8% of the intervals, the difference between reported volume extracted from the C_{VT} dataset and that observed from ground truth volume observations is at most 1 vehicle. Across all 70 intervals, the average volume difference between intervals is -0.23 vehicles, and the average absolute difference is 1.05 vehicles.

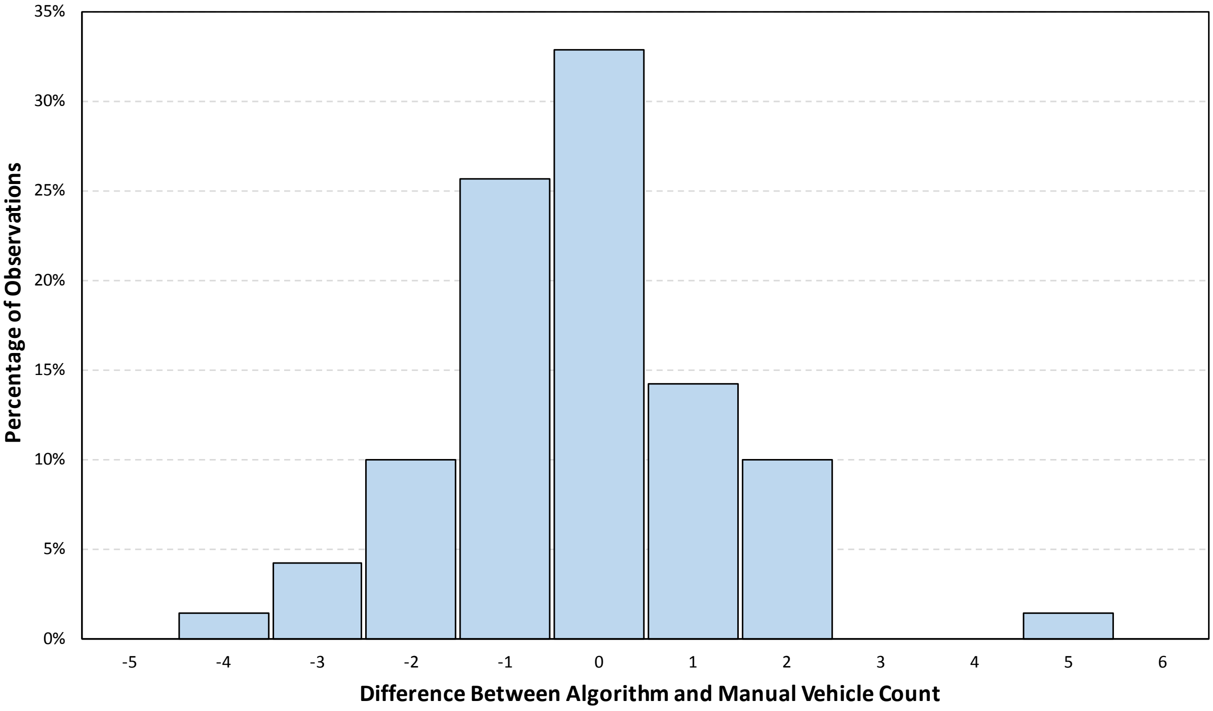


Figure 12. Error Analysis for Volume Obtained Using a Y_T value of 105 feet

As in the case of previous work, error values are reported in terms of vehicles instead of using a percentage. The argument for using vehicle numbers instead of percentage values is because reporting error as a percentage could be a misleading performance measure since a 1 vehicle difference for a low volume period is magnified while the difference is diminished on a higher volume interval. Given that the volume for the periods included in the analysis ranges from 12 to 46 vehicles with a median of 26 vehicles the 1.05 vehicles average absolute difference over a 5-minute interval suggests the C_{VT} accurately represents the actual conditions on the field.

2.3.3 Sensitivity Analysis

The error analysis presented in the previous section is based on a Y_T value equal to 105 feet. One of the most important functions of the Y_T parameter in the analysis process is the identification of Type E trajectories that need to be removed from the analysis. Selecting a Y_T value that is too high could lead to the misclassification of vehicles on the approach as Type E trajectories thus resulting undercounting. A Y_T value that is too low could result in overcounting of vehicles since vehicles from the conflicting approach that should be considered Type E trajectories are included in the C_{VT} dataset instead of been removed. Figure 13 shows the average vehicle difference per 5-minute interval as a function of Y_T.

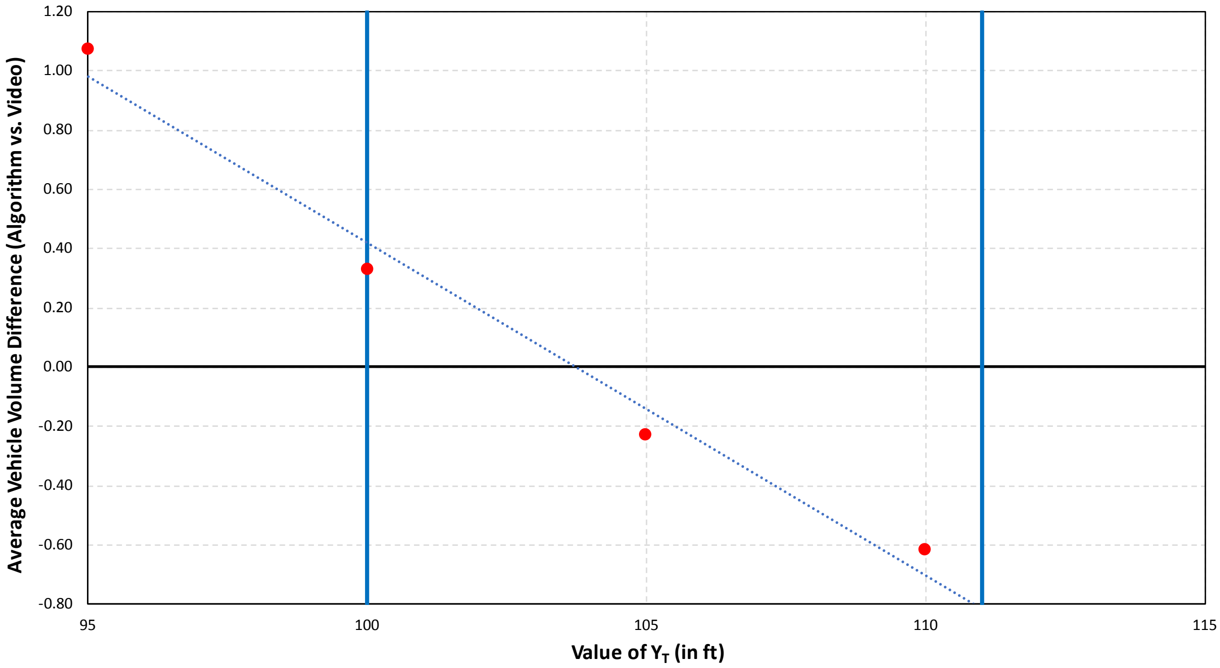


Figure 13. Average Vehicle Volume Difference as Function of Y_T

As Figure 13 shows, the impact of Y_T values on the average difference is as expected with high-end values resulting in undercounting and low-end values resulting in overcounting. In the figure, the vertical lines shown at 100 feet and at 111 feet represent the position of the Y_{B1} and Y_{B2} asymptotes previously shown in Figure 7. Based on the behavior observed in the figure, the optimal value for Y_T represents a Y coordinate upstream of the Y_{B1} asymptote but downstream of the Y_{B2} asymptote.

2.4 IMPLICATIONS FOR COMMERCIALIZATION

The previous sections described filtering techniques that provide a research-level approach to noise removal and characterization. As a result of the noise removal procedures described and characterization effort, a detailed understanding of the type of noise expected in a vehicle trajectories dataset was obtained. From a commercialization perspective, there is one item described that will provide the greatest impact on noise removal and therefore improve the quality of the vehicle trajectory dataset that is used to obtain performance measures from the intersection such as vehicle volume. That item is the use of a horizontal asymptote, Y_T, to identify those vehicle trajectories that should be considered as valid for performance monitoring purposes. Based on feedback from the research team, the commercialization partner added an interface to their data collection device that makes it possible for the user to specify the location of the Y_T asymptote; thus, improving reliability and eliminating the complexities associated with the automated detection procedure used by the algorithm developed as part of the Type 1 IDEA project.

CHAPTER 3. ASSIGNING LANE INFORMATION TO VEHICLE TRAJECTORIES

The dataset created after the application of the filtering procedures outlined in Chapter 2 is referred to as C_{VT} and will be treated as containing an accurate representation of the traffic that uses the intersection approach. While accurate, one of the limitations of the C_{VT} dataset is that no information about the lane used by vehicles is included. Certainly, the X coordinates of the vehicle trajectories can be used for lane assignments. However, such an approach will require the definition of fixed lane boundaries that could produce lane assignment errors when small shifts in vehicle paths occur due to common weather events. When traditional zone-based detection is used, obtaining detailed performance measures summarized by lane is a challenge due to the potential for false calls resulting from the definition of fixed boundaries. When a vehicle travels closer than expected to the edge of a lane, there is a risk for the activation of a zone in the adjacent lane which produces a false call. Therefore, at the very least, these false calls produce incorrect lane distributions which can negatively affect the quality of performance measures computations that rely on lane assignment values.

The sections ahead outline the steps that can be followed to assign lane information to vehicle trajectories part of the C_{VT} dataset. The sections ahead demonstrate that by analyzing trajectory data, accurate lane classification information can be assigned to all vehicles that exit an intersection approach without the need to define lane boundaries. The analysis procedures described overcome the limitations of existing lane classification procedures that rely on fixed boundaries since by relying on trajectory data false calls are virtually eliminated. A well know center-based clustering technique, known as k-means, is key to the analysis procedures. All procedures presented ahead were implemented using the R programming language; however, these procedures can also be implemented in other programming languages such as Javascript and Python.

3.1 K-MEANS CLUSTERING

Visual identification of lane centers from a trajectory data plot is a simple task. However, numerical procedures that automate the identification of the middle of a lane require the identification of clusters of trajectory points along the paths followed by thru-traveling vehicles. The selection of cluster locations needs to be done systematically and based on objective measurements. In the lane classification methodology presented in the sections ahead, cluster identification is achieved through the use of the k-means cluster analysis technique for which abundant documentation exists. Gan, Ma, and Wu highlight that the k-means method is partitional (or nonhierarchical) and is one of the most used clustering methods [4]. By design, the k-means method clusters numerical data around a center called the mean of the cluster or centroid. With the k-means method, each point in a dataset is assigned to a centroid that is the closest based on a distance function that, in the case of the C_{VT} dataset, is based on a distance measured along the X-axis.

3.1.1 UNDERLYING THEORY AND AVAILABLE LIBRARIES

Identifying the location of centroids within dataset requires the use of a search algorithm. For the lane classification scheme presented, the Hartigan and Wong algorithm is used [5] to find the centroids. The goal of the Hartigan and Wong algorithm is to divide the dataset provided, e.g., the x coordinates of thru-traveling vehicles near the stop bar, into a number of clusters, *K*, in such a way that an objective function value is minimized [4]. The objective function for which the algorithm will find the optimal (lowest) value is shown in the equation ahead.

$$P(W,Q) = \sum_{j=1}^{k}\sum_{i=1}^{n} w_{ij} d_{euc}\left(x_i, q_j\right)$$

In the equation shown, *k* is the number of clusters, *n* is the number of observations in the dataset, $d_{euc}(x_i, q_j)$ is a function that computes the Euclidean distance between a data point and the assigned centroid, and w_{ij} controls the impact that d_{euc} will have on the calculation. Computational steps required to find cluster centroids and that minimize the objective function shown above can be implemented via the *k-means* method part of the *stats* library of the R programming language [6]. The library used allows the implementation of k-means in a fully automated approach as well as through a guided approach the controls the number of iterations and starting point for the identification of clusters.

3.2 LANE CLASSIFICATION PROCEDURE AT STOP BAR

Assigning a lane to vehicles in the C_{VT} dataset identified as exiting the stop bar requires the identification of lanes that cross the Y_{B1} asymptote on the approach. Once identified, the x coordinates of the vehicle positions immediately upstream of the stop bar are isolated for analysis. The x coordinates are used to identify the center of clusters representing the center of the paths followed by the vehicles. Lane values are then assigned to each vehicle based on proximity to the centers when crossing the stop bar.

3.2.1 IDENTIFY LANE BOUNDARIES AT STOP BAR

Each vehicle in the C_{VT} dataset that crosses the Y_{B1} asymptote is identified and referred to as V_{SB}. Trajectory points immediately upstream of the stop bar for vehicles part of S_{BV} are isolated for analysis if the points represent the position of the vehicle while moving. The subset of points is referred to as C_{XD}. Only the position of vehicles while in motion is considered to avoid the inclusion of points in the analysis that skew the results towards the stopped position. Cluster centers for the X coordinates within the C_{XD} dataset are then identified using the Hartigan and Wong algorithm implementation using the number of lanes at the stop bar as the input. Figure 14 shows a histogram of the x coordinate within C_{XD} along with the cluster centers identified and displayed as red dashed lines.

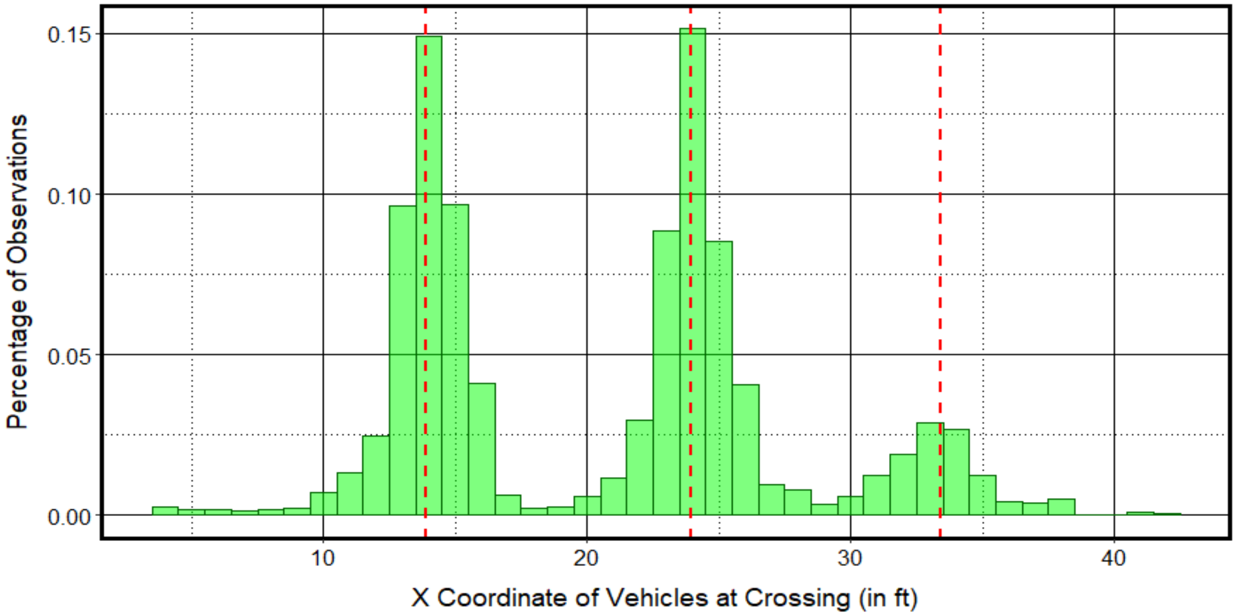


Figure 14. Center of Clusters at Stop Bar

3.2.2 ASSIGNING LANE IDENTIFIERS TO TRAJECTORY DATA

For each vehicle part of V_{SB}, the coordinate at the time of the stop bar crossing is identified. To identify the x coordinate at the time of the crossing the closest observations to the stop bar upstream and downstream are identified. Using the two observations, interpolation is used to determine x coordinate which represents the crossing location (X_{SB}). The distance between the x coordinate of the stop bar crossing moment and the x coordinate of lane cluster centers is then computed for each vehicle. Based on the proximity to the cluster centers a lane assignment is made for each vehicle in the C_{VT} dataset.

Lane assignments are made based on what is the closest center to X_{SB} value of each vehicle in the V_{SB} dataset. Figure 15 shows a plot of all points associated with the vehicles in the V_{SB} dataset. Trajectory points in the plot are color-coded by lane. In the figure, points associated with the position of a stopped vehicle are not included to make the data clearer. As the figure shows, lane assignment values appear to be accurate. An objective evaluation of the accuracy of the lane assignment process is discussed in Section 3.4.

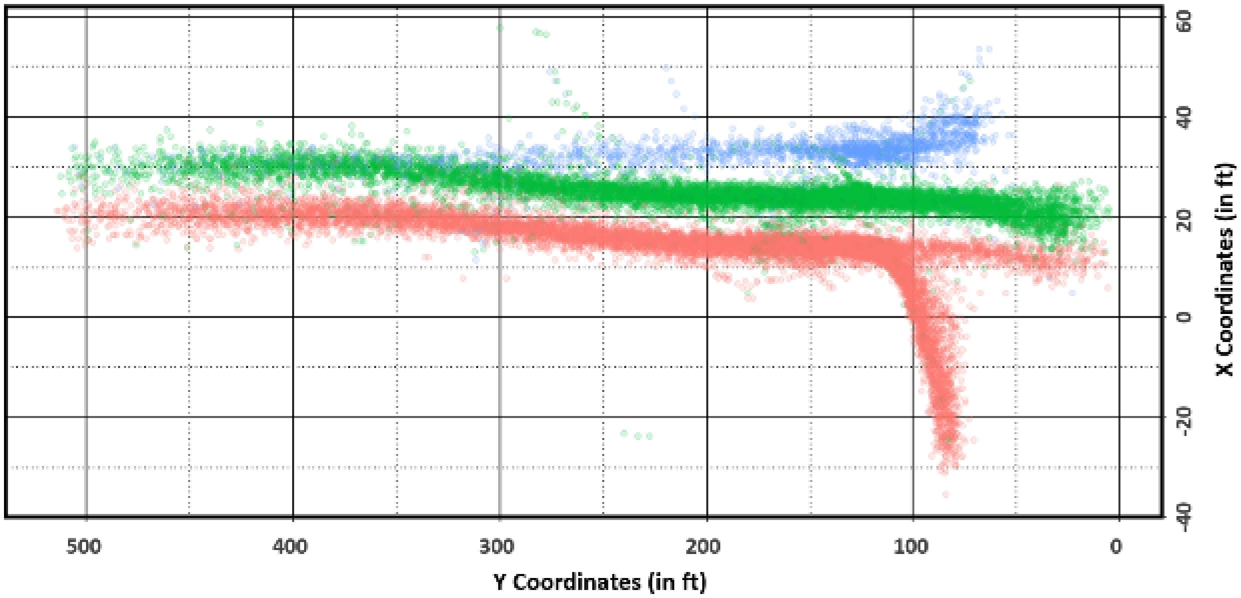


Figure 15. Trajectories Dataset for Vehicles that Cross the Stop Bar

3.3 LANE CLASSIFICATION UPSTREAM OF STOP BAR

For the purpose of determining vehicle movements, understanding the lane of the vehicle at the stop bar, or more precisely at the Y_T asymptote, is enough since the movement and lateral changes used to classify a vehicle as going through, right, or left happens downstream of the stop bar. However, if lane boundaries upstream of the stop bar are needed, then these boundaries can also be identified using k-means clusters. Once a V_{SB} dataset with a departure lane assignment is available, the full trajectories of vehicles in the dataset are available for analysis. All trajectory points associated with the dataset can then be segmented by the Y coordinate value.

Once segmented, the k-means clustering procedures are run for each segment of trajectories and the lane boundaries identified using identical procedures to those described in Section 3.2. Figure 16 shows a visualization of lane boundaries produced using the k-means clustering procedures to identify clusters and boundaries upstream of the stop bar. As the figure shows, the center of lanes and boundaries can be correctly identified using the k-means clustering procedures described. However, when running a cluster analysis, a key decision is what is the length of a segment along the Y-axis, i.e., how many iterations of the clustering detection procedures need to be performed.

3.3.1 SELECTING THE LENGTH OF SEGMENTS FOR CLUSTER ANALYSIS

Selecting the length, along the Y-axis, of segments upstream of the stop bar for which iterations of the cluster detection procedures need to be executed can be done by relying on the speed of the intersections approach. A good rule of thumb for selecting the length of the segment is to select a length equal to the distance traveled by a vehicle over a period of one second. For example, for an approach with a posted speed equal to 35 MPH, a vehicle is expected to travel 51 feet over a period of 1 second. Therefore, for a 35 MPH approach, a segment length of 50 feet can be used. The goal of selecting the distance traveled over a 1 second period for the length is that the frequency at which trajectory data is collected, 2 Hz, would result in at least one vehicle trajectory point falling within the length covered by the segment.

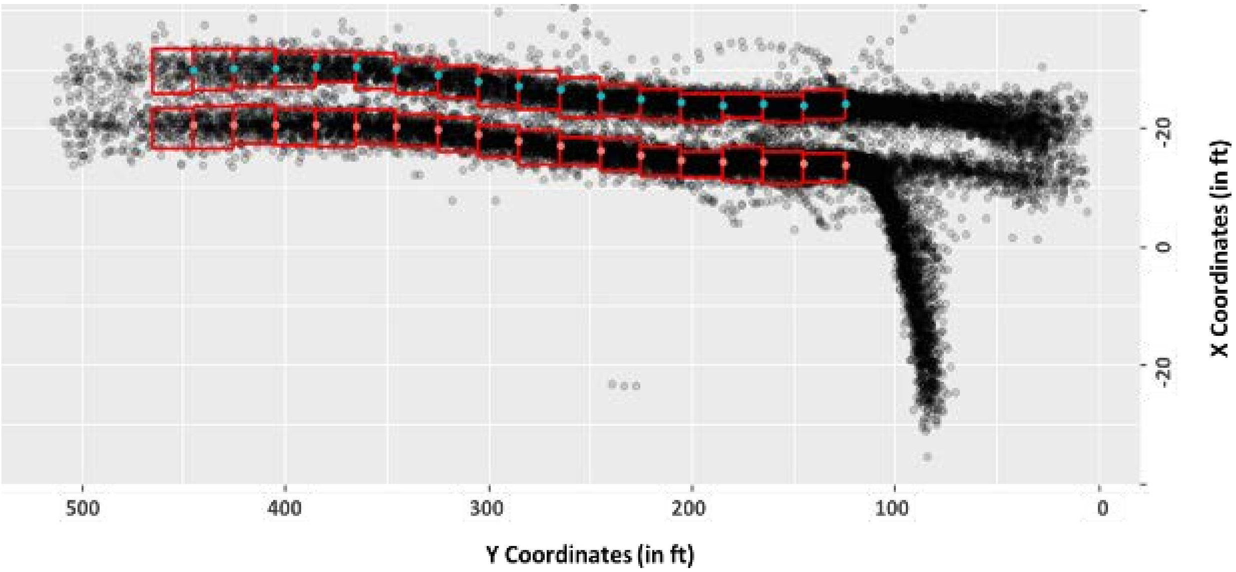


Figure 16. Upstream Lane Boundaries Identified

3.3.2 SELECTING TRAJECTORIES FOR IDENTIFYING LANE BOUNDARIES UPSTREAM

In the case that the approach for which lane boundaries upstream of the stop bar need to be identified do not have the same number of arrival lane as departure lanes, a typical scenario when channelized lanes exist, the continuous lanes along the approach should be identified. Once the continuous lanes are identified, the V_{SB} dataset can be queried for trajectories associated with these lanes and used for detecting clusters by segments. In fact, in Figure 16, only trajectory points associated with continuous lanes along the approach are shown and used to identify the cluster centers shown.

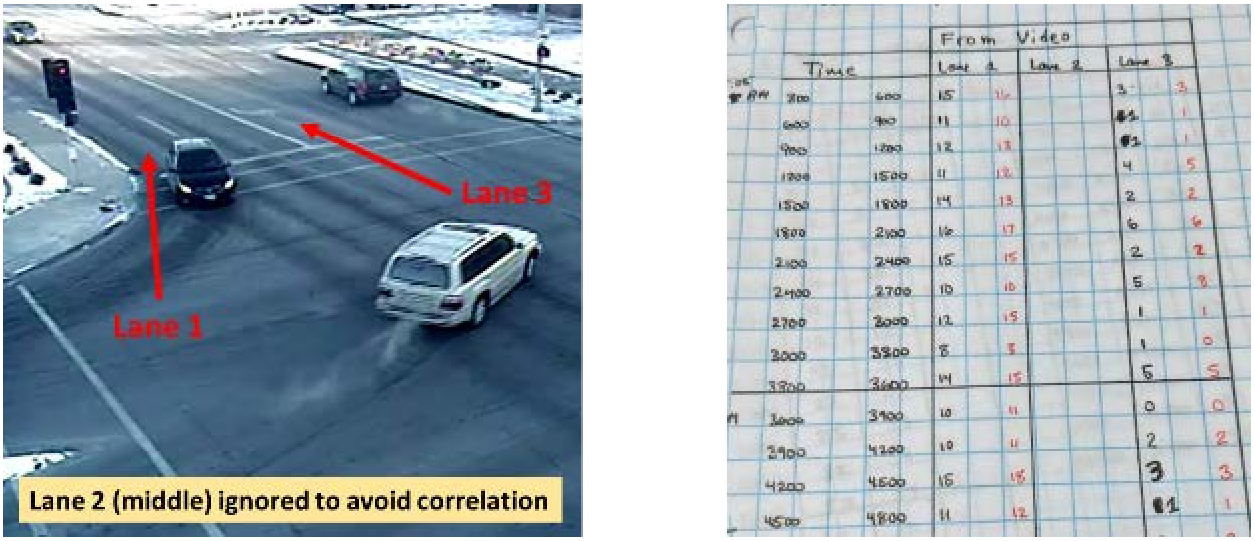
Due to the multiple upstream-downstream lane configurations that are possible, the procedures for identifying lane boundaries upstream could require a mapping between upstream and downstream lanes. As a result, that makes the procedures better suited for user-controlled evaluations instead of fully automated and algorithm-driven evaluations. A user-controlled analysis process has the potential for better quality control that can be beneficial for the type of safety and operational evaluations that requires lane classification upstream of the stop bar.

3.4 ACCURACY OF LANE CLASSIFICATION PROCEDURE

Understanding the accuracy of the lane classification process described in the previous sections requires comparing ground-truth lane classification observations with the corresponding observations obtained through the procedures described in the previous sections. The comparison procedures and results are presented ahead. As will be shown, results presented suggest that the lane classification procedures provide a high degree of classification accuracy.

3.4.1 OBTAINING GROUND TRUTH VEHICLE VOLUME BY LANE

A video-based manual count procedure was used to obtain vehicle counts classified by lane on the Southbound approach of the Wisconsin Avenue and North Meade Street. A screenshot of the approach is shown in Figure 17. In the figure, the rightmost lane (along the direction of travel), as well as the leftmost lane, are highlighted since those were the lanes for which volume data were collected. The timestamps when vehicles on Lane 1 and Lane 3 crossed the crosswalk were documented, and the volume per lane was summarized into 5-minute intervals. Vehicle volume data by lane for a total of 70 intervals were documented. An example of the summary sheet used to document the ground truth volume is also shown in Figure 17.



	Time		From Video				
			Lane 1		Lane 2	Lane 3	
AM	300	600	15	16		3	3
	600	900	11	10		1	1
	900	1200	12	13		1	1
	1200	1500	11	12		4	5
	1500	1800	14	13		2	2
	1800	2100	16	17		6	6
	2100	2400	15	15		2	2
	2400	2700	10	10		5	3
	2700	3000	12	15		1	1
	3000	3300	8	5		1	0
	3300	3600	14	15		5	5
	3600	3900	10	11		0	0
	3900	4200	10	11		2	2
	4200	4500	15	15		3	3
	4500	4800	11	12		1	1

Figure 17. Generating Comparison Dataset

Vehicle volumes by interval for Lane 2 were not documented in the ground truth dataset since the purpose of generating the dataset is understanding the accuracy of the classification process. Since the vehicle volume for an interval is fixed, errors associated with Lane 2 are correlated to errors in Lane 1 and/or Lane 3 and could lead to misleading average error values by canceling the effect of pairs of volume differences.

3.4.2 ACCURACY OF K-MEANS LANE CLASSIFICATIONS

For each of the 5-minute intervals for which ground truth vehicle volume was manually obtained the corresponding volume obtained by analyzing the V_{SB} dataset was also obtained. A visualization of the volume values on both datasets is shown in Figure 18 using a scatter plot with a 1:1 reference line. The figure shows that across all volume magnitudes reported, ground truth and algorithm-based vehicle volume are generally in agreement. While a total of 70 pairs of observations are represented in the figure a lower number appears visible due to the pairs with the same values overlapping each other.

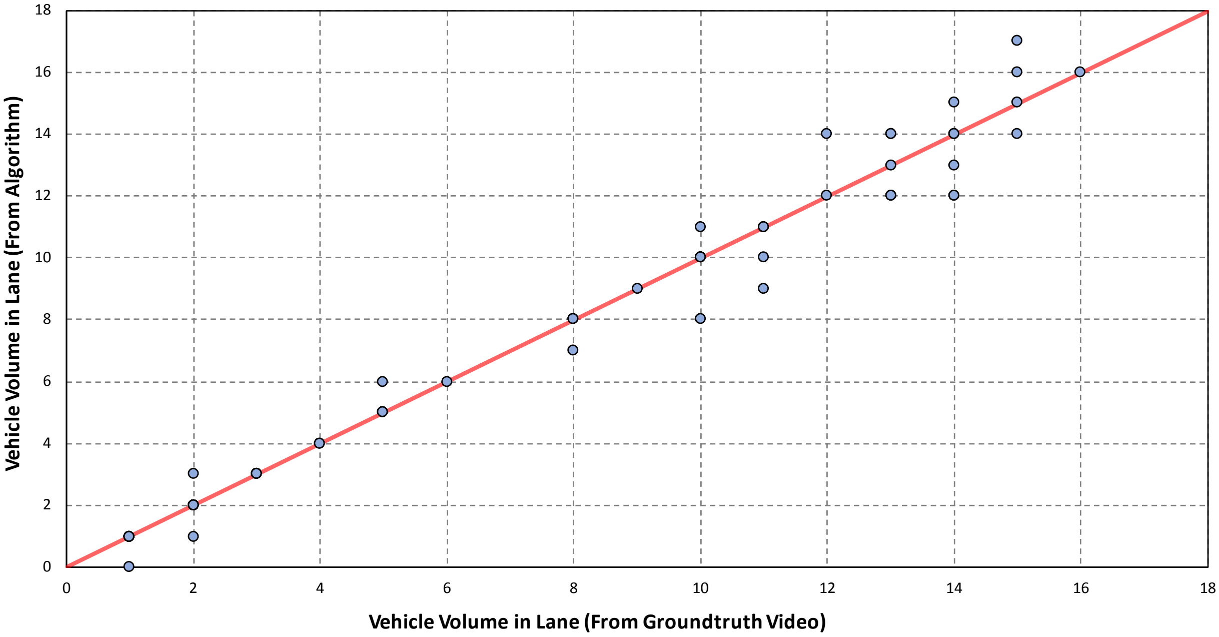


Figure 18. Lane Volume from Algorithm and Lane Volume from Ground Truth Video

Figure 19 shows a histogram visualization of the volume comparison shown in Figure 18. As the figure shows, results from the lane classification process presented are promising and highlight the quality of the results that is possible. As shown in the figure, volume differences in the comparison dataset for 91% of the 5-minute intervals considered in the analysis range between -1 and 1 vehicles. The average volume difference across intervals for the comparison dataset is -0.07 vehicles, while the absolute average difference is 0.40 vehicles.

3.5 LIMITATIONS FOR TRAJECTORY DATASETS

A challenge with the clustering procedures described in the previous sections is that no constraints are placed on the distance between clusters. Therefore, while unlikely, it is theoretically possible that in scenarios where the vehicle volume on one lane is significantly lower than the volume on another lane the k-means method could identify two clusters within the same lane. In other words, a possibility exists that two cluster centers are reported as being physically closer than theoretically possible. Under such a scenario, three alternatives should be considered. First, increase the time covered by the dataset analyze to increase the vehicle volume included in the dataset analyzed. A second alternative is to select a set of starting centers that are spaced apart using a common lane width and reduce the number of iterations conducted by the algorithm to identify the cluster centers. A final, and third alternative, involves adding constraints to the clustering algorithm to prevent the identification of clusters that violate the expected lane widths.

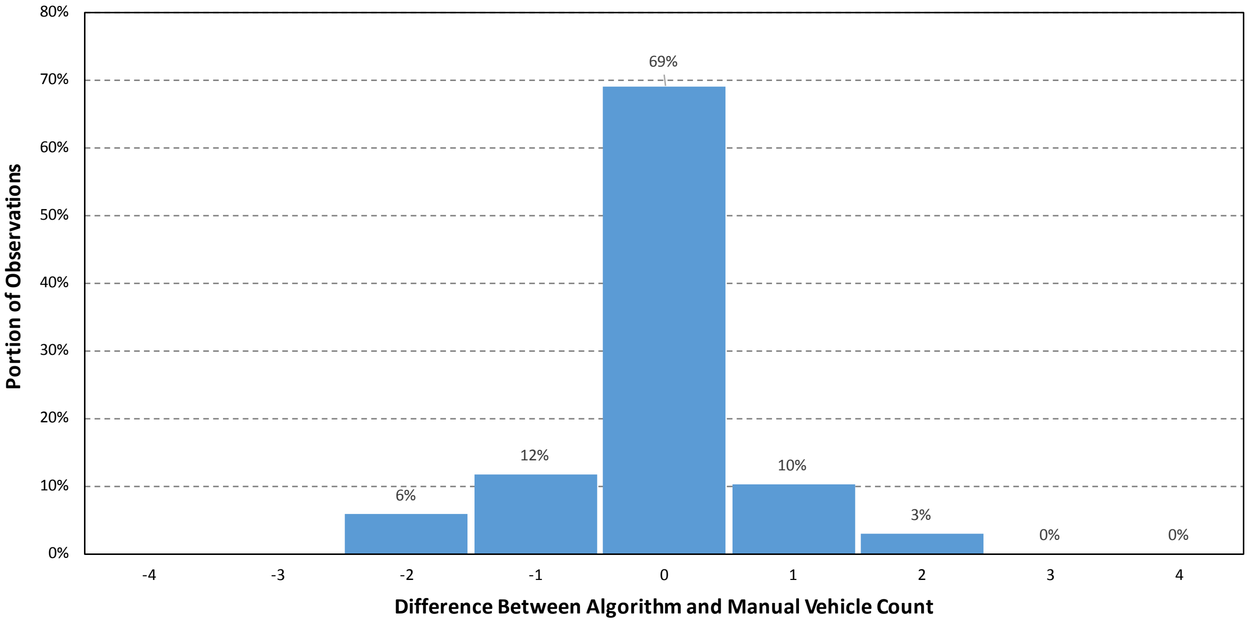


Figure 19. Histogram of Lane Classification Differences

3.6 IMPLICATIONS FOR COMMERCIALIZATION

The procedures described can be used to improve lane prediction when no information about lane boundaries is available during an analysis procedure. However, since a data collection system implemented by the commercialization partner provides lane boundaries by default, the techniques described in this chapter can be used when having to conduct a detailed research-level analysis of vehicle trajectory data during adverse weather events such as snowstorms. The reason for use during adverse weather events is that during these events, pre-defined lane boundaries are usually not followed and the classification procedures outlined in Appendix C, which rely on pre-defined boundaries, will not be sufficient.

CHAPTER 4. PLANS FOR IMPLEMENTATION

In coordination with the commercialization partner, an improved version of the previously developed classification algorithm is expected to be available to customers as a software update after going through the corresponding licensing processes at the University of Wisconsin-Madison. The data collection device that can support the new version of the classification algorithm through a software upgrade is already available as a product. The commercialized device includes a new configuration interface to support the collection of the necessary data for running the improved classification algorithm and other performance measures that can be derived from vehicle trajectory datasets. The device, which can run data collection and analysis tools developed by the research team, allows for future expansion opportunities and the continued improvement of analysis procedures.

Changes made to the device platform as a result of lessons learned during the project will enable commercialization of not only the existing research but future research. Some of the changes made to the device platform by the commercialization partner after considering feedback from the research team and their own experienced are presented ahead. Also presented is a description of some of the changes to the data collection, storage, analysis procedures, and recommended steps for the best implementation of the classification algorithm. The term platform is used to describe the software that makes commercializing the research possible. From the hardware perspective, everything described is made possible by relying on a Raspberry Pi computer that has additional storage. Figure 20 shows an example of a prototype of the data collection device installed inside a signal cabinet.

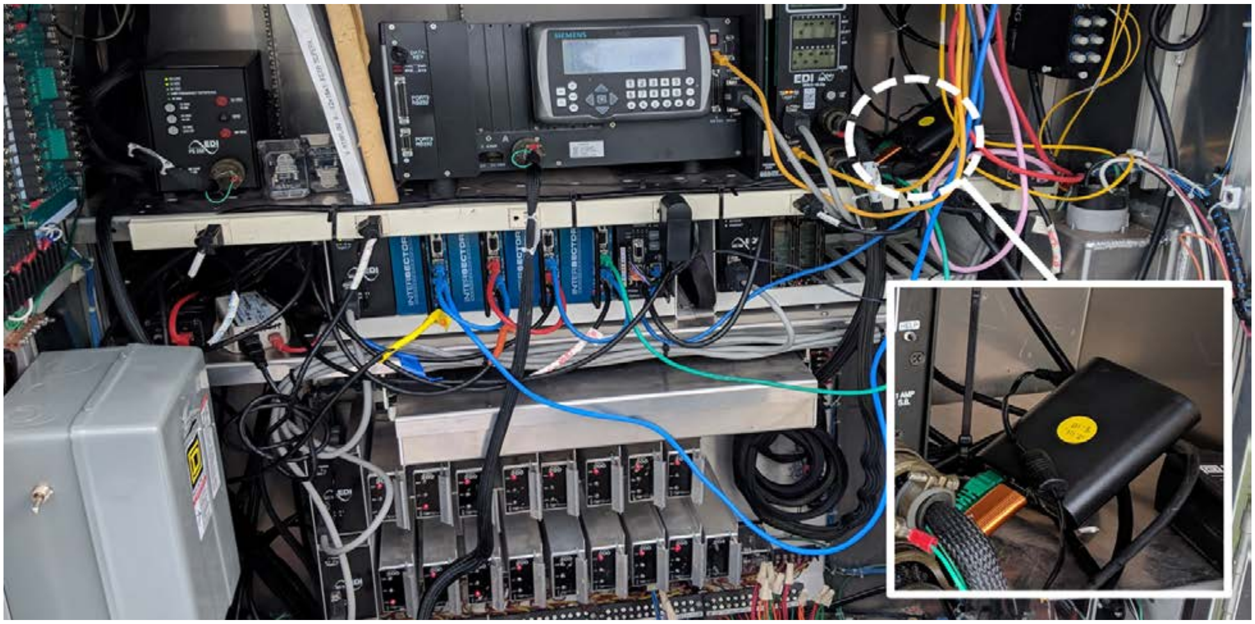
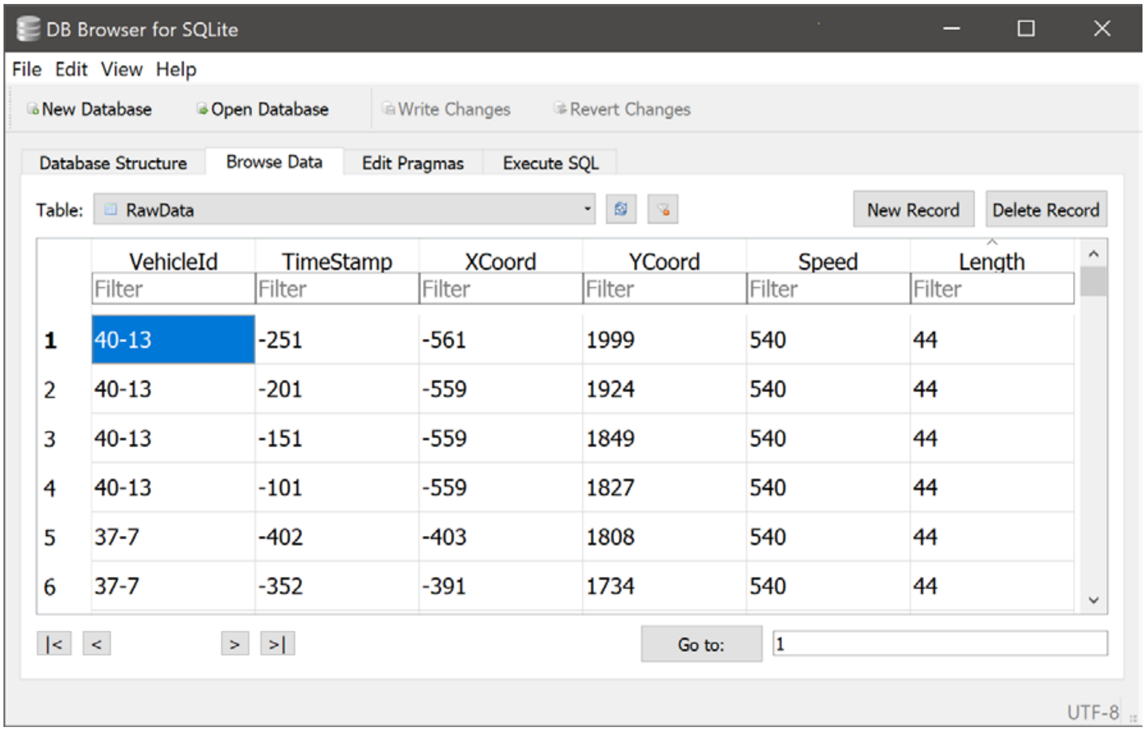


Figure 20. Example of Prototype Installed at Signalized Intersection

4.1 CHANGE OF DATA STORAGE FORMAT

In coordination with the commercialization partner, the research team evaluated the data collection and storage practices that were implemented in an early prototype of the data collection system. Early versions of the data collection system stored numbers associated with vehicle trajectories using a single (or double) precision number format. In other words, when a value such as 14.4 was stored up to 64-bits (8-bytes) of memory were used. This data storage procedure was certainly an architectural overlook likely driven by the inertia of default settings. When the type of data produced by the radar sensor was considered, as well as the type of analyses that used the data, it became obvious that storing trajectory data using 2 to 4-bytes integers through scaling made more technical sense. In other words, a change was made in the database and data collection procedures to store values such as 4.4 as 44 and then handle the conversion to the original value during the data analysis process. Figure 21 shows a screenshot of sample trajectory data collected using the new data storage format.



	VehicleId	TimeStamp	XCoord	YCoord	Speed	Length
1	40-13	-251	-561	1999	540	44
2	40-13	-201	-559	1924	540	44
3	40-13	-151	-559	1849	540	44
4	40-13	-101	-559	1827	540	44
5	37-7	-402	-403	1808	540	44
6	37-7	-352	-391	1734	540	44

Figure 21. Example of New Data Collection Format

The result of the aforementioned change, while simple in nature, makes it possible to store approximately one year of vehicle trajectory data on the local data collection device while minimizing the amount of "industrial-grade" storage that needs to be attached to the data collection system. This change in the storage format also made a previously planned cloud storage approach unnecessary thus eliminating/reducing barriers for product adoption. Therefore, the final product that will be commercialized will not require any cloud connectivity but will still benefit from it by providing a web-based interface.

4.2 CHANGE IN UNDERLYING OPERATING SYSTEM PLATFORM

A previous version of the data collection and analysis platform relied on research-style software tools to implement the noise removal procedures and an early version of the classification algorithm. In the past, data collection was handled via Python scripts or a VB.Net program while data analysis and noise removal were handled using scripts written in the R programming language. These tools are powerful and remain valuable. In fact, the noise removal and lane classification procedures described in this report as improvements were developed and tested using these tools. However, deploying a commercial product that uses tools such as the R programming language requires a level of back-end technical complexity that was deemed unnecessary.

Based on experience during testing and by considering the software requirements of the project, the commercialization partner changed the underlying operating system used to collect vehicle trajectory data. The operating system selected to support the data collection system is a Linux distribution geared towards embedded devices known as LEDE; at the time of the decision, LEDE was a “fork” of the popular OpenWRT distribution. One of the advantages of relying on the aforementioned Linux distribution was that it made it possible for the data collection system to have a web-based interface for configuration thus enabling a shift of the data analysis procedures to the client-side of the typical client-web-server relationship. Details of the architecture used in the final version of the prototype shown in Figure 20 are discussed in the sections ahead.

4.2.1 Data Collection Interface

The data collection system suffered several changes over the years and during this project that help in the process of commercialization. Beside one a key change, the one described in Section 4.1, additional changes to the data collection interface have been made. Changes to the data collection interface were made as a result of experience gained through additional data collection, deployments, and analysis iterations. In the past, the role of the data collection system was primarily focused on recording raw data that was then analyzed by scripts written in the R programming language. Now, the data collection system consists of two modules. The first module is a streamlined version that collects raw vehicle trajectory data, and the second provides vehicle summary information about a data collection period.

On a nightly basis, when the data collection system shown in Figure 20 is deployed on the field, the data collection module summarizes vehicle trajectory and eliminates noise using a subset of the procedures described in Chapter 2. The summarizing procedure takes advantage of functionality added by the commercialization partner to the platform which allows the end-user of the system to specify the position of the Y_T (effective start of the physical intersection) which is the key parameter during the noise removal process. Once the device shown in Figure 20 is installed inside the signal cabinet, it can be configured over the web (if a network connection exists to the signal cabinet) or via WiFi. The wireless connectivity to the device is possible because the operating system selected to run on the hardware essentially converts the Raspberry Pi device into a router. Furthermore, an interface known as LuCI that runs on top of the operating system enables configuration of all data collection parameters using a web-based interface like the one shown in Figure 22.

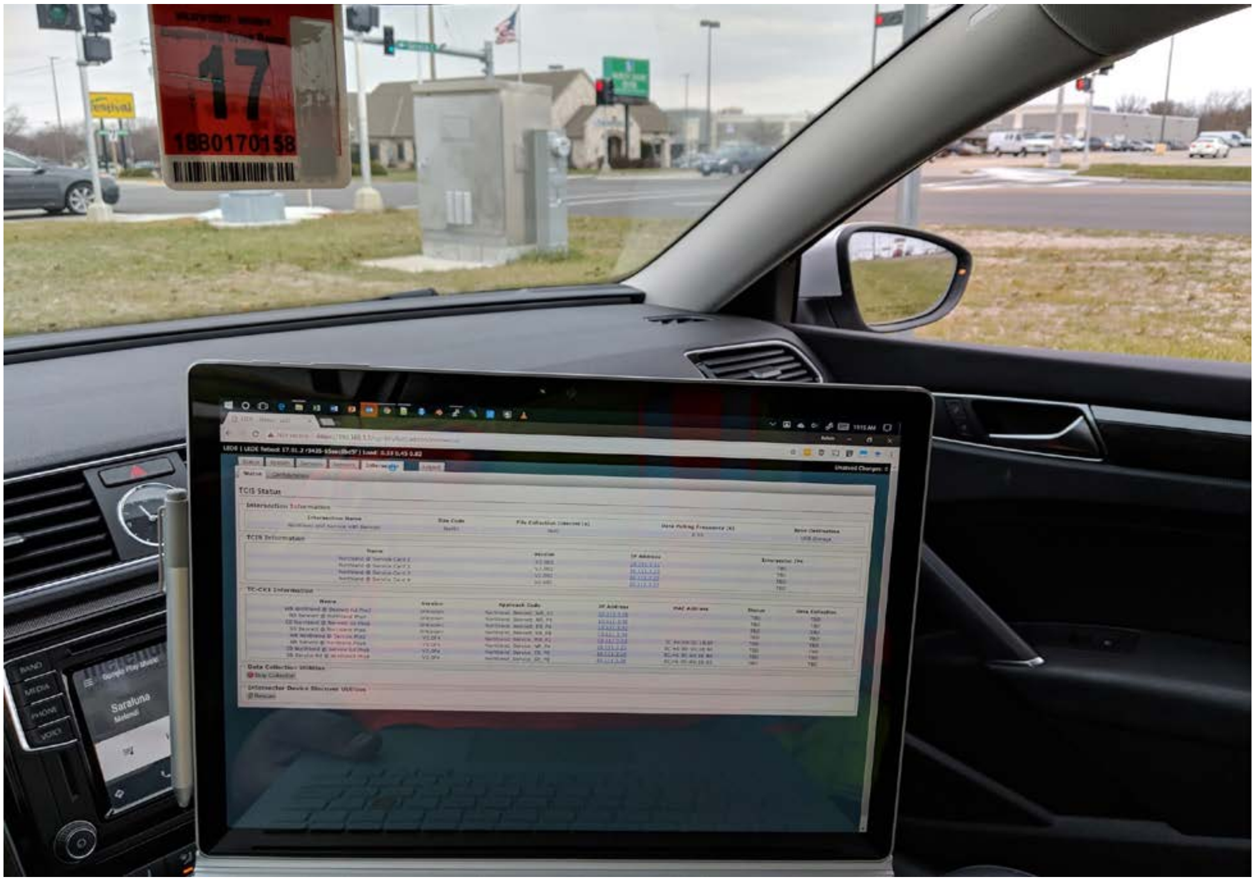
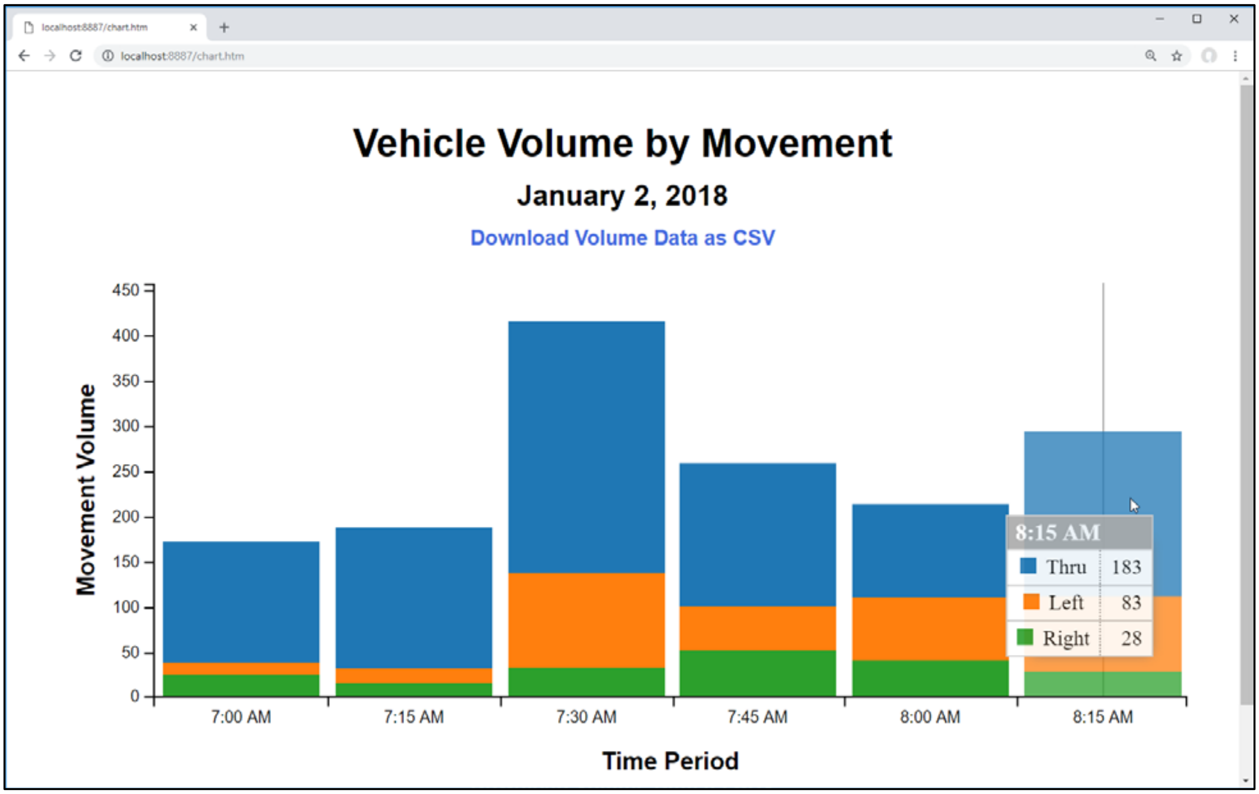


Figure 22. Example of Remote Connection to Device

4.2.2 Data Analysis Interface

One of the advantages of the operating system selected as the supporting platform is that it makes it easy for the data collection system to also act as a web server. By acting as a web server, it enables the analysis procedures that produce performance measures such as volume metrics to be performed by a web browser instead of the software running directly on the device. Running the analysis procedures on the web browser is possible by writing the data analysis scripts using the JavaScript programming language and creating a web page that displays the results and makes it possible to select different analysis parameters. In coordination with the commercialization partner, a data exchange format was created (along with the necessary software running in the data collection device) to expose vehicle summary data using JSON format thus simplifying the process of developing analysis scripts using JavaScript. Data exposed in JSON format includes information about the location of raw trajectory files, raw vehicle summary files, and geometric information about the intersection.

Therefore, besides for the procedures that summarize vehicle trajectory data, no other analysis script is executed on the data collection device. When an end-user wants to access performance measures from an intersection, it will connect to the device via the WIFI network (or over a wired network), type the IP address of the data collection device on a web browser and select the time period, and approach, from which performance measures are desired. The web page that will be displayed to the user will then execute JavaScript code on the user browser that will query vehicle summary information exposed on the data collection format using JSON and generate/visualize performance measures. The analysis and visualization capabilities are enabled by the D3.js and C3.js libraries. The visualizations enabled by the platform are vector-based, thus making them viewable at different resolutions; the visualizations are also dynamic, thus allowing the user to hide individual aspects. An example of the type of visualization possible by using the C3.js framework and the type of data exposed by the data collection system is shown in Figure 23.



8:15 AM	
🟦 Thru	183
🟧 Left	83
🟩 Right	28

Figure 23. Example of Visualization Possible with the C3.js Framework

Based on feedback received during the project, functionality was also added to download the data associated with the visualizations as well as the underlying raw trajectory data. Access to raw trajectory data over the network makes it possible to expand the capabilities of the system and to develop meta-analysis techniques that look at network-wide data and create visualizations. Expanding the capabilities of the system to enable network-wide analyses will open the doors to monitoring procedures that today require advanced signal controller infrastructure and high-resolution data collection capabilities.

4.3 RECOMMENDED IMPLEMENTATION APPROACH

Based on the results of the analysis procedures described in Chapters 2 and 3, by considering the analysis procedures developed in the Type 1 research effort, and by considering feedback during the additional data collection conducted for this project the sections ahead describe recommendations that will help the commercialization partner take a final product to market. Most of these recommendations have been already implemented and are available to test customers as part of the data collection system created by the commercialization partner; others, specifically those related to trajectory classification, should be implemented prior to making a final product available to customers. Recommendations are grouped into two categories: infrastructure and configuration, as well as software implementation. The sections ahead describe the recommendations made for each category.

4.3.1 Infrastructure and Configuration

The research team has had the opportunity to collect trajectory data over many years and to review different versions of trajectory datasets, including merged trajectory and video datasets (see Appendix A). Based on the experience gained, the following are key things that should be considered from the perspective of how to configure the radar device for optimal data collection.

- Position the radar in such a way that the line of sight is as close to the centerline of the monitored approach as possible. By doing this, filtering procedures that rely on vehicles crossing the Y_T asymptote to be considered valid will work better.
- Configure the radar in such a way that vehicles are not “dropped” until an entire cycle has been completed. For example, if the signal cycle is 120 seconds, the device should not be configured to drop a vehicle from tracking if it “lost” the vehicle until 120 seconds have elapsed.
- Select a value of the Y_T asymptote that is as close to the physical area of the intersection as possible. This will make it possible to reduce the number of vehicles that are not counted by the radar during high volume periods since the vehicle is dropped upstream of Y_T and is only picked up again downstream of Y_T. See Appendix B for an example of the scenario described.

4.3.2 Software Implementation

When implementing a version of the trajectory classification algorithm, the following are the recommendations made to the commercialization partner. The recommendations are based on an analysis of vehicle trajectory data, intersection video, and research results described in this report.

- Make changes to the data collection and summary procedures to implement the filtering techniques described in Chapter 2. Implement the changes in the data collection module written in the Python programming language and focus on implementing the procedures associated with the removal of trajectories that do not cross the horizontal asymptote (Y_T).
- Expand the trajectory summary procedures to include additional information about each trajectory such as the change in horizontal position downstream of the Y_T as well as time at which the Y_T is crossed.
- Expose lane information to analysis scripts that execute client-side. This change will eliminate the need to add statistical methods such as the k-means clustering technique described in Chapter 3 and will simplify the implementation of the classification algorithm.
- Implement a modified version of classification procedures developed in the Type 1 project to determine the movement associated with a trajectory. Technical details of the recommended version of the algorithm are described in Appendix C.
- Expose raw vehicle trajectory data to client-side scripts to facilitate future expansion, support research tasks, and prevent users from being “locked” into an analysis interface.

CHAPTER 5. CONCLUSIONS

This report documented procedures that can be used to improve the quality of results produced by an algorithm that can classify vehicle trajectories into movements based on data characteristics downstream of the stop bar of an intersection approach. After a review of the results and limitations of the previous work, a decision was made that the most beneficial approach to improving the quality of results of the previously developed algorithm was to improve the quality of the data used as input. This required the development of a new filtering procedure focused on eliminating noise in vehicle trajectory datasets obtained from a radar-based vehicle detection system as well as the identification of techniques that can be used to classify vehicles into lanes if lane configuration is not available as input for the data analysis procedures.

Details of the filtering techniques and lane classification procedures developed, along with the results, are described in Chapters 2 and 3. Based on advice from the research team as well as their own experience, the commercialization partner created a data collection device that can be deployed at signalized intersections. Early versions of the device were tested in cities across 3 states (Ames, IA; Appleton, WI; and, Bloomington, IL). An initial version of the device (focused on volume data collection and supplemental performance measures such as speed) was released as a product in the Summer of 2019. The released device implements several recommendations from the research team such as the ones described in sections 4.3.1 and 4.3.2 thus making it possible to push a streamlined version of the classification algorithm (see Appendix C for streamlined classification rules) as a software update to the device. In addition to multiple test deployments at cities, an iteration of the device has been commercialized as a research tool and purchased by the University of Massachusetts – Amherst and by the University of Louisville.

5.1 ACCURACY AND PERFORMANCE OF NEW PROCEDURES

Performance of the currently deployed data collection procedures, measured in terms of vehicle volume accuracy, is comparable to the performance of the classification procedures developed as part of the Type 1 project; however, accuracy values achievable by the new procedures are more reliable. The higher level of reliability in the performance of the filtering procedures presented (and thus the streamlined classification procedure described in Appendix C) is due to the resulting trajectory datasets more accurately representing the reality of the combined vehicle volume at an intersection. Previous filtering techniques were not able to eliminate noise at the same level as the techniques described in this report due to a lack of understanding of the type of noise generated; therefore, a comparison of accuracy values alone does not offer the same level of reliability because the noise in the data could have impacted the values. For example, a ghost trajectory such as the ones described in Chapter 2 could have a misleading but positive impact on accuracy calculations by “offsetting” a vehicle trajectory missed by the detection system. This type of scenario explains the focus of the research team on eliminating noise and more accurately representing the vehicle volume on an intersection approach as a key step to improving results produced by the classification algorithm developed in the Type 1 project and of which a streamlined version is described in Appendix C.

5.2 DIFFERENCE BETWEEN COMMERCIALIZATION AND RESEARCH VERSIONS

Data analysis procedures presented were implemented using research tools and are meant to provide the highest level of accuracy possible. The research team recognizes that not all analysis procedures described will be implemented in the product that will be commercialized by the commercialization partner. Commercialization of an analysis procedure must meet thresholds that include the ability to support the procedures in the future from a technical and customer support point of view. The commercialization decision also includes an exercise that must balance the complexity of an analysis procedure against the benefits obtained. For example, based on the results of the research, changes were made to the data summary procedures to rely on the position of the Y_T asymptote. These changes were implemented in the upcoming commercial version because it can generate a vehicle trajectory dataset that can be used as input for monitoring multiple performance measures, including the generation of turning movement count data.

The research described also demonstrated that vehicle trajectory data can be classified into lanes without the need to define lane boundaries. This approach, which relies on clustering techniques can be useful for research applications and was initially considered as the next step in improving the classification algorithm, but the complexities and supporting such an automated procedure outweigh the commercial benefits. However, since the addition of lane information can improve the trajectory classification process by eliminating some of the automated filtering techniques developed as part of the Type 1 project, changes to the configuration interface were made to allow entering lane boundaries. These lane boundaries entered by the user make it possible to streamline the procedures developed as part of the Type 1 project while still maintaining the same underlying classification procedures based primarily on the X coordinate change downstream of the Y_T asymptote.

5.3 FUTURE RESEARCH AND APPLICATION OPPORTUNITIES

The type of trajectory data obtained from the data collection system, which is cleaned using the noise removal procedures described in Chapter 2, and that can be supplemented by the lane classification procedures described in Chapter 3, can be used to gain insights about the performance of an intersection that are not possible with traditional monitoring techniques. For example, safety-related performance measures such as time-to-collision between vehicles could be obtained by analyzing the speed and position of vehicles while using the intersection approach and using their speed and distance to calculate the time-to-collision safety performance measure. Vehicle trajectories from multiple approaches can be analyzed together to have a better understanding of how left turn and thru vehicles interact during permissive phases; something that would be possible since using the trajectories from individual approach, the time when vehicles arrived at a common conflict point can be identified and used in the calculations.

Furthermore, previous research [7] has demonstrated the feasibility of monitoring red light running from radar-based vehicle trajectories thus opening the doors to using the data collection system not only for volume monitoring but also for quantifying the number of red-light runners per intersection, a proactive safety monitoring technique. As a research tool, techniques and classification procedures open the door to understanding detailed vehicle interactions between vehicles and vulnerable road users such as pedestrians by allowing researchers to monitor the change in the speed of vehicles as they approach a conflict point.

CHAPTER 6. REFERENCES

[1] NTOC, “2012 National Traffic Signal Report Card: Technical Report,” Washington, D.C., Technical Report, 2012.

[2] NTOC, “2007 National Traffic Signal Report Card: Technical Report,” Washington, D.C., Technical Report, 2007.

[3] K. Santiago-Chaparro, M. V. Chitturi, A. R. Bill, and D. A. Noyce, “Automated Turning Movement Counts for Shared Lanes: Using Existing Vehicle Detection Infrastructure,” Final Report NCHRP IDEA Project 177, 2016.

[4] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications*. Society for Industrial and Applied Mathematics, 2007.

[5] J. A. Hartigan and M. A. Wong, “Algorithm AS 136: A k-means clustering algorithm,” *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[6] R Core Team, *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing, 2015.

[7] K. Santiago-Chaparro, M. Chitturi, A. Bill, and D. Noyce, “Expanding the capabilities of existing vehicle detection infrastructure to monitor red light running,” *International Journal of Engineering Management and Economics*, vol. 5, pp. 196–208, 2015.

APPENDIX A: TOOLS TO SUPPORT ANALYSIS AND RECOMMENDATIONS

Across the report, statements are made indicating that recommendations made take into account the experience of the research team with collecting vehicle trajectory data. The experience includes fieldwork as well as time spent analyzing video and matching the video to vehicle trajectory datasets. Matching video to trajectory datasets is a time-consuming process but invaluable as it provides qualitative insights into a dataset based on quantitative data. A software tool written in MATLAB was created to combine video with vehicle trajectory data and to simplify the trajectory-matching process. Figure 24 ahead shows a screenshot of the two aforementioned datasets merged into one. On the left side of the figure video from the intersection is shown, while on the right side, vehicle trajectory data points for the corresponding time on the video are also displayed along with the corresponding vehicle identifiers. In addition to quantitative data, this type of merged datasets is what allows the research team to make observations about the impact of settings on the device on the quality of the trajectory data available for analysis.

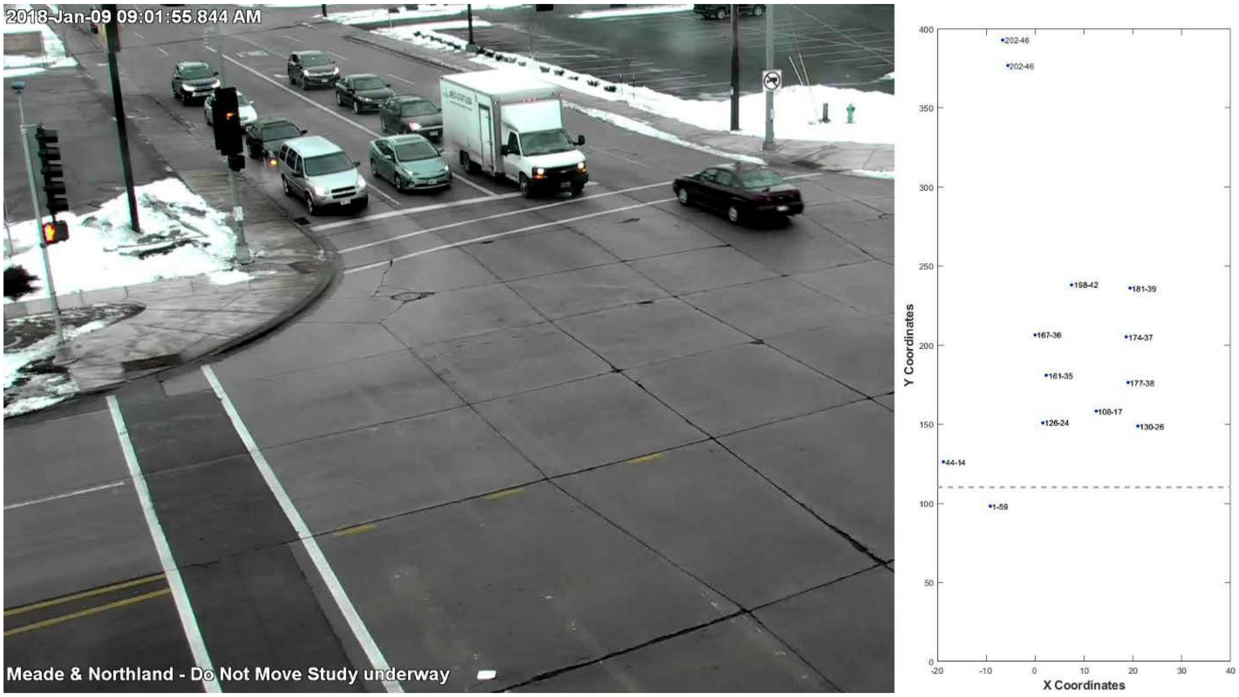


Figure 24. Results of Merging Vehicle Trajectory Data and Video

Using the datasets merging tool, the research team was able to visually confirm the moment that vehicles cross the Y_T asymptote used to filter trajectory data and eliminate noise. The tool also makes it possible to take a look at points in time when mishaps in the data collection process happen and to better understand those failures. Examples of these mishaps include vehicles prematurely dropped by the radar sensors, instances of sensor confusion, and instances of ghost trajectories among others. The merge tool also provides a simple interface to validate results and to match vehicles in the video with the corresponding vehicle identifier in the trajectories dataset. For example, in addition to the scenario shown in Figure 24, the scenario shown in Figure 25 demonstrates the use of the dataset merging tool to observe additional information about a vehicle such as the maximum speed observed by the radar sensor.

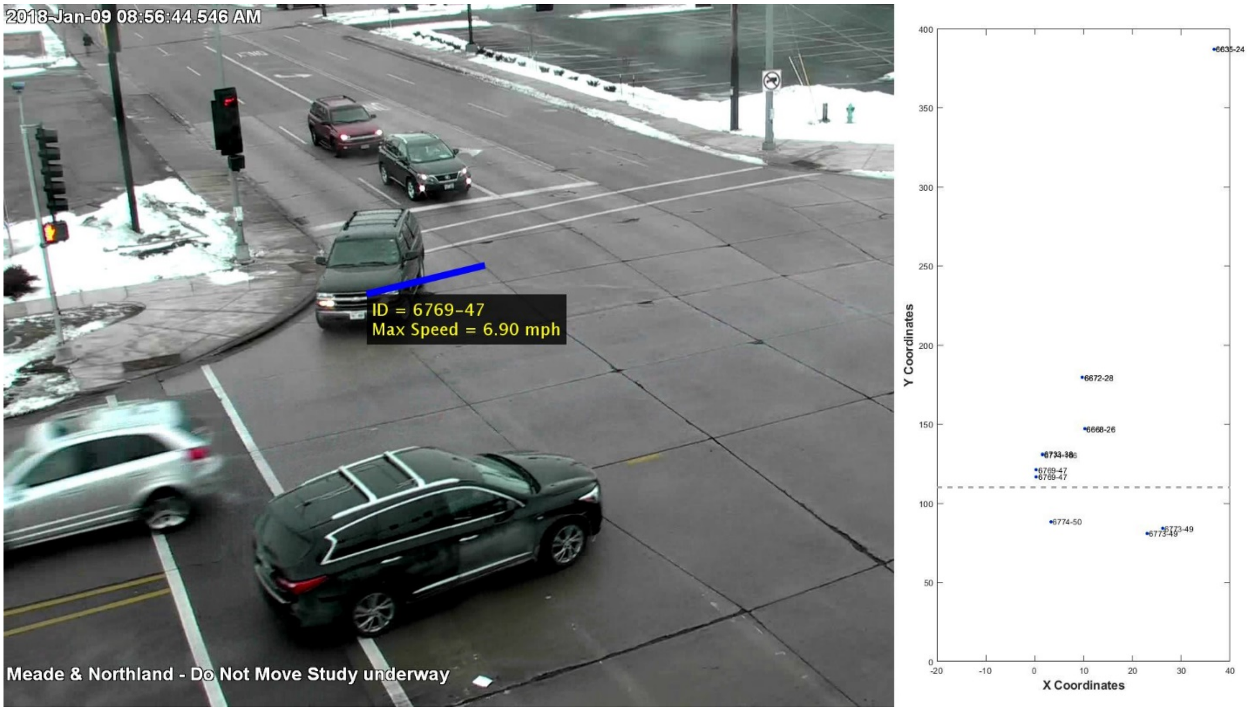


Figure 25. Example of Additional Information for Vehicle

APPENDIX B: SUPPLEMENTAL VISUALIZATIONS

Figure 26 ahead shows an expanded version of the typical trajectory dataset used throughout the research and mentioned in the report. The dataset is for the northbound approach of the Meade and Northland intersection in Appleton, WI. The red line shown in the dataset represents all the data available for a specific vehicle identifier and demonstrates the importance of using a Y_T value as low as possible. For example, if the asymptote represented by the upper dashed line was used to filter valid vehicles, then the trajectory shown in red would not have been considered a valid observation while the use of the lower dashed line, on the other hand, would have included the vehicle as a valid observation.

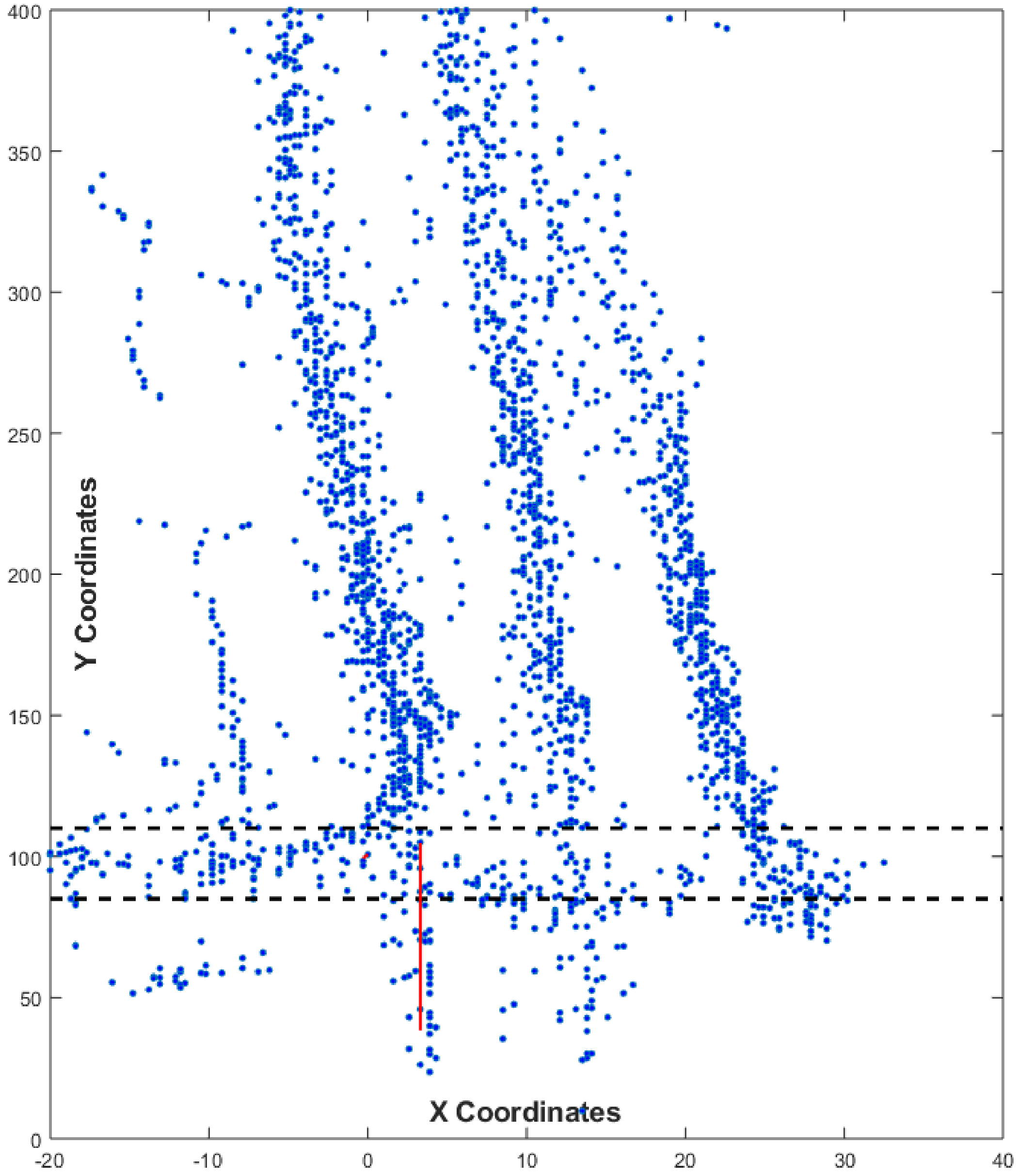


Figure 26. Expanded Visualization of Approach Trajectory Data

APPENDIX C: TECHNICAL NOTES ON IMPROVED CLASSIFICATION ALGORITHM

The technical details of the streamlined version of the classification algorithm mentioned throughout the report are described ahead. The algorithm relies on data generated by following the filtering procedures described in this report, which can be obtained from the datasets generated by the data collection system commercialized by the commercialization partner. The algorithm also relies on lane information assigned by taking into accounts lane boundaries that can be defined in the data collection system commercialized by the commercialization partner.

Input Data

Data used by the classification algorithm are those vehicles that after undergoing the filtering procedures described have been found to cross the Y Cutoff point shown in Figure 27 between the xMin and xMax locations identified in the same figure. As previously discussed, the Y Cutoff point is a horizontal asymptote located between the approach stop bar and the edge of the intersection conflict area. Each vehicle trajectory found to satisfy the described conditions is assigned a lane based on the X Coordinate at the Y Cutoff point (X_{YT}). Furthermore, for each vehicle trajectory, the final X coordinate (X_F) is known as well as the total change in X coordinate downstream of Y Cutoff (D_X) and the total change in Y coordinate downstream of Y Cutoff (D_Y).

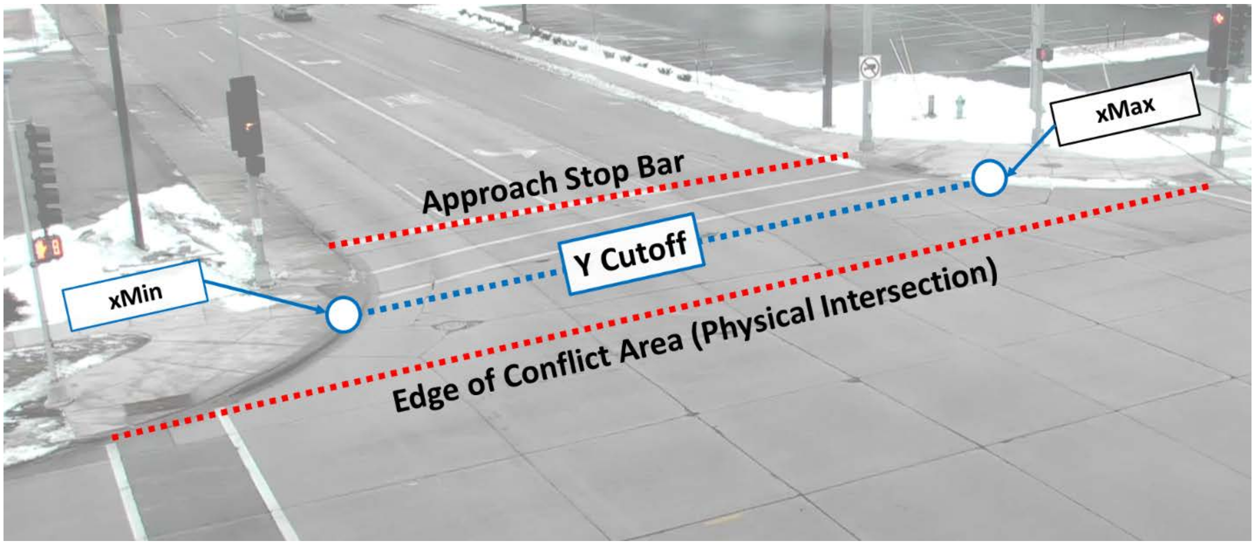


Figure 27. Supporting Figure for Parameter Description

Classification Procedure

Based on the information available for each trajectory, the rules listed ahead can be implemented to assign a movement for each vehicle trajectory part of a trajectory dataset. Once a movement is assigned to a trajectory, no other rules are applied to a trajectory. The rules are applied in the order presented. For implementation purposes, X_{T1} represents the leftmost boundary of the lanes on which thru movements are permitted while X_{T2} represents the rightmost boundary. Also, X_1 and X_2 represent the leftmost and rightmost boundaries of all lanes on the approach used by vehicles. The number of lanes that allow thru movements is referred to as N_T. An assumption is made that right, thru, and left movements are possible.

Rule: $X_F < X_1$.
Decision: Classify trajectory as a right turn.

Rule: $X_F > X_2$.
Decision: Classify trajectory as a left turn.

Rule: $X_F \geq X_1$ and $X_1 \leq X_2$.
Decision: Classify trajectory as thru.

Rule: $X_F < X_{T1}$ and $D_Y < D_X$.
Decision: Classify trajectory as a right turn.

Rule: $X_F < X_{T1}$ and $D_X < (X_1 – X_2) / 2^*N_T$ [Only test if vehicle on lane that allows right turns]
Decision: Classify trajectory as right turn.

Rule: $X_F > X_{T2}$ and $D_X > (X_2 – X_1) / 2^*N_T$ [Only test if vehicle on lane that allows right turns]
Decision: Classify trajectory as right turn.

Rule: $D_Y > D_X$ [Only test if the vehicle is on a lane that allows thru movements]
Decision: Classify trajectory as thru.

Rule: $X_F > X_{T1}$ [Only test if the vehicle is on a lane that allows right turns]
Decision: Classify trajectory as thru.

Rule: $X_F < X_{T2}$ [Only test if the vehicle is on a lane that allows left turns]
Decision: Classify trajectory as thru.

Rule: All remaining vehicle trajectories
Decision: Assign the movement based on the most common movement on the lane.

Performance Evaluation

The performance of the streamlined version of the algorithm was evaluated on a dataset similar to that used for the Type 1 project from the Wisconsin Ave and North Meade Street intersection in Appleton, WI. Ground truth 15-minute volume by vehicle movement was compared with the equivalent volume reported by the algorithm. A total of 108 intervals were analyzed. In the analysis dataset, each analysis interval was associated with a unique vehicle movement and 15-minute time period. The result of the comparison is shown in Figure 28. As the results show, in 64.8% of the intervals, the difference between the ground truth volume and the volume reported by the algorithm had a ±2 vehicles accuracy.

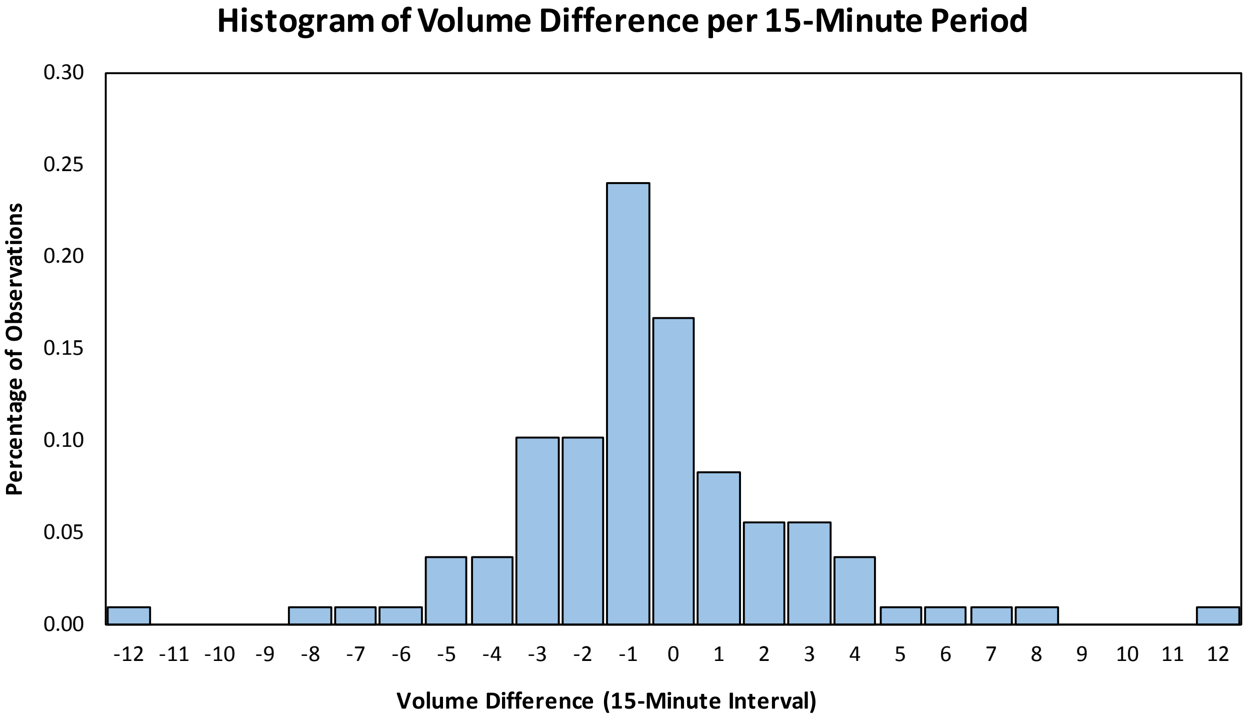


Figure 28. Side by Side Comparison of Ground Truth Volume and Algorithm Results

Further analysis of the algorithm performance reveals that on average the difference between volume reported by the algorithm and the ground truth volume is -0.53 vehicles, an average absolute error of 2.25 vehicles, and an average absolute error (expressed as a percentage) is equal to 9.48%. These performance results are similar to those obtained in the previous version of the algorithm and were obtained with simpler procedures made possible by improved filtering techniques and which can be commercialized without having to rely on complex scientific software tools.

Research Results

Turning Movement Counts on Shared Lanes

Improved algorithm to obtain turning movement counts from radar-based vehicle detection systems and helped create a prototype data collection system.

Program Steering Committee:
NCHRP IDEA Program Committee

AUGUST 2019

Title:
Turning Movement Counts on Shared Lanes: Prototype Development and Analysis Procedures

Project Number:
NCHRP IDEA 198

Start Date:
March 6, 2017

Completion Date:
May 31, 2019

Principal Investigator:
David A. Noyce, Professor
danoyce@wisc.edu
608-265-1882

WHAT WAS NEEDED?

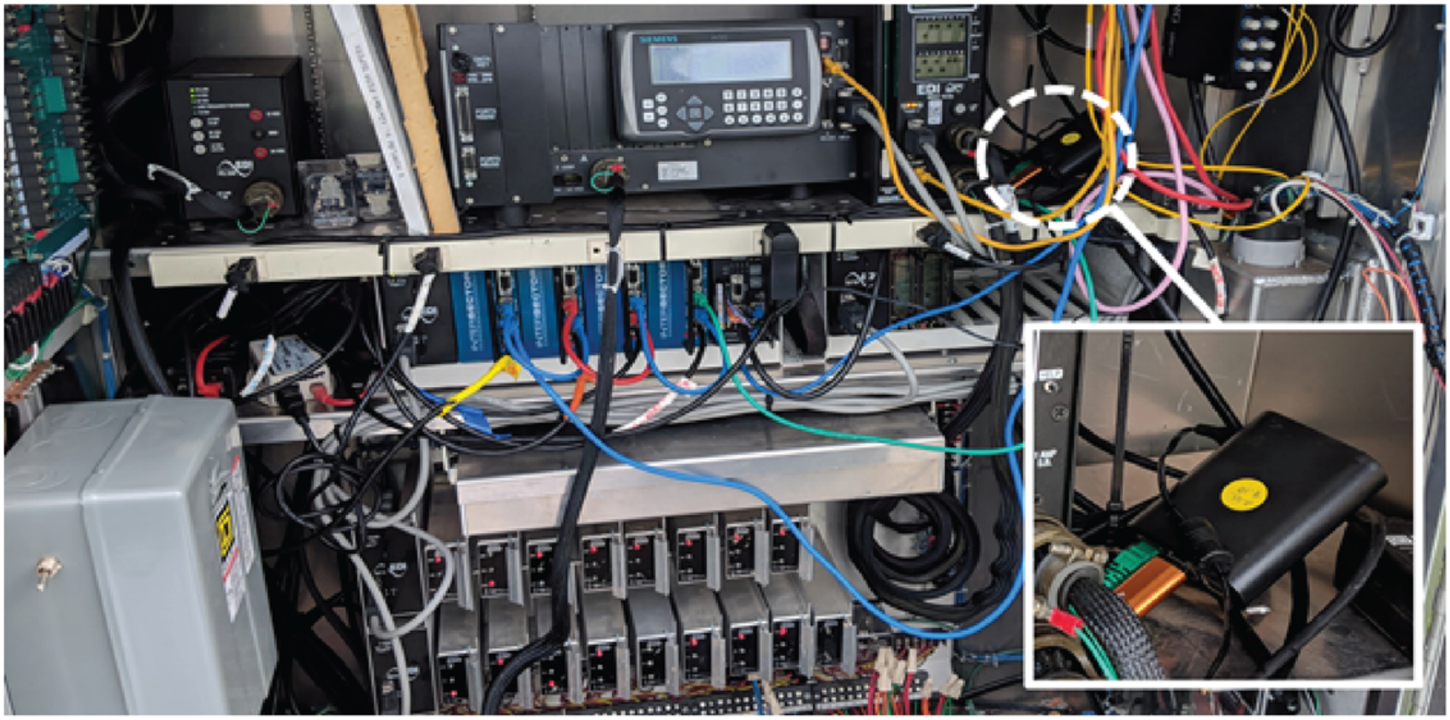
Turning movement counts are a crucial component of data-driven decision-making processes used by transportation agencies. Unfortunately, having quality turning movement count data is not the norm for transportation agencies. To this day, and even as technology advances, some of the current methods that are used to collect turning movement count data, and that are considered standard practice, are still based on manual procedures. Furthermore, automated procedures that rely on data from vehicle detection systems provide limited insights into the performance of an intersection since the procedures often report data by emulating the behavior of inductive loop detectors and are unable to break vehicle volume by movement on shared lanes.

Typical Installation of Radar-Based Vehicle Detection System on Signalized Intersection

As the use of radar-based vehicle detection at intersections continues to grow as an alternative to loop detectors, opportunities for a data collection system that can log vehicle trajectories and obtain advanced performance measures emerges. Such a system was developed by the research team (in the form of a research prototype) and was used to demonstrate that based on vehicle trajectory data, continuous turning movement counts can be conducted at signalized intersections. The demonstration was part of a Type 1 IDEA project. However, prior to attempting to move forward with a commercialization effort, there was a need to improve the quality of the data generated by the device and to streamline the vehicle trajectory analysis procedures.

WHAT WAS OUR GOAL?

The primary goal of the research project was to improve the quality of the dataset used by a previously developed classification algorithm to assign a turning movement to vehicle trajectories regardless of the lane configurations of the intersection approach. It was determined that the best approach to improve the quality of the data was to improve the noise removal procedures applied to vehicle trajectory datasets and to generate vehicle summaries that help streamline the data analysis procedures.


Prototype Device Deployed by Commercialization Partner at Signalized Intersection | Appleton, WI

WHAT DID WE DO?

By working with our commercialization partner, data collection and analysis procedures were developed to generate accurate vehicle trajectory datasets that describe the path of vehicles that use a signalized intersection approach instrumented with a radar-based vehicle detection system. The improved accuracy of the trajectory dataset is the result of noise removal procedures that were developed by analyzing the characteristics of vehicle trajectory datasets as well as supplemental work by the commercialization partner to develop a configuration interface for the data collection system that supports deploying streamlined versions of performance monitoring procedures. As part of the development of procedures for removing noise in trajectory datasets, a characterization of typical noise found in the dataset was also completed. By using clustering algorithms, procedures were also developed to automate the classification of vehicle trajectories into lanes when no definition of lane boundaries is available.

WHAT WAS THE OUTCOME?

The outcome of the Type 2 research project is a data collection system and data analysis platform that can be deployed at signalized intersections instrumented with a radar-based vehicle detection system that acts as an alternative to loop detectors. The data collection system and analysis platform make it possible to obtain performance measures from signalized intersections that rely on a commercially available radar-based vehicle detection system without having to consider the signal controller platform or to interfere with the typical operation of a signalized intersection. Because of architecture decisions made by the commercialization partner and the research team, the data collection can be expanded in the future to support data analysis procedures that are not limited to the generation of traditional performance measures such as volume and speed.

WHAT IS THE BENEFIT?

By improving data collection practices, better signal timing and safety monitoring procedures can be applied to signalized intersections, thus contributing to a better operation of the transportation network. Furthermore, the data collection system commercialized, and that implements procedures developed as part of this research project, can be expanded to support advanced safety evaluations, thus contributing to much needed proactive safety monitoring practices across the transportation network. Proactive safety management could allow agencies to better understand the impact of safety treatments without having to wait for crash history data to become available. For example, using the type of data provided by the system commercialized, engineers could conduct detailed analysis of the speed of vehicles, deceleration rates, acceleration rates, and compliance with traffic laws. These detailed analyses could then be used to quantify the behavioral impact of treatments without the need for the aforementioned crash history dataset.

LEARN MORE:

Visit www.topslab.wisc.edu to obtain the full research report.