

# JOINT TRANSPORTATION RESEARCH PROGRAM

INDIANA DEPARTMENT OF TRANSPORTATION  
AND PURDUE UNIVERSITY



## Guidelines for Evaluating Safety Using Traffic Encounters: Proactive Crash Estimation on Roadways with Conventional and Autonomous Vehicle Scenarios



**Andrew P. Tarko, Mario A. Romero,  
Vamsi Krishna Bandaru, Xueqian Shi**

## RECOMMENDED CITATION

Tarko, A. P., Romero, M. A., Bandaru, V. K., & Shi, X. (2023). *Guidelines for evaluating safety using traffic encounters: Proactive crash estimation on roadways with conventional and autonomous vehicle scenarios* (Joint Transportation Research Program Publication No. FHWA/IN/JTRP-2023/02). West Lafayette, IN: Purdue University. <https://doi.org/10.5703/1288284317587>

## AUTHORS

### **Andrew P. Tarko, PhD**

Professor of Civil Engineering and Director of Center for Road Safety  
Lyles School of Civil Engineering  
(765) 494-5027  
tarko@purdue.edu  
*Corresponding Author*

### **Mario A. Romero, PhD**

Research Scientist and Program Manager for the Center for Road Safety  
Purdue University

### **Vamsi Krishna Bandaru**

Graduate Research Assistant  
Lyles School of Civil Engineering  
Purdue University

### **Xueqian Shi**

Graduate Research Assistant  
Lyles School of Civil Engineering  
Purdue University

## JOINT TRANSPORTATION RESEARCH PROGRAM

The Joint Transportation Research Program serves as a vehicle for INDOT collaboration with higher education institutions and industry in Indiana to facilitate innovation that results in continuous improvement in the planning, design, construction, operation, management and economic efficiency of the Indiana transportation infrastructure. [https://engineering.purdue.edu/JTRP/index\\_html](https://engineering.purdue.edu/JTRP/index_html)

Published reports of the Joint Transportation Research Program are available at <http://docs.lib.purdue.edu/jtrp/>.

## NOTICE

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views and policies of the Indiana Department of Transportation or the Federal Highway Administration. The report does not constitute a standard, specification or regulation.

## TECHNICAL REPORT DOCUMENTATION PAGE

<b>1. Report No.</b> FHWA/IN/JTRP-2023/02	<b>2. Government Accession No.</b>	<b>3. Recipient's Catalog No.</b>	
<b>4. Title and Subtitle</b> Guidelines for Evaluating Safety Using Traffic Encounters: Proactive Crash Estimation on Roadways with Conventional and Autonomous Vehicle Scenarios	<b>5. Report Date</b> January 2023		<b>6. Performing Organization Code</b>
	<b>7. Author(s)</b> Andrew P. Tarko, Mario A. Romero, Vamsi Krishna Bandaru, and Xueqian Shi		
<b>9. Performing Organization Name and Address</b> Joint Transportation Research Program Hall for Discovery and Learning Research (DLR), Suite 204 207 S. Martin Jischke Drive West Lafayette, IN 47907	<b>8. Performing Organization Report No.</b> FHWA/IN/JTRP-2023/02		<b>10. Work Unit No.</b>
	<b>11. Contract or Grant No.</b> SPR-4439		
<b>12. Sponsoring Agency Name and Address</b> Indiana Department of Transportation (SPR) State Office Building 100 North Senate Avenue Indianapolis, IN 46204	<b>13. Type of Report and Period Covered</b> Final Report		
	<b>14. Sponsoring Agency Code</b>		
<b>15. Supplementary Notes</b> Conducted in cooperation with the U.S. Department of Transportation, Federal Highway Administration.			
<b>16. Abstract</b> <p>With the expected arrival of autonomous vehicles, and the ever-increasing levels of automation in today's human driven vehicles, road safety is changing at a rapid pace. This project aimed to address the need for an efficient and rapid method of safety evaluation and countermeasure identification via traffic encounters, specifically traffic conflicts that are considered useful surrogates of crashes. Recent research-delivered methods for estimating crash frequencies based on these events were observed in the field. In this project we developed a method for observing traffic encounters with two LiDAR-based traffic monitoring units, called TScan, which were recently developed in JTRP-funded projects SPR-3831 and SPR-4102.</p> <p>The TScan units were deployed in the field for several hours to collect data at selected intersections. These large data sets were used to improve object detection and tracking algorithms in order to better assist in detecting traffic encounters and conflicts. Consequently, the software of the TScan trailer-based units was improved and the results generated with the upgraded system include a list of potential encounters for further analysis.</p> <p>We developed an engineering application for analyzing the trajectories of vehicles involved in the pre-selected encounters to identify final traffic encounters and conflicts. Another module of the engineering application visualized the traffic encounters and conflicts to inspect the spatial patterns of these events and to estimate the number of crashes for the observation period. Furthermore, a significant modeling effort resulted in a method of producing factors that expand the conflict-based crash estimates in short observation periods to an entire year. This report provides guidelines for traffic encounters and conflicts, the user manuals for setting up and operating the TScan research unit, and manuals for the engineering applications mentioned above.</p>			
<b>17. Key Words</b> safety, traffic conflicts, encounters, rapid safety analysis, TScan		<b>18. Distribution Statement</b> No restrictions. This document is available through the National Technical Information Service, Springfield, VA 22161.	
<b>19. Security Classif. (of this report)</b> Unclassified	<b>20. Security Classif. (of this page)</b> Unclassified	<b>21. No. of Pages</b> 120 including appendices	<b>22. Price</b>

## EXECUTIVE SUMMARY

### Introduction

With the expected arrival of autonomous vehicles, and the ever-increasing levels of automation in today's human-driven vehicles, road safety is changing at a rapid pace. Therefore, an efficient and rapid method of safety evaluation and countermeasure identification is needed to effectively monitor and analyze safety. Traffic encounters (defined as any event that brings the possibility of a crash) have the potential to be an effective surrogate measure. The two long-standing hurdles of applying traffic encounters to engineering practice—(1) cumbersome field observations, and (2) the lack of proven methods for estimating the crash frequency—are both being resolved through recent or current studies. Two JTRP funded projects, SPR-3831 and SPR-4102, have resulted in the Center for Road Safety developing two LiDAR-based traffic monitoring systems that are capable of recording traffic encounters. Traffic conflicts are severe cases of traffic encounters, and they are associated with a high risk for crashes. The theoretical basis of estimating crash frequency based on traffic conflicts has recently been advanced and published. The preliminary evaluation of the method's ability to estimate the frequency of rear-end collisions is encouraging results. This project made progress toward real world implementation of traffic conflicts-based method using the TScan system.

### Findings

The TScan units were deployed in the field for prolonged periods to identify and track traffic at select intersections. The

data sets were then used to evaluate and improve object detection and tracking algorithms to improve traffic conflict detection results. Guidelines for collecting and interpreting encounters and conflicts and estimating the annual number of crashes are included in this report.

To facilitate application of the guidelines, computer-based engineering tools were developed to analyze trajectories and identify traffic encounters and conflicts. A visualization tool—the encounter diagram builder—presents the spatial distribution of encounters and conflicts in a similar way as the already-existing collision diagram tool. The numerical outcomes produced with the tools include a list of traffic encounters, a subset of encounters confirmed as conflicts, and the estimated annual number of crashes based on traffic conflicts.

### Implementation

The upgraded two TScan trailer-based prototypes are available for use. The guidelines in the appendices introduce the end user to the process of using traffic encounters and conflicts for safety evaluation, and the companion manuals guide users through the process of data collection and processing.

Efficient use of the developed system requires properly training end users. An INDOT-CRS collaborative process was developed to gradually introduce INDOT personnel to using TScan in safety evaluations. This period will also be an opportunity for collecting feedback from the end user and for making limited modifications to the system and documentation.

## CONTENTS

1. INTRODUCTION . . . . .	1
2. TSCAN PROCESS FOR OBSERVING TRAFFIC ENCOUNTERS . . . . .	1
2.1 TScan Units. . . . .	2
2.2 Observing Traffic Encounters . . . . .	2
3. IMPROVEMENTS TO TSCAN ALGORITHM FOR IDENTIFYING TRAFFIC ENCOUNTERS . . . . .	3
3.1 TScan Algorithm Overview . . . . .	3
3.2 Multiple LiDAR Self-Alignment . . . . .	4
3.3 Dimension Estimation . . . . .	4
3.4 Final Smoothing . . . . .	6
4. TRAFFIC ENCOUNTERS. . . . .	6
4.1 Encounters Definition and Importance. . . . .	6
4.2 General Procedure to Identify Traffic Encounters . . . . .	6
4.3 Fast Instantaneous Time to Collision $T$ Computation . . . . .	7
4.4 Identifying Evasive Maneuver . . . . .	7
4.5 Time to Collision $T$ Using Trajectory Replacement . . . . .	7
4.6 Visualizing Traffic Encounters. . . . .	9
5. TRAFFIC CONFLICTS . . . . .	11
5.1 Traffic Conflicts Definition and Importance . . . . .	11
5.2 Identifying Traffic Conflicts . . . . .	11
5.3 Expansion Factors . . . . .	12
6. EVALUATION . . . . .	13
6.1 Evaluation Methodology. . . . .	13
6.2 Data Collected Using TScan Units. . . . .	13
6.3 Crash Data . . . . .	14
6.4 Results Comparison . . . . .	15
6.5 Learnings from Evaluation . . . . .	16
7. FUTURE WORK AND CONCLUSION . . . . .	20
REFERENCES . . . . .	20
APPENDICES	
Appendix A. Guidelines for Traffic Encounters and Conflicts . . . . .	21
Appendix B. TScan Engineering Applications Manual . . . . .	21
Appendix C. TScan User Manual . . . . .	21
Appendix D. Crash Expansion Factors . . . . .	21

## LIST OF FIGURES

<b>Figure 2.1</b> TScan unit	2
<b>Figure 3.1</b> TScan algorithm overview	4
<b>Figure 3.2</b> Self-calibration for alignment of two LiDARs	5
<b>Figure 3.3</b> Orientation estimate of turning vehicle	5
<b>Figure 4.1</b> Calculating $T$ between two vehicles	8
<b>Figure 4.2</b> Estimation of hypothetical collision using the most similar evasion-free trajectory	9
<b>Figure 4.3</b> Traffic encounters detection and visualization	10
<b>Figure 5.1</b> Profile of crash estimates obtained for a sequence of tried $T_t$ thresholds—the identified breakpoint is $T_c$	12
<b>Figure 5.2</b> Examples of a different approach pairs	12
<b>Figure 6.1</b> LiDAR coverage at the intersection of Sagamore Parkway and Cumberland Avenue	14
<b>Figure 6.2</b> LiDAR coverage at the intersection of Sagamore Parkway and Yeager Road	15
<b>Figure 6.3</b> LiDAR coverage at the intersection of Morehouse Road and W 350 N	16
<b>Figure 6.4</b> Crashes vs. encounters at Sagamore Parkway and Cumberland Avenue	17
<b>Figure 6.5</b> Crashes at Sagamore Parkway and Yeager Road (2018, 2019, 2021)	18
<b>Figure 6.6</b> Crashes at Morehouse Road and W 350 N (2018, 2019, 2021)	19

## LIST OF TABLES

<b>Table 2.1</b> TScan-based steps when observing encounters	2
<b>Table 4.1</b> Thresholds used to detect traffic encounters	8
<b>Table 5.1</b> Traffic conflicts criteria	11
<b>Table 5.2</b> Approach pairs at a four-legged intersection	13
<b>Table 6.1</b> Data collected	15
<b>Table 6.2</b> Crash data for selected intersections involving two vehicles	16
<b>Table 6.3</b> Encounters, conflicts, and crashes at observed intersections	19

## 1. INTRODUCTION

The current road safety management method developed for the US users is prescribed in the *Highway Safety Manual* (AASHTO, 2010). It relies on crash records and other data in a period of several years. The method is believed to produce correct results as long as the safety knowledge used to develop the method and represented with safety performance functions and crash modification factors reflect the period analyzed by end users. Updating the method requires analyzing new crash data collected typically in 3 years. From this perspective, the current safety management method is reactive, and the safety estimates are averaged over several years.

Autonomous vehicles are expected to emerge within several years and to substantially reduce the number of crashes. The transition from human-driven to autonomous vehicles poses safety challenges, most of them not yet known. At the same time, road safety will be changing at rates much higher than in the past. The current safety management methods will not keep up with such a rate of changes. An efficient method of safety evaluation and countermeasures identification is needed to effectively monitor and analyze safety during and after the transition period.

After decades of research, traffic encounters (including traffic conflicts or near-crash events) are believed to provide the means of evaluating safety timely as they are much more frequent than crashes and, in the case of traffic conflicts, are clearly connected with crashes. Furthermore, non-conflict encounters provide a more adequate measure of exposure to crashes than traffic volumes. There have been two well-recognized hurdles of applying encounters and conflicts to engineering practice: (1) cumbersome field observations, and (2) the lack of a method of estimating corresponding expected crash frequency.

To address the first hurdle, the Purdue Center for Road Safety has developed a van-based LiDAR-based system for traffic data collection at intersections and short road segments. The system was named Traffic Scanner, or TScan, and the development was funded via the JTRP project SPR-3831. In the next JTRP project SPR-4102, two trailer-based prototypes were developed. The primary source of traffic information for the system is a pair of LiDAR units. A video camera was added in the design for inspecting traffic by the end user after data collection to confirm the validity of the obtained numeric data. The trailer-based prototypes were designed to be set up by a single operator and to be sustainable for several days of data collection with only limited supervision. The TScan output files include all motion characteristics of individual vehicles in a convenient format appropriate for engineering studies such as: speed studies, counting turning vehicles, gap acceptance studies, measuring saturation flows, etc. The TScan results include types of vehicles and their dimensions. The positions of the tracked objects are updated 10 times per second.

The second hurdle with identifying traffic conflicts and estimating the corresponding expected crash frequency was addressed in (Tarko, 2018). The preliminary evaluation of the method showed encouraging results (Tarko, 2019) and the method's ability to estimate the expected number of crashes was confirmed for rear-end collisions in (Tarko & Lizarazo, 2021). The method of estimating frequency of crashes from conflicts needs to be confirmed for collision other than rear-end collisions.

There are three objectives of this study:

1. developing techniques and procedures of observing traffic encounters with TScan,
2. expanding traffic encounters observations to all manners of collision, and
3. developing documentation to help collect traffic encounters and to support safety analysis.

The organization of the report is as follows. A brief overview of the TScan system is presented in Chapter 2. The improvements of the TScan tracking and follow-up procedures to aid in detecting traffic encounters are presented in Chapter 3. Chapter 4 introduces to traffic encounters; it describes identification of encounters with TScan-collected data and presents an example visualization of encounters to supplement the crash-based safety analysis. Chapter 5 introduces traffic conflicts, their connection to crashes, the procedure of extracting conflicts from encounters and the method of expanding the conflict-based estimated number of crashes to the annual frequency. Chapter 6 presents evaluation of the connection between the identified traffic encounters and the reported crashes. Chapter 7 discusses the knowledge gathered during the project, the conclusions drawn regarding the method's performance and future work. The appendices present the following components that are outcomes of the presented research:

- A. guidelines for analyzing road safety using traffic conflicts,
- B. TScan engineering applications manual,
- C. TScan user manual, and
- D. crash expansion factors.

## 2. TSCAN PROCESS FOR OBSERVING TRAFFIC ENCOUNTERS

This chapter briefly explains the process to be followed by the end user when using TScan system for observing traffic conflicts. The hardware is introduced first to understand the capabilities of the system. This knowledge is necessary to decide how many units are needed for effective coverage of the intersection and where to position the units at the intersection. Then, each step of the process (as shown in Table 2.1). TScan-based steps when observing encounters Table 2.1) is explained. Steps 1–4 are facilitated with the TScan computer applications and the TScan hardware, while Steps 5 and 6 are executed by applying end-user computer developed in this project. In addition, Steps 3 and



TABLE 2.1  
TScan-based steps when observing encounters

- |                                                    |
|----------------------------------------------------|
| 1. Offsite preparation                             |
| 2. On-site TScan setup                             |
| 3. Data collection and real time processing        |
| 4. Data post processing                            |
| 5. Extracting traffic encounters                   |
| 6. Visualization of encounters for safety analysis |

4 were improved as part of this project to increase the quality of detecting traffic encounters.

### 2.1 TScan Units

Knowledge of the TScan hardware helps the user decide whether a particular intersection should be observed with a single unit or with two units. The primary source of information is two trailer mounted TScan units with their own fields of view (FOV) that

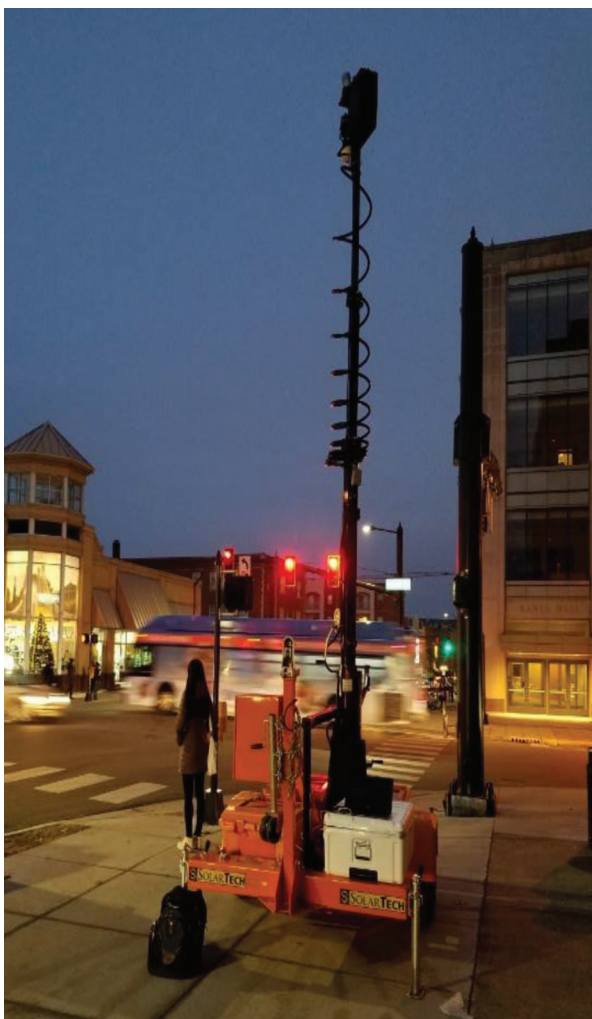
must overlap if both of them are used to obtain a sufficiently large FOV of an intersection.

Furthermore, each TScan unit has two LiDAR sensors mounted on top of a telescoping mast via their own pan-tilt mechanisms to allow their independent directing towards the observed intersection to effectively enable coverage an entire intersection, if one TScan units is used, or part of the intersection if two TScan units are used (Figure 2.1). The telescopic masts have to be raised sufficiently high to reduce the occlusion of cars and parts of an intersection by large vehicles such as trucks.

The effective range of tracking for the TScan system is determined by the LiDAR’s range and scanning pattern. The current TScan units used in the research study had an effective range of 200 ft.

### 2.2 Observing Traffic Encounters

The TScan process includes six phases outlined in Table 2.1—offsite preparation, on-site setup, data



(a) TScan unit collecting data



(b) Head unit containing sensors

Figure 2.1 TScan unit.

collection and real time processing, data post processing, extracting and visualizing traffic encounters for safety analysis.

### 2.2.1 Offsite Preparation

During the offsite preparation, the user defines several intersection area that include approach lanes, intersection exit areas, intersection area, sidewalks, medians, and curb parking lanes. Each area on an intersection ortho-normal image (orthoimage) is marked by drawing with a provided tool. These polygons represent the parts of the intersection that the user is interested in tracking vehicles, pedestrians, and other road users (Appendix C.2). The inputs entered by the user are later provided in the corresponding fields in the TScan output files. Traffic movements and corresponding path through the intersection are defined with their origin and destination polygons. This information about turning movements is used to identify and classify traffic encounters.

### 2.2.2 On-Site Preparation

The onsite process involves the following steps:

1. positioning and leveling the trailer,
2. raising the mast to the desired height,
3. directing the lidar sensors at the intersection for required coverage,
4. mapping the lidar points to the orthoimage,
5. identifying the background,
6. inspecting the sample tracking results, and
7. repeat Steps 2 through 6 if necessary.

Mapping the LiDAR points to the orthoimage is an important step as this affects the tracking quality and the ability to identify encounters. Inaccuracies in this step cases the necessity for repeating Steps 2 through 6. This process has been modified to make it simpler and more intuitive (Appendix C.4.6) thereby reducing certain types of errors. This improvement has provided considerable time saving in setting up the system.

### 2.2.3 Real Time Data Collection and Processing

Once the sample tracking results are deemed satisfactory, the user proceeds to collect data for the desired period of time. The raw point cloud data produced by the LiDAR sensors are processed in real time by a computer included in the TScan head. The resulting trajectories are stored in the TSFF format.

### 2.2.4 Data Post-Processing

The quality of tracking results produced by the real time process was found insufficient for effective identification of traffic encounters. Therefore, a post-processing module was developed to improve the quality of tracking results. This applies also to the

results obtained from data collected with multiple masts and then merged in the post-processing stage.

### 2.2.5 Extracting Traffic Encounters

The post-processing results are an input to the tool developed by the research team. The tool analyzes every pair of vehicles simultaneously present in the FOV to identify and saves the ones that are potentially involved in encounters. Then, the motion characteristics of individual vehicles, the nearness of the potential collisions, and the potential impact speeds are estimated to extract and save the confirmed encounters.

### 2.2.6 Visualization of Encounters for Safety Analysis

The extracted encounters can be visualized using the encounter diagram tool. The user can view the data as disaggregated individual encounters or grouped by pairs of movements as defined by the user.

## 3. IMPROVEMENTS TO TSCAN ALGORITHM FOR IDENTIFYING TRAFFIC ENCOUNTERS

When analyzing data collected in this project, shortcomings of the TScan algorithms that limited the system's ability to identify traffic encounters were discovered. This chapter summarizes these shortcomings of the algorithm and describes subsequent changes made to address these shortcomings. A more detailed description of the TScan algorithms can be found in the SPR-4102 report.

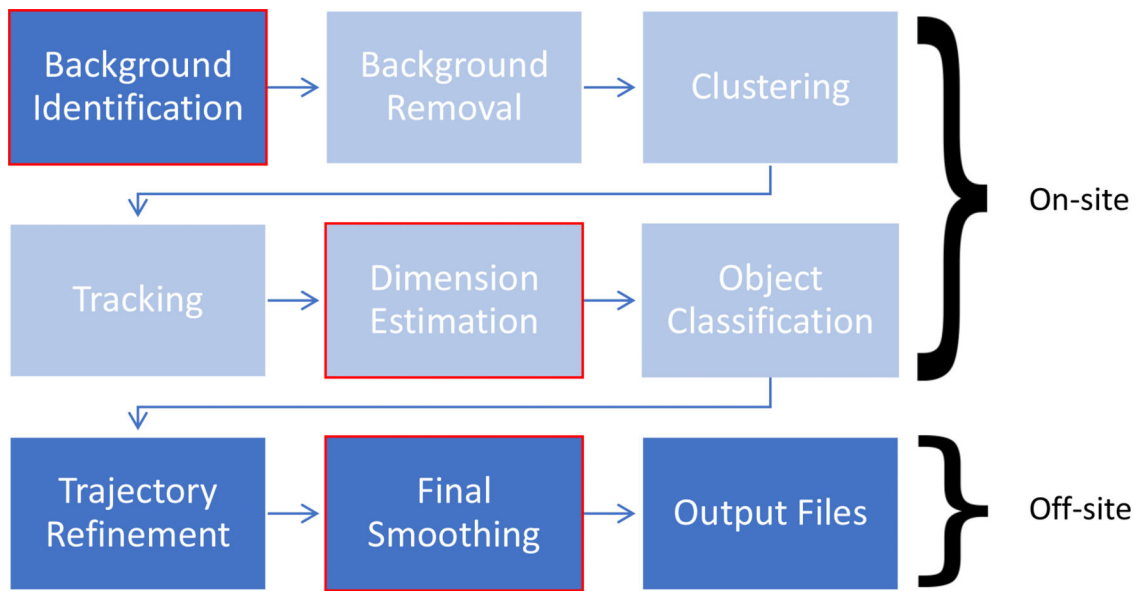
### 3.1 TScan Algorithm Overview

The flowchart of the process is shown in Figure 3.1. Background identification takes place during on-site setup. The boxes in light blue are steps that are executed in real time where bulk of the processing takes place. The quality of real time tracking proved insufficient for applications like traffic encounters. Thus, a post processing module containing the final three steps marked offsite was developed. The boxes marked with a red border indicate the processes that have been improved compared to those in SPR-4102 report.

To provide proper context, a brief explanation of the overall algorithm is provided in the following part of the section.

The raw data from the sensors is converted to 3D points in cartesian coordinates. This process is unique to each sensor model. Then, the points from all the LiDARs are brought to a common reference frame (Section 3.2). Next, a sample of 15 minutes is collected to identify background.

During real time processing, only points assumed to be part of moving objects (foreground) are retained after background removal. The points that belong to the same moving object are grouped together. Objects are then tracked over time.



**Figure 3.1** TScan algorithm overview.

Once an object is finished being tracked, the entire history of the object is used for estimating the dimension of the object. Then a box of the estimated dimension is placed back on the point clouds.

Finally, the class of the object (non-heavy vehicle, heavy vehicle, bicycle, and pedestrian) is identified (Section 3.9 of SPR-3831 (Tarko, Ariyur, et al., 2016)) and the results of real time processing are stored in TSFF format (Appendix C of SPR-4102 (Bandaru et al., 2021)).

Post processing is carried out offline and in the office. If multiple TScan units were used to collect data, then the tracking results from each of them are combined in this phase. The primary purpose of trajectory refinement is to ensure that each cluster ID represents one physical object. This is achieved by iteratively repeating two algorithms until convergence is achieved. The first, identifies and splits up incorrect associations of tracking. The second identifies and merges clusters that belong to the same physical object yet have separate cluster IDs. These are detailed in Section 3.7 of SPR-3831 report (Tarko, Ariyur, et al., 2016). Dimension of the changed clusters are recomputed. A final smoothing operation is performed (Section 3.4) and the output is stored in TSFF format (Section 4.5 of SPR-4102 (Bandaru et al., 2021)).

The subsequent sections explain the improvements made to the algorithm since the SPR-4102 study.

### 3.2 Multiple LiDAR Self-Alignment

This step is part of the background identification process. TScan prototypes have two LiDARs working together to provide enough coverage of the intersection. Each LiDAR has its own internal coordinate system in which it makes measurements. When using multiple LiDARs it is imperative that all the points generated

are brought to one common coordinate system. Section 5.3 in SPR-4102 report describes a novel self-alignment method was developed to achieve the same. That method has been tweaked as to further improve alignment accuracy.

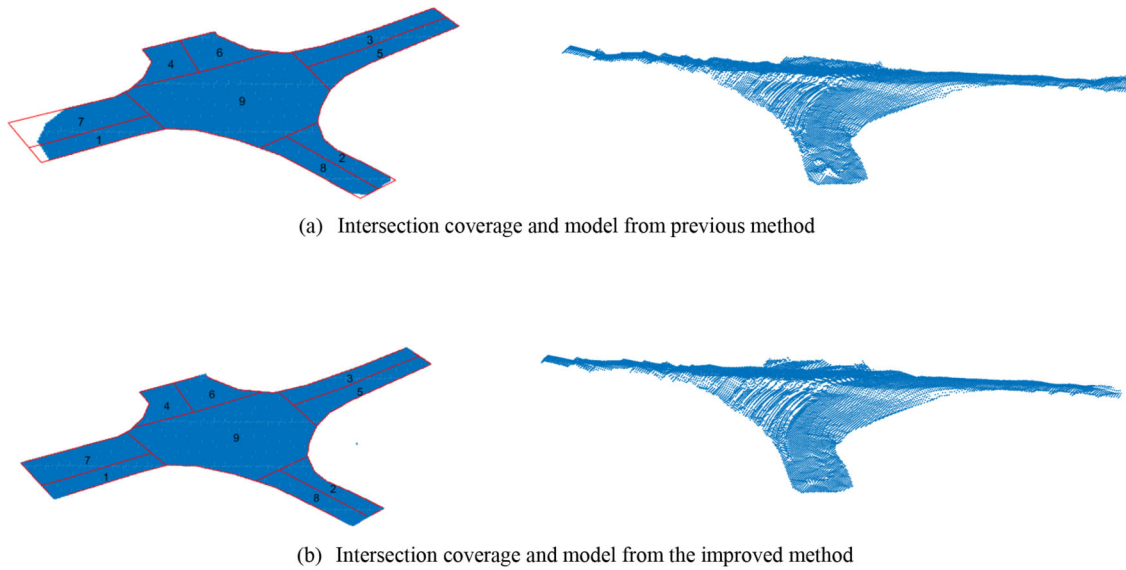
Previously, the self-alignment method looked for common areas that are flat in nature (for instance road pavement) where points from both LiDARs are available. Two planes were then fit individually for each LiDAR's points. Then, the transformation matrix to align the two planes was computed. In practice, there is considerable overlap of the scan lines closer to the mast than farther out. This introduces a bias in the plane fitting process. To overcome this, a Delaunay triangulation is used to interpolate between scan lines. This interpolation is valid for relatively flat surfaces like road pavements. An illustration of the performance of the new method is shown in Figure 3.2.

### 3.3 Dimension Estimation

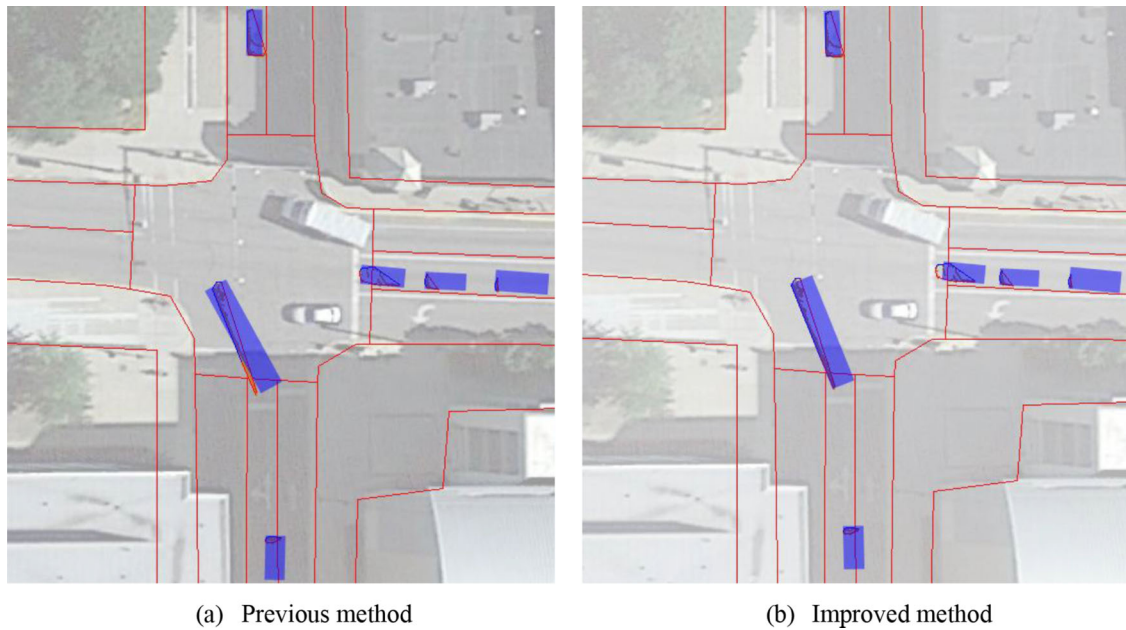
This phase uses the entire history of the vehicle to estimate dimensions and placing the box of the estimated dimension back on the point cloud in each frame. The box orientation estimation is explained next.

#### 3.3.1 Orientation Estimation

The trajectory of the object obtained from the previous step was used to estimate the orientation of the object at each time step. Having a good orientation angle estimate is crucial as this angle is used to transform the object dynamics from the global  $x$ - $y$  coordinate system used in tracking to the body coordinate system of longitudinal and lateral velocity, acceleration etc. that is used in encounters identification. Moreover, side swipe encounters are particularly sensitive to



**Figure 3.2** Self-calibration for alignment of two LiDARs.



**Figure 3.3** Orientation estimate of turning vehicle.

orientation estimation because a small change in orientation could be difference between two vehicles being on a collision course or not.

There are two estimates of the orientation of the object available (see Figure 3.3).

1. Orientation estimate from the trajectory of the object (after MBF smoother is applied, Section 5.7 in SPR-4102 (Tarko et al., 2021)).
2. Orientation estimate from the point cloud spread on the object (MinErrorRect procedure). See Section 3.4.3 in the SPR-3831 report (Tarko et al., 2021).

The orientation angles obtained from the smoothed trajectory did not reflect the reality when a vehicle

was executing a turning maneuver, especially for large vehicles such as busses and trucks. This is because when vehicles execute a turning maneuver, they do so by only turning the front set of wheels. The tangent to the path of the vehicle is not the orientation of the vehicle during a turn. When a vehicle is in the entry or exit lanes, typically it moves in a straight line during this phase, the tangent of the path is indeed the orientation of the vehicle.

The orientation angle obtained from the min-ErrorRect procedure was a better estimate of the orientation of the object when the object was executing a turning maneuver. But this estimate had the following issues.

1. The error of the orientation angle estimate increased with the decrease in number of points that make up the point cloud. This typically happened when the object is far from the LiDAR sensor.
2. The objective of angle estimate was to identify the orientation of the side (edge in 2D representation) of the object that has the highest number of LiDAR points. Hence, this estimate may be off by a multiple of 90 degrees from true orientation of the object depending on which face of vehicle is visible to the LiDAR.

The two-angle estimate methods have complementary strengths and weaknesses. Thus, a method was devised to combine the two estimates to produce a third estimate that did not have any of the weaknesses of the two underlying estimates.

### 3.3.2 Better Treatment of Stopped Objects

The LiDAR sensor has inherent and somewhat troublesome behavior of not hitting the same physical point in space after each rotation. This results in a non-stationary centroid of a stationary object. On top of this, each LiDAR reading has an error standard deviation of  $\pm 5$  cm to  $\pm 10$  cm depending on the distance of the object. To account for this a two-pronged approach was proposed in Section 5.7 in SPR-4102 to identify stopped regions. In the identified stopped region, the position of the object was fixed to a single point in space as is the case in reality. But this fixing, resulted in an artificial motion just before stopping and immediately after the object starts moving. To account for this, a simple five point moving average smoother is used just before and after the stopped region.

### 3.4 Final Smoothing

The Kalman Filter model and the MBF smoother used in Tracking and smoothing is explained in Sections 3.5 and 3.6.1 of SPR-3831 (Tarko, Ariyur, et al., 2016). With jerk (rate of change of acceleration) being needed to identify conflicts, in the final smoothing phase a Kalman Filter model that includes jerk is used. The new model is described below.

The state vector for the Kalman Filter loop is:

$$X_k = (p_x \ v_x \ a_x \ j_x \ p_y \ v_y \ a_y \ j_y)^T \quad (\text{Eq. 3.1})$$

where:

$p_x, v_x, a_x, j_x$  represents the position, velocity, acceleration, and jerk along the x axis, and

$p_y, v_y, a_y, j_y$  represents the position, velocity, acceleration, and jerk along the y axis.

The Kalman Filter assumes that the system evolves from time  $k - 1$  to time  $k$  according to the following equation:

$$X_k = F_k X_{k-1} + w_k \quad (\text{Eq. 3.2})$$

where:

$F_k$  represents the state transition model,

$w_k$  represents process noise which is assumed to be drawn from a zero mean multivariate normal distribution with covariance  $Q_k$ .

At time  $k$  an observation (or measurement)  $z_k$  of the true state  $x_k$  is made according to:

$$z_k = H_k X_k + v_k \quad (\text{Eq. 3.3})$$

where  $H_k$  represents the measurement matrix which maps the values in the state space to the values in the observed space and  $v_k$  presents the observation noise which is assumed to be zero mean Gaussian white noise with covariance  $R_k$ . The error covariance matrix is denoted by  $P_k$ .

Correspondingly, the Modified Bryson-Frazier method (Section 3.6.1 of SPR-3831 (Tarko, Ariyur, et al., 2016) used to smooth the Kalman Filter estimates is updated with the new model.

## 4. TRAFFIC ENCOUNTERS

This chapter defines the concepts of traffic encounters in general and traffic encounters that are conflicts. Then, a method developed and implemented in TScan for extracting traffic encounters is presented in detail.

### 4.1 Encounters Definition and Importance

A traffic encounter is an interaction between two road users that brings a possibility of crash. Interactions caused by aggressive behavior of road users who seem to control the situation also qualify as encounters by definition as the risk of crash is non-zero. Due to a close proximity of other vehicles, any execution error (for example, faulty steering) or perception error (for example, insufficient pavement friction or obstructed sight of another vehicle) may convert the event into a collision.

Thus, a traffic encounter is closer to a collision and more relevant to safety than an unaffected passage of an intersection. Consequently, the volume of traffic encounters is a better measure of exposure to crash than the volume of vehicles entering an intersection. Furthermore, analyzing types of traffic encounters, their distribution in an intersection area, and circumstances under which they occur may help determine causes of these events and prompt corresponding safety countermeasures.

### 4.2 General Procedure to Identify Traffic Encounters

Trajectories for any two road users simultaneously present in the field of view (FOV) must be analyzed to determine if the two users are involved in an encounter between themselves. There may be secondary encounters caused by preceding (primary) encounters. These secondary events are considered independently from the primary ones.

The following conditions must be satisfied to claim an event as a traffic encounter.

- Two road users can be involved in an encounter between themselves only if their paths cross or overlap one another (spatial condition).
- The two road users must be heading towards a collision spot with the potential of arriving at the spot at approximately the same time (temporal condition).
- One of the two users must have performed an evasive action to avoid a potential crash (failure condition) (Tarko, 2020).
- The impact speed of the hypothetical collision must be sufficiently high in order for the event to be safety relevant (severity condition) (Tarko & Lizarazo, 2021).

In general, the following procedure is applied for the identification of traffic encounters.

1. Identify potential pairs in a conflicting situation based on extrapolation of the trajectory for a reasonable period of time using an assumption of constant speed and direction.
2. Among all the detected pairs, find the ones in which at least one vehicle has exhibited a significant change in lateral or longitudinal acceleration (jerk indicating potential evasive action).
3. Find another vehicle free of evasion whose trajectory is close to the trajectory of the vehicle performing evasion.
4. In the pair of vehicles formed in Step 1, replace the evading vehicle with the vehicle free of evasion found in Step 3 and check if the two vehicles would collide.
5. If so, then estimate redefined time-to-collision based on the time between the ending of evasion and hypothetical collision point. Perform Steps 2–5 for all the remaining potential pairs of vehicles in conflicting conditions from Step 1.

The procedure described previously is applicable to all road users in theory. In practice, it is hard to characterize evasive actions and find alternative trajectories in interactions involving pedestrians. So, this study only focuses on vehicle-to-vehicle interactions. The following sections describes each step of the process in detail.

#### 4.3 Fast Instantaneous Time to Collision $T$ Computation

The first step in the process is to identify pairs of objects that whose temporal separation is less than a specific predefined threshold. Given a list of trajectories, instantaneous time to collision is computed between each pair of vehicles present in every time step (“frame”).

Let subscripts  $a$  and  $b$  indicate two vehicles for which instantaneous time to collision needs to be calculated. Assuming that both vehicles do not change their direction of motion and continue to move with the same velocity ( $v_a$  and  $v_b$  respectively), the future position can be predicted using the equation:

$$p'_a = p_a + v_a \times dt \quad (\text{Eq. 4.1})$$

where  $p_a$  is the initial position at a given time instant of vehicle A and  $p'_a$  is the final position predicted after time  $dt$  has elapsed.

A suitably long duration, say 5 seconds, is chosen as the upper limit  $T_e$ . Encounters with  $T$  beyond this value have almost no impact on safety. A box in space between rear of the vehicle in its initial position and the front of the vehicle at the predicted position after time  $T_e$  encompasses all the possible positions the vehicle would be occupying before reaching the predicted position (Figure 4.1b). A similar box is identified for vehicle B as well. If the two boxes intersect, then the vehicles’ paths intersect. Then, the motion of both the objects are simulated with a suitable time resolution to check if they collide with the same assumption of constant velocity. If they do, then the time taken to do so is the required  $T$  value.

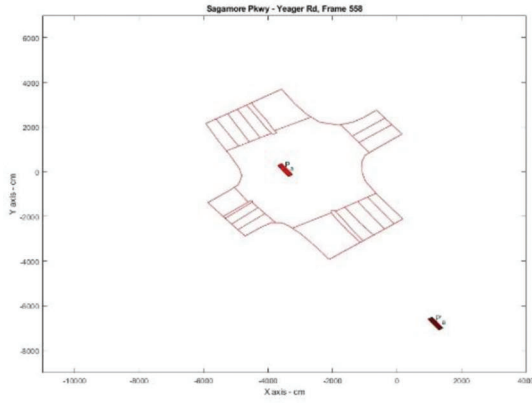
#### 4.4 Identifying Evasive Maneuver

A necessary condition for an event to be classified as an encounter is that at least one of the vehicles in the encounter must have performed an evasive maneuver. If there is no evasive maneuver, then both the drivers perceive no threat and there is no failure. So, the trajectory of each vehicle involved in an encounter is checked to see if at least one vehicle has performed an evasive maneuver. Hard breaking is the most common evasive maneuver taken by drivers (Kiefer et al., 2005; Mollicone et al., 2019). Swerving and hard acceleration are some of the other types of evasive maneuver performed by drivers (Musicant et al., 2010). These maneuvers can be detected by analyzing their jerk profiles. The thresholds used in the identification of evasive maneuvers are shown in Table 4.1.

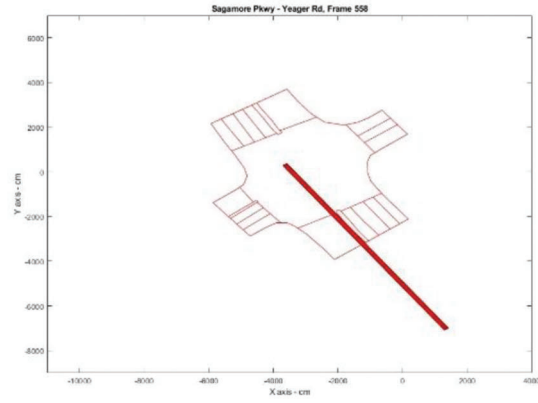
#### 4.5 Time to Collision $T$ Using Trajectory Replacement

To calculate the time to collision  $T$ , the evasive action is removed to simulate the trajectory without a response. This method allows estimation of  $T$  instead of  $T$  since no assumptions are imposed in terms of the constant relative speeds between the road users. This methodology also provides a better characterization of the proximity measure and risk estimation of the encounters.

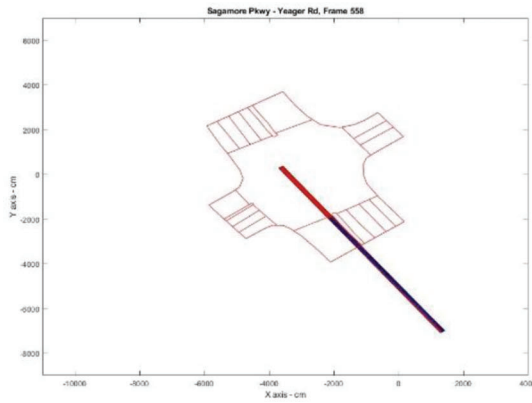
To predict the trajectory adopted by the vehicle in the “what-if” scenario of no evasion, Lefèvre et al. (2014) discussed two possible groups of methods. The first group uses empirical knowledge of dynamics from vehicles to simulate the most probable response; and the second group, which is more appealing in area-wide detection systems, is the characterization of evasion-free trajectories collected from the same location and type of movement (Figure 4.2). Hence, from the partial trajectory before evasion for road user, it is possible to conduct a comparison from a set of evasion-free trajectories  $H = \{H_1, \dots, H_n\}$  with the same origin and destination areas in the intersections to determine the most similar one. Relying on the assumption that the road user will continue along a similar path, the movement of the road user in this “what-if” scenario can be extrapolated, and estimation of the hypothetical



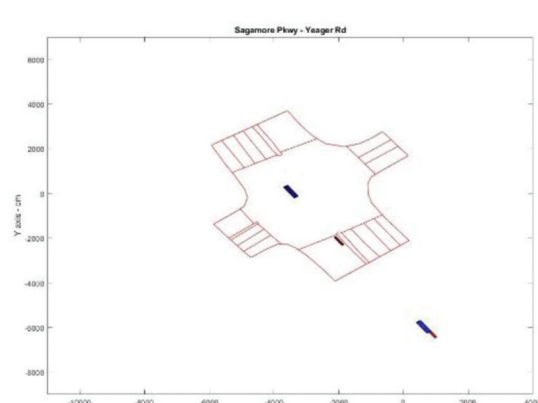
(a) Initial position and position predicted after  $T_e$  seconds



(b) Region of space occupied by the vehicle from  $t = 0$  to  $t = T_e$



(c) Vehicle A and B cross paths within  $T_e$  from current time



(d) Compute precise  $F$

**Figure 4.1** Calculating  $F$  between two vehicles.

**TABLE 4.1**  
**Thresholds used to detect traffic encounters**

Symbol	Encounter Threshold Name	Recommended Value
$B_e$	Minimum longitudinal jerk	10 ft/s <sup>3</sup>
$S_e$	Minimum lateral jerk	6 ft/s <sup>3</sup>
$T_e$	Maximum time to collision	5 s
$\Delta_e$	Minimum impact speed	3 mi/h

collision point can be applied using the simulated trajectory.

The starting point of evasion in time  $t_e$  is defined as follows. It is the first point in time in when the vehicle's jerk (lateral or longitudinal) is above the threshold shown in Table 4.1 following which the instantaneous time to collision ( $T$ ) reaches its minimum value. Thus, the time  $t_e$  and the corresponding cartesian coordinates form the starting point of evasion. It is from this point that an alternate trajectory will be used for calculating actual  $T$ .

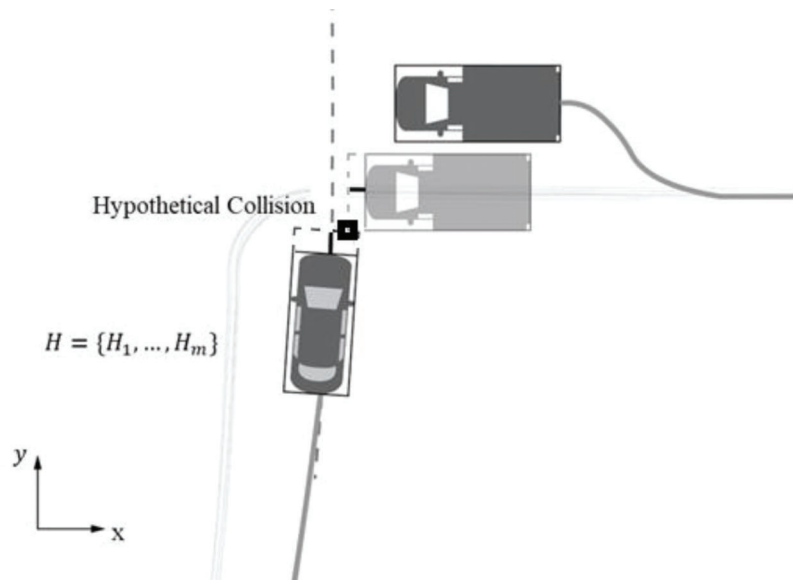
Different similarity metrics are reported in the to obtain the most similar trajectory in  $H$  from the partial

one before evasion. Among the most common similarity metrics, authors have used Euclidean distance, the modified Hausdorff metric (Atev et al., 2010), the longest common subsequence metric (Buzan et al., 2004), and its translation and rotation invariant version of the quaternion-based rotationally Invariant LCS (Hermes et al., 2009).

The Euclidean distance between the points of the trajectories is the most intuitive measure. Consecutive points reported at a constant rate  $1/\delta$  are compared and the average cumulative distance between points is estimated as:

$$R_{ij} = \frac{\sum_{m=1}^K \|X_i - H_j\|^2}{K}, 1 \leq i, j \leq K \quad (\text{Eq. 4.2})$$

where  $K$  the number of centroids compared between trajectories  $X$  and  $H$ . Given the relatively low measurement noise of the sensing system and the accuracy of the tracking algorithm, the above method was used. Additionally, the evasion free trajectories are interpolated in time to account for the fixed scanning pattern of the sensor.



**Figure 4.2** Estimation of hypothetical collision using the most similar evasion-free trajectory.

Once the closest evasion-free trajectory is identified for each vehicle that performs an evasive action, a new trajectory is created by replacing the original trajectory parameters (location, speed, acceleration, and jerk) after the point evasion with the corresponding ones from the closest evasion-free trajectory.

The new trajectories thus obtained along with the bounding box of each vehicle is used to check if the two are on a collision path. If the bounding boxes intersect, then the vehicles are on a collision path and the actual  $T$  is the time between the time of evasion and the time instant when the hypothetical collision happens. If there is no collision detected after trajectory replacement, then the two vehicles were not involved in an encounter.

Finally, if the calculated time-to-collision value  $T$  is lower than the maximum time-to-collision threshold  $T_e$  and the impact speed lower than  $\Delta_e$  (Table 4.1), then the event involving the two vehicles is considered an encounter.

#### 4.6 Visualizing Traffic Encounters

TScan engineering application has a tool for detecting and visualizing traffic encounters. The output files in TSFF format from post processing step of TScan

process containing a list of trajectories serves as input to the tool.

Figure 4.3 shows the user interface of the encounter detection tool. After the TScan results are accessed with the application, the intersection movement O/D matrix (interface Area 1) is created as described in the TScan engineering application manual (Appendix Section B.7.2). Clicking the *Extract* button (Area 3) in the *Encounter* window (Area 2) starts the process of processing each potential encounter event. Only events that meet the encounter thresholds are considered as valid encounters and they are displayed in the *Encounter* window. The results of the analysis include two lists: the list of detected traffic encounters (Area 4), and the list of road user actions (Area 5).

A visualization of the results in the form of encounter diagram (Area 7) is obtained by clicking on the encounters tab (Area 6). The height of the bars in the *Aggregated* mode is proportional to the number of encounters. Both the individual encounters and the bars are color coded by severity depending on the time to collision. Traffic encounters have to be grouped by pair of conflicting paths.

Appendix Section B.11 provides details on how to use the traffic encounters engineering application.



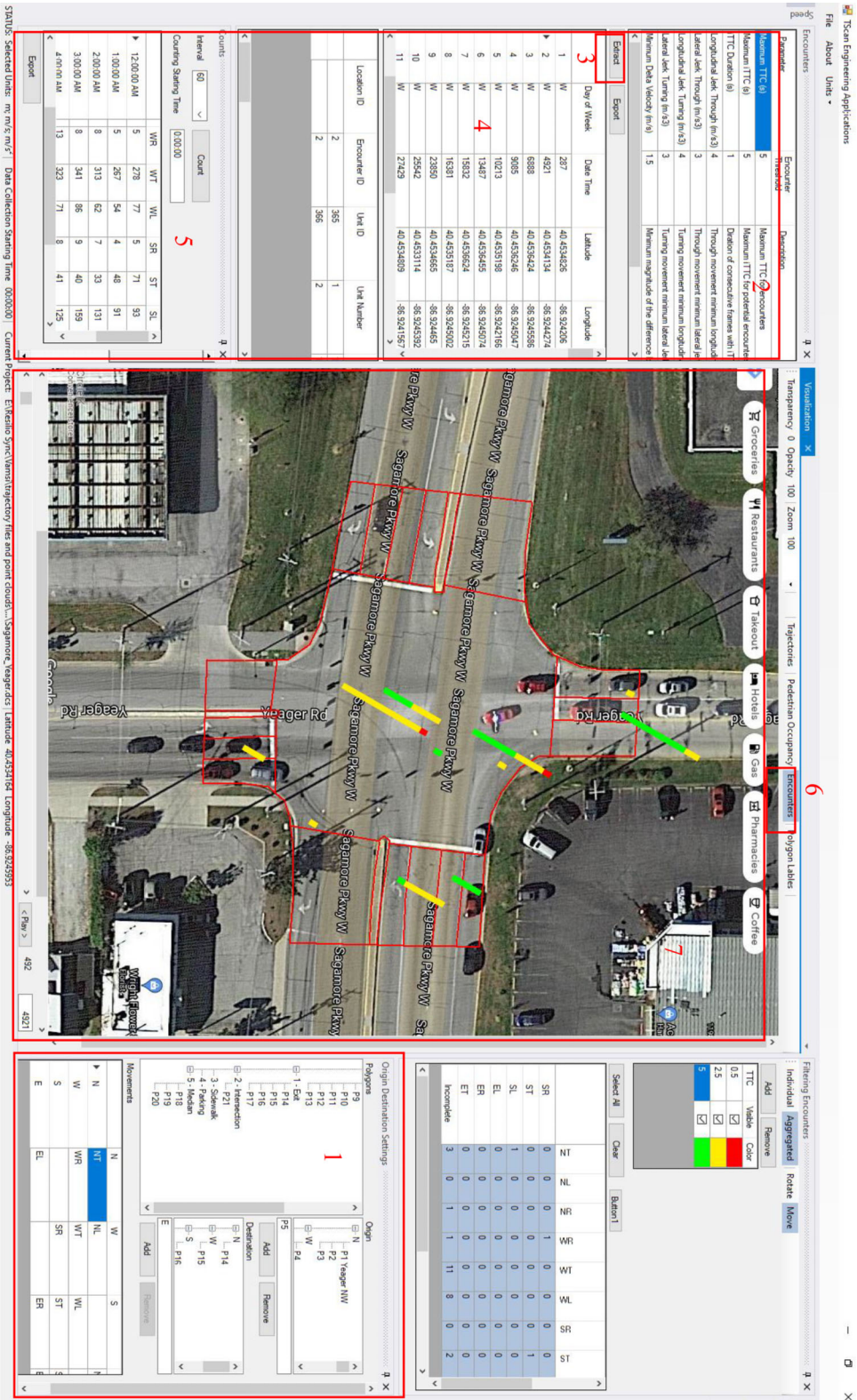


Figure 4.3 Traffic encounters detection and visualization.

## 5. TRAFFIC CONFLICTS

This section provides a definition for traffic conflicts, explains additional steps needed to select conflicts from encounters already identified, and provides a method to convert the observed conflicts to the corresponding number of crashes expected in the observed period.

### 5.1 Traffic Conflicts Definition and Importance

Traffic conflicts are a subset of encounters. They are caused by errors made by road users or other failures such as potholes, traffic signals malfunctioning, or shortage of sight distance. These encounters lead to an unacceptably short times to collision and an uncharacteristically strong reactions from road users to prevent a collision.

The existence of an error makes a traffic conflict etiologically consistent with a crash. It is because traffic conflicts and crashes are both caused by errors. Furthermore, assuming that an error precedes a traffic conflict simplifies its analytical representation because it is easier to observe an error than to predict its occurrence. Estimating the probability of collision given an error is reduced to estimating the probability of a response time longer than the time remaining to a looming collision.

Traffic conflict events are characterized with time to collision  $T$  values that are not acceptable even to aggressive drivers. Thus, the observed evasive maneuvers are stronger than ones performed by aggressive drivers that control the risk. Consequently, conflict threshold  $T_c$  is unacceptably low and shorter than encounter's threshold  $T_e$ , while conflict jerk thresholds  $B_c$  and  $S_c$  are stronger than the corresponding encounter's thresholds.

### 5.2 Identifying Traffic Conflicts

Values of the conflicts criteria must be determined for each group of traffic conflicts by analyzing corresponding encounters detected based on their own encounter criteria listed in Table 4.1. The traffic conflicts criteria must be determined based on crash estimation results and their trends analyzed for all encounters of the same type as conflicts (see Table 5.1).

The method of determining the best set of conflict thresholds (maximum conflict to collision, minimum longitudinal jerk, minimum lateral jerk, minimum impact speed) expressed in the  $s$ - $b$  coordinates can be summarized as estimating multiple times the number of crashes ( $Q_c$ ) expected in the encounter observation period under various sets of threshold values and searching for the best set. The best set is characterized with the largest number of observed conflicts used to obtain the consistent  $Q_c$  estimate that does not experience any trend (increasing or decreasing) in multiple estimates of this quantity obtained by each time gradually reducing threshold  $T_c$  while keeping the other thresholds at their best values. The following

TABLE 5.1  
Traffic conflicts criteria

Name	Description
$T_c$	Conflict maximum TTC
$J_c$	Conflict minimum longitudinal jerk (initial)
$S_c$	Conflict minimum lateral jerk
$A_c$	Impact speed—minimum magnitude of difference between two velocities at collision for conflicts

formula is applied to encounters to obtain the initially calculated  $Q_c$  estimate (Tarko, 2020):

$$Q_c = n \cdot 2^{-k} \quad (\text{Eq. 5.1})$$

where  $n$  is the number of observed events (encounters and crashes) and  $k$  is calculated with equation (Tarko, 2023):

$$k = \frac{n - c}{\sum_{i=1}^{n-c} \ln(1 + T_i/T_e) + c \ln(2)} \quad (\text{Eq. 5.2})$$

where:

$n$  = number of encounters and crashes,

$c$  = number of crashes,

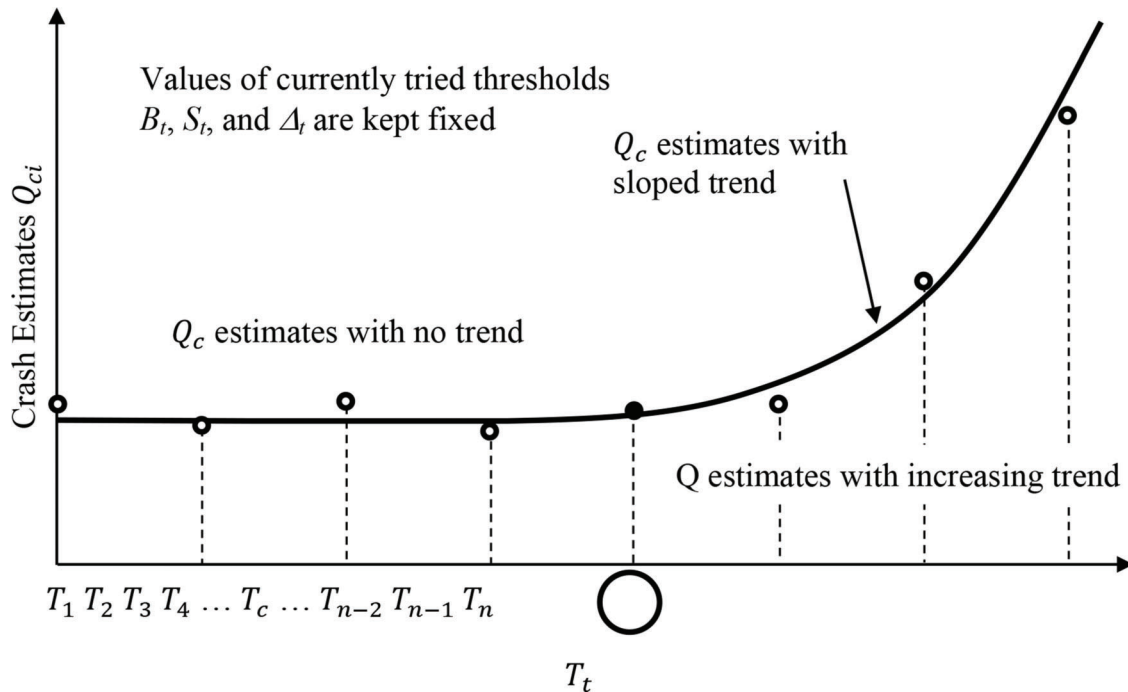
$T_e$  = maximum time to collision used to detect encounters, and

$T_i$  = observed time to collision in encounter  $i = 1.. n - c$ .

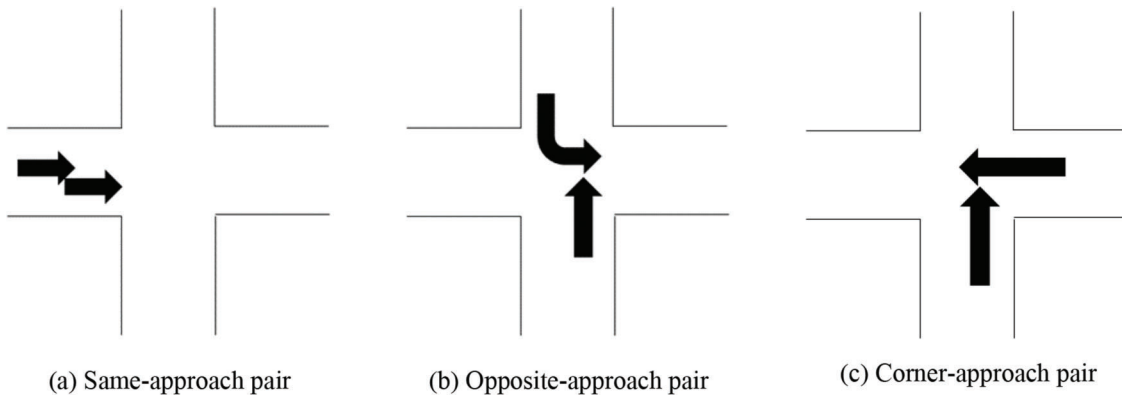
Figure 5.1 presents a typical sequence of estimated expected number of crashes resulted from applying a sequence of different tried thresholds  $T_t$  to the same set of observed encounters. The profile of the crash estimates includes a portion that is sloped and a portion that is flat. The attempt is to determine a breakpoint between the two portions which is the searched for  $T_c$ .

Let us consider  $n$  estimates of expected number of crashes  $Q_c$  obtained for threshold separations  $T_t$ ,  $t = 1.. n$ , and currently tried jerk and impact speed thresholds:  $B_t$ ,  $S_t$ ,  $A_t$  (Figure 5.1). The task is to determine the longest separation threshold  $T_t$  that belongs to the sequence of estimates exhibiting no trend. If the flat portion of the profile is confirmed, then the current threshold  $T_t$  is set at  $T_c$  as it produces unbiased  $Q_c$  estimates under the assumed other thresholds  $B_t$ ,  $S_t$ ,  $A_t$ . The above procedure of finding the  $T_c$  threshold for traffic conflicts under assumed ( $B_t$ ,  $S_t$ ,  $A_t$ ) triad may be extended to find the optimal  $B_c$ ,  $S_c$ , and  $A_c$  values by starting with  $B_t = B_e$ ,  $S_t = S_e$ ,  $A_t = A_e$  and repeating the procedure for gradually tightened thresholds  $B_t$ ,  $S_t$ ,  $A_t$ .

Each one-dimensional solution under assumed thresholds ( $B_t$ ,  $S_t$ ,  $A_t$ ) yields  $T_c$  estimate. Thus, the best solution from all the obtained solutions should be selected as the final one. The final solution is the one that delivers the lowest Akaike measure, and the number of traffic conflicts is at least the minimum (10 has been used successfully in the past). If multiple solutions with similar Akaike values are obtained, the solution with the largest number of conflicts in the flat part of the profile is selected. The proposed method is



**Figure 5.1** Profile of crash estimates obtained for a sequence of tried  $T_t$  thresholds—the identified breakpoint is  $T_c$ .



**Figure 5.2** Examples of a different approach pairs.

robust if they tried jerk and speed thresholds exhaustively cover the range of threshold values applied to observed traffic encounters.

In some cases, estimates  $Q_c$  of the expected number of crashes do not stabilize by the time the number of events reach their minimum. In such a case, the user may repeat the procedure by further aggregate the encounters into fewer but larger groups further within the same approach-pair type (Figure 5.2). An aggregated group of conflicts may include various pairs of maneuvers by vehicles entering from the same pair of corner approaches. There are four of such corner approach pairs at a four-leg intersection. Even higher aggregation is accomplished by combining the four pairs of corner approaches. Such aggregation exercise comes at a cost of losing the granularity of safety results. Only an aggregate expected number of crashes can be obtained.

If the expected number of crashes do not stabilize at the highest aggregate level of approach pairs, then this situation may indicate that the conflicts observation period was too short. Another possibility is that the observed conflicts are applied to too dissimilar types of collisions. For example, rear-end and right-angle crashes, and corresponding traffic conflicts, are quite different in many ways. Combining them together may be questionable. There is no formalized knowledge to propose any guidance other than good judgment and repeating the analysis.

### 5.3 Expansion Factors

The TScan system can be used continuously at one intersection for up to 4 days with a 12-hour observation period per day due to the system's limited battery capacity and human operation requirements. This

implies that traffic conflicts can only be observed continuously for a few days at a given location. After recharging the batteries, observations can be repeated multiple time. All the elements of the method are applicable to multiple disjoint periods that may be analyzed together. Even then, the overall observation period is much shorter than a year. A reliable method is thus needed to convert the collected conflict data to a standard road safety measurement namely annual crash frequency. Thus, the need for an expansion factor, defined as the ratio of annual expected crash frequency and expected crash frequency for the observation period.

Traffic events at an intersection are classified into three types based on the intersection legs from where two involved vehicles approach the intersection before colliding one with another. Namely, same-approach (SA) pairs, opposite-approach pairs (OA), corner approach (CA) pairs (Figure 5.2). A typical four-legged intersection would have ten different approach pairs as shown in Table 5.2.

To get the desired expansion factors between short and long periods, the crash probability would be analyzed at a very disaggregated level. According to current safety practice and the resolution of potential data at their sources, an hour is a proper unit for estimating crash probability and calculating the expansion factors afterward. Considering the limited traffic conflict data collection time by TScan every day, an hourly level analysis could allow more flexibility in the final expansion factors calculation.

The probability that a crash happens during an observation time interval (hour) is modeled with various explanatory variables regarding traffic, speed, road characteristics, and weather conditions. The total expected crash frequency is then the sum of the predicted crash frequency (one multiplied by the estimated crash probability) for each hourly observation categorized by each type of approach pair.

To calculate expansion factors, input datasets must be assembled for target intersections. The data preprocess and assembly procedure are similar to the sample preparation for modeling (Sections D.3 and D.4).

TABLE 5.2  
Approach pairs at a four-legged intersection

Approach 1	Approach 2	Type
W	W	SA
E	E	SA
N	N	SA
S	S	SA
W	E	OA
N	S	OA
W	N	CA
W	S	CA
E	N	CA
E	S	CA

The total number of records  $n$  in this dataset is given by

$$n = 24 \times 365 \times m \quad (\text{Eq. 5.3})$$

where  $m$  is the number of approach pairs for the target intersection. A typical four-legged intersection would have 87,600 observations. Using these observations and the logistic regression models developed for each type  $k$  of approach movement pairs (Section D.5) the corresponding expansion factors from a short period  $H$  with  $h$ -indexed hourly intervals to year  $Y$  with  $y$ -indexed hourly intervals are calculated with the following equation:

$$EF_{k|T,Y} = \frac{\sum_{y \in Y} P_{ky}}{\sum_{t \in T} P_{kh}} \quad (\text{Eq. 5.4})$$

where:

$EF_{k|H,Y}$  = expansion factor from a short period  $H$  to year  $Y$  for crashes between approach pair  $k$ ,

$P_{kh}$  = probability of crash between approach pair  $k$  in hour  $h$ , and

$P_{ky}$  = probability of crash between approach pair  $k$  in hour  $y$ .

## 6. EVALUATION

The following chapter describes the efforts to evaluate the method to detect encounters from trajectory data and identify traffic conflicts.

### 6.1 Evaluation Methodology

Data were collected at several intersections for short durations to initially improve the algorithms. Data collection for longer periods were performed at three intersections (described in the next sections) using the TScan units for evaluating safety through traffic encounters and conflicts.

Mechanical spinning LiDAR's such as the ones used in TScan units have an upper bound on the number of points they can generate per second. Therefore, increasing the scan rate decreases the number of points per scan. The LiDARs present in the TScan units are capable of scanning at 5–20 Hz. Based on prior experience, a scan rate of 10 Hz was chosen for all the data collection sessions as it provided the best balance between detection rate and positional accuracy.

### 6.2 Data Collected Using TScan Units

The data used in the evaluation were collected at intersections presented in the following sections. The two intersections present on Sagamore Parkway were suggested by SAC committee members. The intersection Morehouse Road was selected due to its geometry providing significant challenges to drivers approaching the intersection from W 350 N.

### 6.2.1 Intersection on Sagamore Parkway at Cumberland Avenue

This intersection is a signalized intersection with pedestrian crossings available two legs of the intersection located on Old US 52 (Sagamore Parkway) near West Lafayette’s Walmart (see Figure 6.1). The intersection experienced mixed traffic with an AADT of 19,275 on the major approach and 5,935 on the minor approach (INDOT, 2021). The speed limits posted on Sagamore Parkway and Cumberland Avenue are 45 mph and 30 mph, respectively. Detailed information on periods of data collection is shown in Figure 6.2

### 6.2.2 Intersection on Sagamore Parkway at Yeager Road.

This intersection is a signalized intersection located on Old US 52 (Sagamore Parkway) near West Lafayette’s Ace Hardware store. The intersection experienced mixed traffic with an AADT of 19,126 on the major approach and 8,708 on the minor approach (INDOT, 2021). The speed limits posted on Sagamore Parkway and Cumberland Avenue are 40 mph and 30 mph, respectively. Detailed information on periods of data collection is shown in table (Table 6.1). The TScan unit was parked in the parking lot of Ace Hardware, overlooking the intersection as shown in Figure 6.2.

### 6.2.3 Intersection on Morehouse Road at W 350 N

This intersection is an unsignalized, two-way stop-controlled intersection located on Morehouse Road

near Tippecanoe Memory Gardens Cemetery (see Figure 6.3). The road geometry at this intersection makes it challenging for users wanting to take a left from W 350 N on to Morehouse Road. The intersection experienced mixed traffic with an AADT of 2,577 on the major approach and 2,085 on the minor approach (INDOT, 2021). The speed limits posted on Sagamore Parkway and Cumberland Avenue are 40 mph and 30 mph, respectively. Detailed information on periods of data collection is shown in Table 6.1.

## 6.3 Crash Data

Crash data for the three intersections at which data were collected for the years 2018, 2019, and 2021. COVID-19 caused significant changes in traffic patterns for the year 2020. Therefore, crash data for the year 2020 was not considered. A summary of crash data for all three intersections is provided in Table 6.2.

When classifying data based on approach pairs, it is evident that same approach crashes are the most common among the three approach pairs. Rear-ends and sideswipes are the two manners of collisions possible for same approach-pairs. Between them, rear ends are the most common as shown by the collision diagrams of the two high-volume intersections (Figure 6.4 and Figure 6.5). These diagrams were generated using the Collision Diagram Builder (CDB) software tool developed by the Center for Road Safety, Purdue University (Tarko, Romero, et al., 2016).

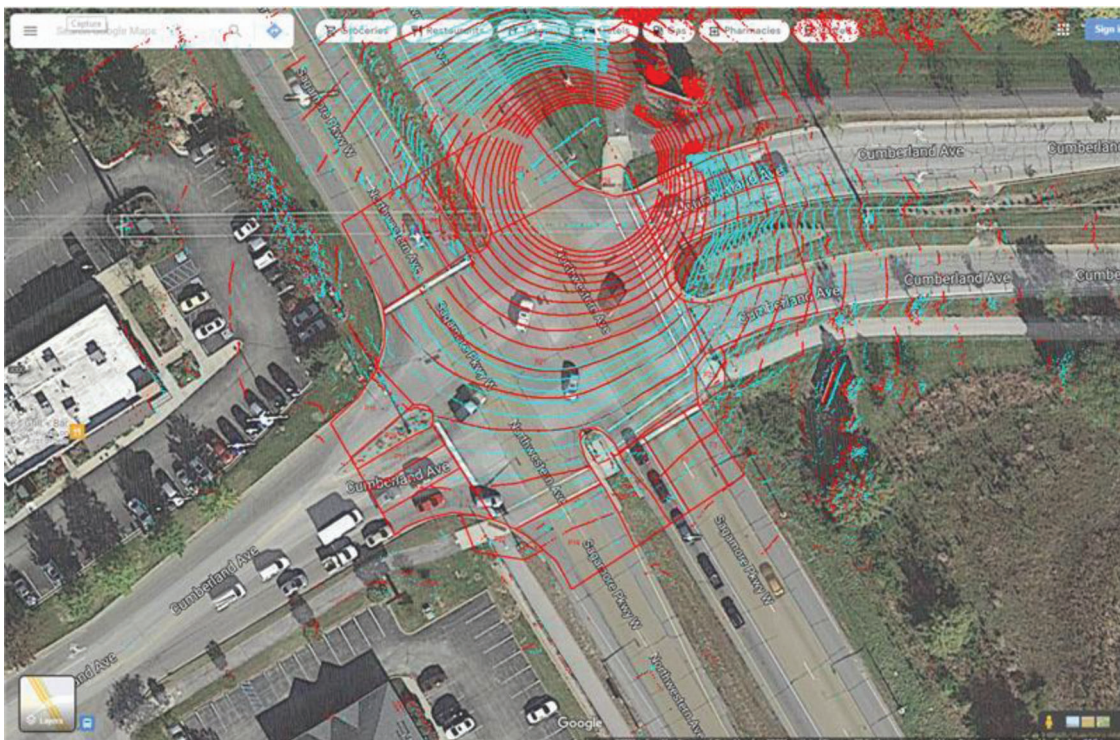


Figure 6.1 LiDAR coverage at the intersection of Sagamore Parkway and Cumberland Avenue.

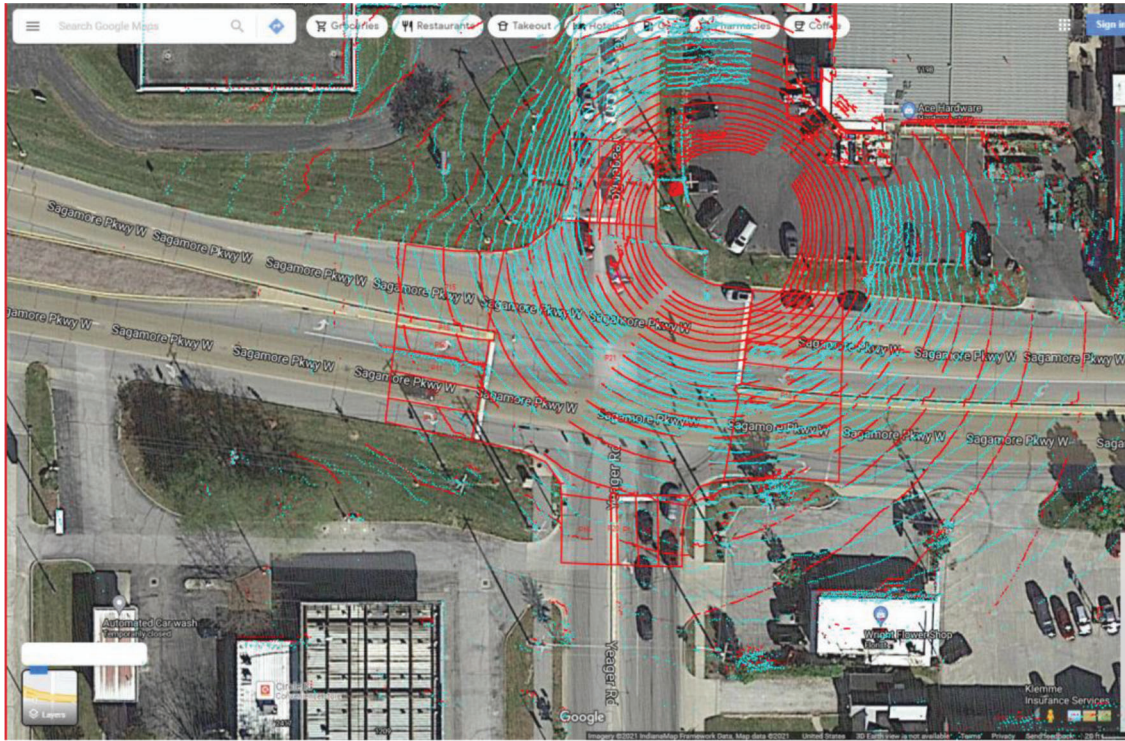


Figure 6.2 LiDAR coverage at the intersection of Sagamore Parkway and Yeager Road.

TABLE 6.1  
Data collected

Location	Date	Start Time	Data Collection		Weather	Temperature		Wind Speed (mph)
			Hours			Min (F)	Max (F)	
Sagamore–Cumberland	09-May-22	9:16 AM	12		Sunny	68	77	14
Sagamore–Cumberland	10-May-22	7:06 AM	12		Cloudy	57	77	21
Sagamore–Yeager	26-Apr-22	8:40 AM	6		Sunny	45	72	16
Sagamore–Yeager	26-Apr-22	2:40 PM	6		Sunny	64	72	25
Morehouse Rd–W 350 N	27-Oct-22	4:00 PM	3.25		Cloudy	46	57	11
Morehouse Rd–W 350 N	28-Oct-22	3:40 PM	3.5		Cloudy	54	61	12
Morehouse Rd–W 350 N	2-Nov-22	3:25 PM	4		Sunny	66	77	26
Morehouse Rd–W 350 N	3-Nov-22	4:30 PM	3		Cloudy	55	43	19

## 6.4 Results Comparison

The encounters engineering application tool was used to identify traffic encounters. The tool implements the procedure described in Chapter 4. The encounter diagrams generated by the tool are provided next to crash diagrams from CDB for each intersection in Figure 6.4, Figure 6.5, and Figure 6.6.

It is to be noted that the scale of the bars in both images are not equal, rather they are scaled relative to the total number of respective events displayed.

In the Cumberland intersection data, eastbound traffic turning right has high volume of vehicles that turn right on red. Moreover, the vehicles have to climb a vertical curve before they can see southbound vehicles. Similarly, vehicles southbound vehicles approaching the intersection from the east have to climb a vertical curve before they can see eastbound through

vehicle. These two issues are highlighted in the encounter diagram shown in Figure 6.4.

In the Sagamore Parkway-Yeager Road intersection, the trailer was parked close to a high-volume approach unlike Sagamore Parkway-Cumberland intersection where the trailer was parked near the low-volume approach. This results in a very different distribution of encounters.

The encounter diagram pattern confirms the initial assumption that vehicles approaching from W 350 N have difficulty turning onto Morehouse Road. Almost all of the encounters are vehicles trying to turn onto Morehouse Road.

As shown in Table 6.3, the number of encounters observed were low. Aggregating to the level of approach pairs and applying the procedure explained in Section 5.2, yielded no conflicts.

TABLE 6.2  
Crash data for selected intersections involving two vehicles

Road 1	Road 2	Crashes (2018, 2019, 2021)	Crashes by Approach Pair			Crashes by Severity		
			SA	OA	CA	PDO	Injury	Fatal
Sagamore Parkway	Cumberland Ave	57	49	0	8	44	13	0
Sagamore Parkway	Yeager Rd	37	27	1	9	28	10	0
Morehouse Rd	W 350 N	3	1	0	2	3	0	0

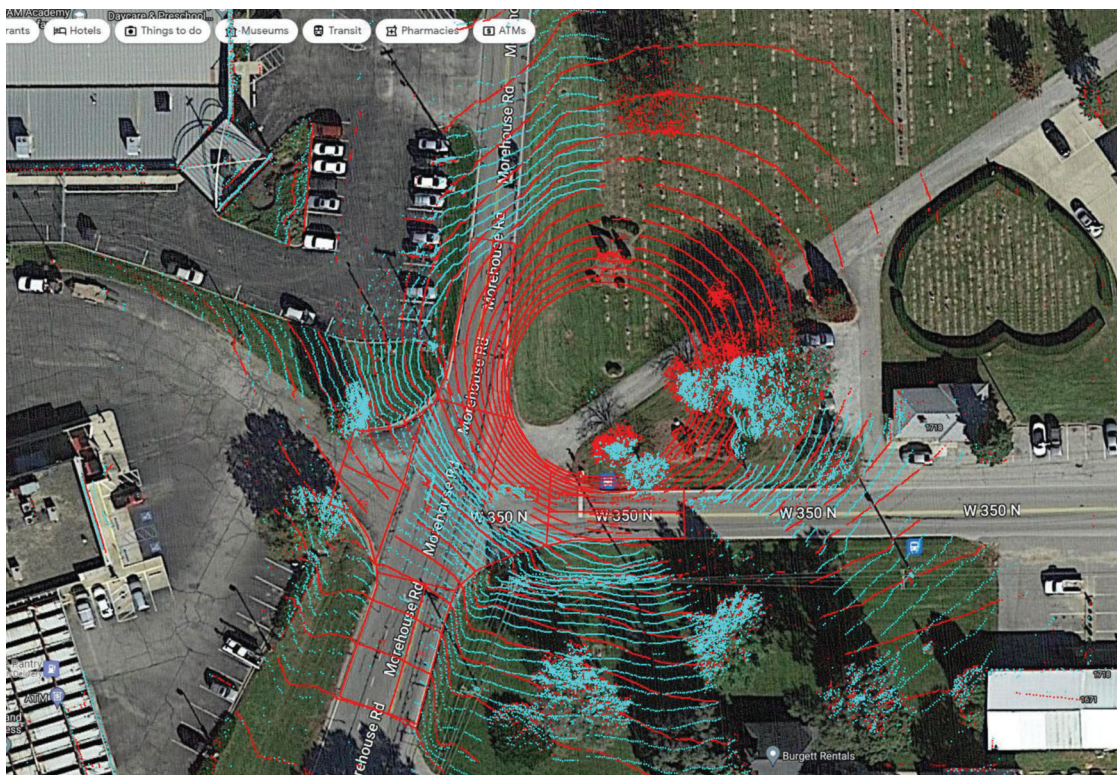


Figure 6.3 LiDAR coverage at the intersection of Morehouse Road and W 350 N.

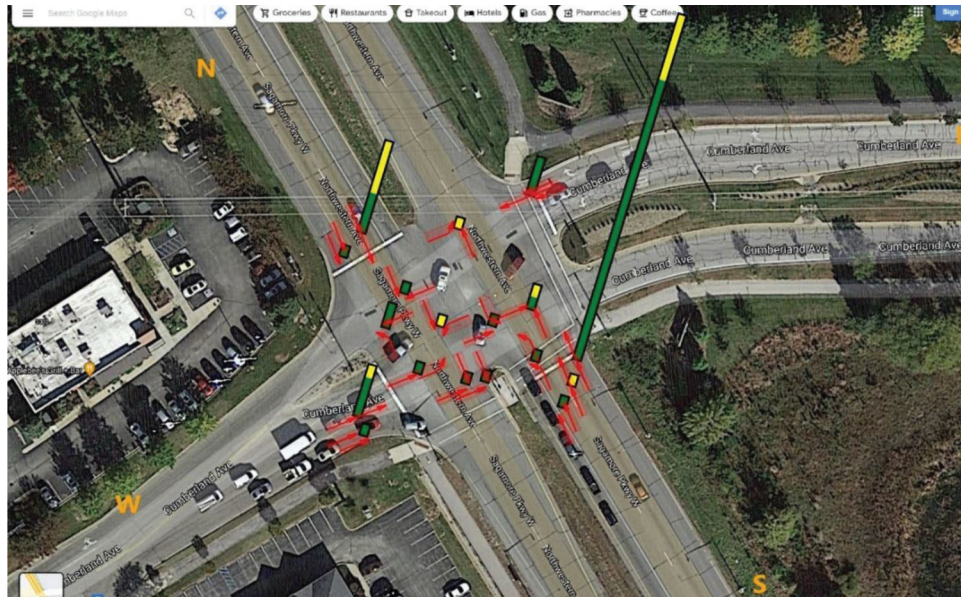
## 6.5 Learnings from Evaluation

The first and foremost conclusion that can be drawn is that longer data collection efforts are needed to get sufficient number of conflicts for safety analysis. Despite that, encounter diagrams do provide useful insights into the manner of collisions that are relatively more likely to occur.

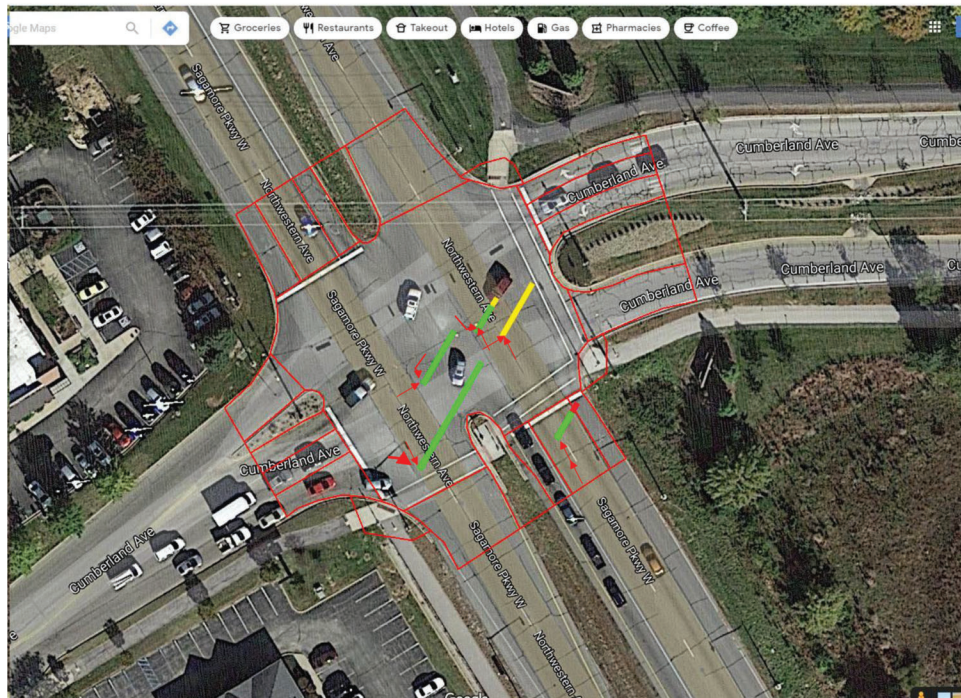
From the evaluations, it is clear that the proportion of rear end encounters observed at an intersection using TScan is not comparable to the proportion of rear-end crashes. The reason for this is twofold. First is the coverage of the intersection by the TScan unit. The units were positioned at a corner beside the intersection with the LiDAR units directed at the middle of the intersection. In such configurations, typically only two or three vehicles are present in the queue at each leg of the intersection. Longer queues can only be observed at approach closest to the corner of where the unit is

stationed. Second is that any crash that happens in a queue is generally attributed to the intersection in the crash reports. This means potentially, collisions occurring as far as 300 ft from the middle intersection are also attributed to the intersection. This is well beyond the detection range of current LiDARs.

One of the steps in identifying traffic encounters is that the path of the vehicle performing an evasive action is replaced with a trajectory of another vehicle that does not make such an action. In order to reliably replace trajectories, at least ten observations before the beginning of evasive action are needed. Therefore, if a vehicle performs evasive action before being tracked for at least ten observations, then that encounter is discarded due to lack of data for replacing trajectory. Higher scan rate could potentially decrease the impact of this problem but there is a trade-off. The current LiDAR models used in the TScan units are capable of



(a) Crashes (2018, 2019, 2021)



(b) Traffic encounters for total observed period (Table 6.3)

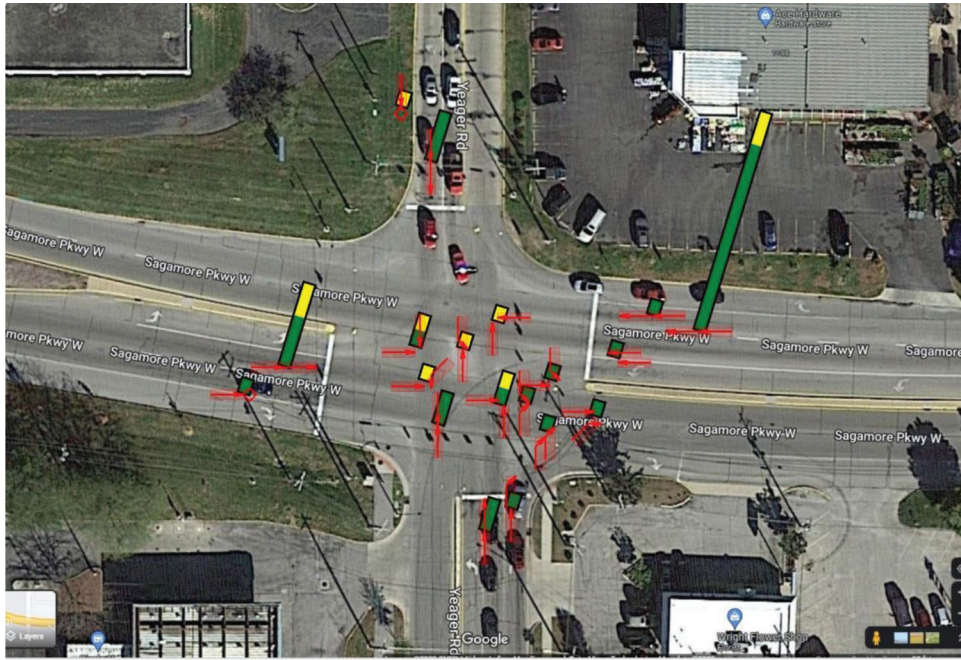
**Figure 6.4** Crashes vs. encounters at Sagamore Parkway and Cumberland Avenue.

20 Hz operations but in that mode the effective detection range of the LiDAR's decreases and the measurement error increases.

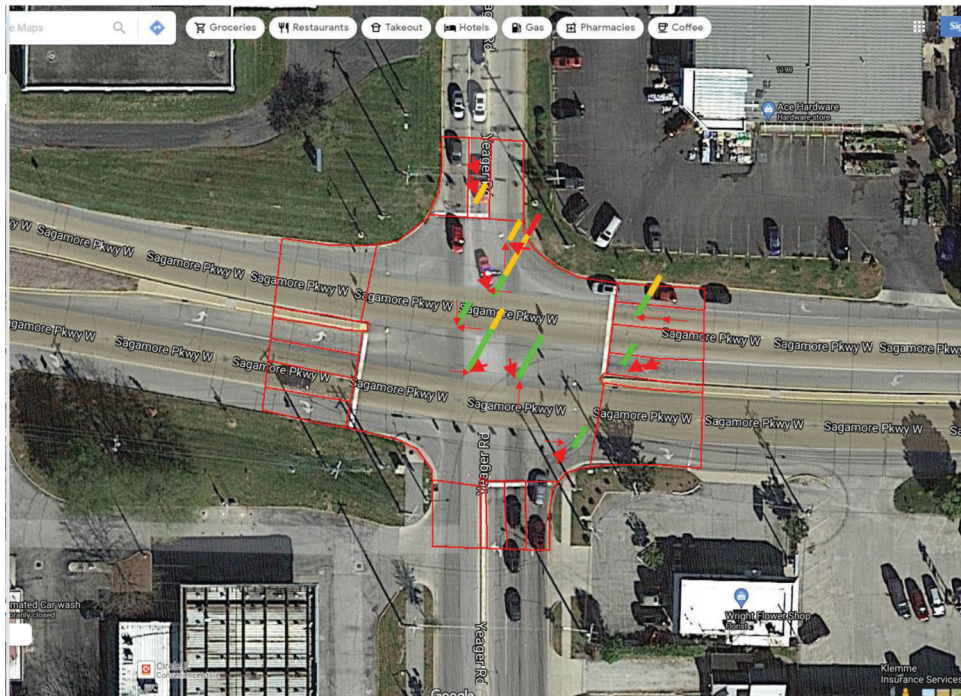
Another practical challenge when collecting data is finding ideal positions for stationing the trailer based TScan units. These units need level ground near an intersection that can be accessed with a vehicle to easily tow and position the trailer. This limits the number of locations where this system can be used. In addition, the area where the trailer is positioned should not have

any overhead high voltage electrical cables as it poses a safety hazard for raising the mast. Ideally, there should not be any infrastructure such as poles obstructing the field of view of the LiDAR's. These obstructions cause "shadows" or regions in space behind the obstructions, where there are no readings from the LiDARs. The algorithm is capable of handling shadows caused by light or signal poles so long they are not more than 1 foot on the ground. Larger obstructions are detrimental to the tracking algorithm's performance.



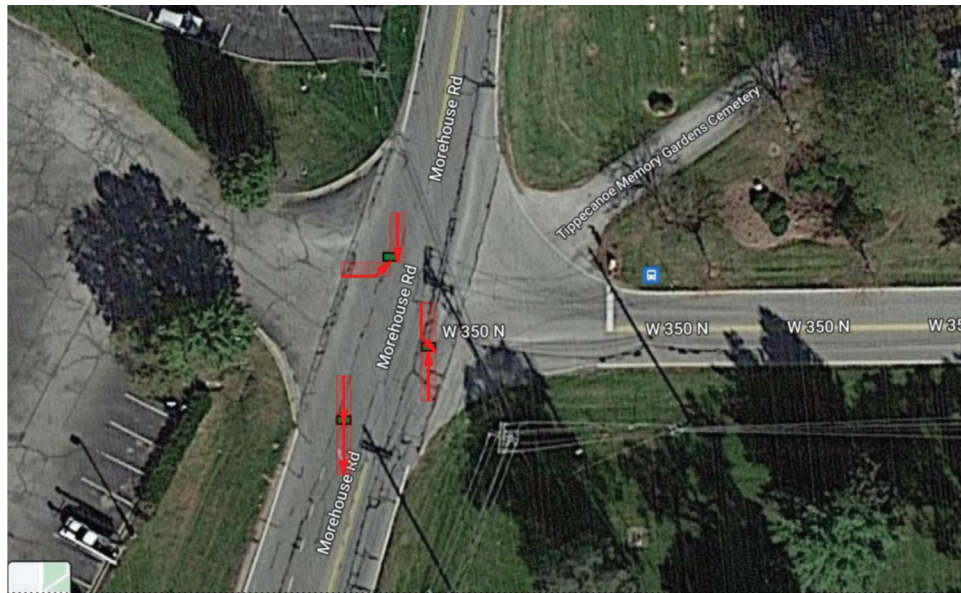


(a) Crashes (2018, 2019, 2021)

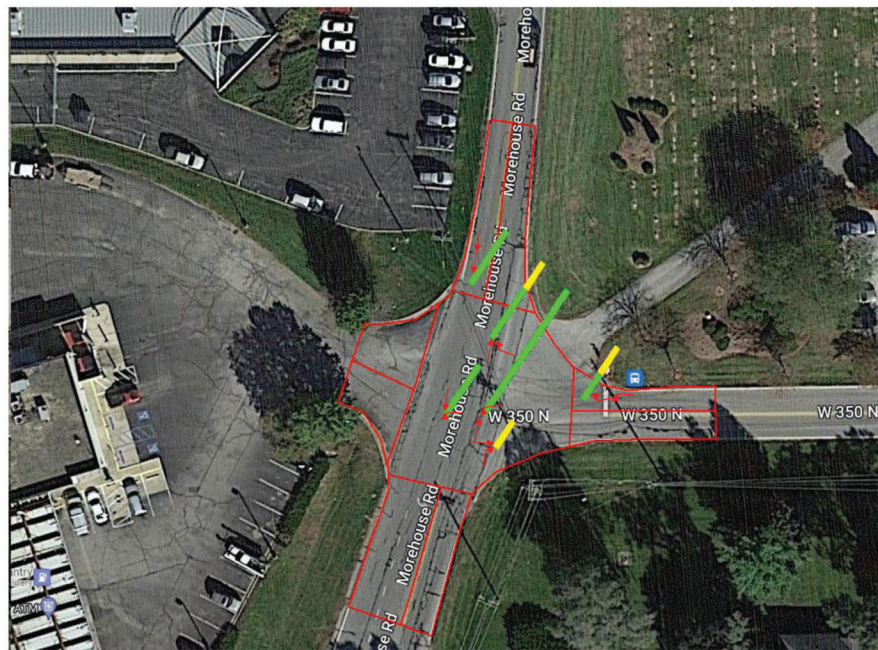


(b) Traffic encounters for total observed period (Table 6.3)

**Figure 6.5** Crashes at Sagamore Parkway and Yeager Road (2018, 2019, 2021).



(a) Crashes (2018, 2019, 2021)



(b) Traffic encounters for total observed period (Table 6.3)

**Figure 6.6** Crashes at Morehouse Road and W 350 N (2018, 2019, 2021).

**TABLE 6.3**  
Encounters, conflicts, and crashes at observed intersections

Location	Data Collection Period (hours)	Encounters	Conflicts	Crashes (from Table 6.2)
Sagamore–Cumberland	24	26	0	57
Sagamore–Yeager	12	29	0	37
Morehouse Rd–W 350 N	14	16	0	3

## 7. FUTURE WORK AND CONCLUSION

Methods and procedures for extracting traffic encounters from trajectories were developed. TScan units were deployed at intersections for long periods of time and traffic encounters were observed. The encounter diagrams revealed pairs of movement that potentially face safety issues. These relatively long data collection and processing efforts revealed some issues with the algorithm. Solutions to these issues have been developed and implemented. An end user application to extract traffic encounters from TScan and analyze them has been developed. In the future, this tool can be adapted to work with newer trajectory data sources such as those generated by autonomous vehicles or instrumented infrastructure.

A method to identify traffic conflicts, a subset of traffic encounters has been proposed along with a system of equations to estimate expected number of crashes for the observed period. Unfortunately, the data collected was not sufficient to identify conflicts among the list of traffic encounters observed. Further long-term data collection efforts such as installing a semi-permanent version of TScan at a single location for several weeks and repeating it for multiple locations would help further understand how long an observation period is necessary to observe sufficient number of traffic encounters that leads to a good estimate of number of crashes.

Significant time was spent on developing expansion factors defined as the ratio of annual expected crash frequency and expected crash frequency for the observation period. With both observed traffic encounters and estimated expansion factors, it is possible to supplement the current Indiana safety management with safety analysis based on the traffic encounters method.

Towards this regard a set of guidelines have been developed to help collect traffic encounters and to support its use in safety analysis.

The current expansion factors. Method can only predict annual crash frequency. For safety applications, knowing crash estimates based on severity provides greater value. Thus, more research is needed to develop models to split the estimated annual crash frequency by severity.

## REFERENCES

AASHTO. (2010). *Highway safety manual* (1st ed.). American Association of State Highway and Transportation Officials.

Atev, S., Miller, G., & Papanikolopoulos, N. P. (2010). Clustering of vehicle trajectories. *IEEE Transactions on Intelligent Transportation Systems*, 11(3), 647–657. <https://doi.org/10.1109/TITS.2010.2048101>

Bandaru, V. K., Romero, M. A., Lizarazo, C., & Tarko, A. P. (2021). *TScan—Stationary LiDAR for traffic and safety applications: Vehicle interpretation and tracking* (Joint Transportation Research Program Publication No. FHWA/IN/JTRP-2021/31). West Lafayette, IN: Purdue University. <https://doi.org/10.5703/1288284317402>

Buzan, D., Sclaroff, S., & Kollios, G. (2004). Extraction and clustering of motion trajectories in video. *Proceedings of the 17th International Conference on Pattern Recognition*, 2, 521–524. <https://doi.org/10.1109/icpr.2004.1334287>

Hermes, C., Wohler, C., Schenk, K., & Kummert, F. (2009). Long-term vehicle motion prediction (pp. 652–657). *2009 IEEE Intelligent Vehicles Symposium*. <https://doi.org/10.1109/IVS.2009.5164354>

INDOT. (2021). *Indiana Department of Transportation traffic count database system* [Webpage]. Retrieved November 15, 2022, from <https://indot.public.ms2soft.com/tcds/tsearch.asp?loc=Indot&mod=>

Kiefer, R. J., Leblanc, D. J., & Flannagan, C. A. (2005). Developing an inverse time-to-collision crash alert timing approach based on drivers' last-second braking and steering judgments. *Accident Analysis & Prevention*, 37(2), 295–303. <https://doi.org/10.1016/j.aap.2004.09.003>

Lefèvre, S., Vasquez, D., & Laugier, C. (2014). A survey on motion prediction and risk assessment for intelligent vehicles. *Robomech Journal*, 1(1). <https://doi.org/10.1186/s40648-014-0001-z>

Mollicone, D., Kan, K., Mott, C., Bartels, R., Bruneau, S., van Wollen, M., Sparrow, A. R., & Van Dongen, H. P. A. (2019). Predicting performance and safety based on driver fatigue. *Accident Analysis & Prevention*, 126, 142–145. <https://doi.org/10.1016/j.aap.2018.03.004>

Musican, O., Bar-Gera, H., & Schechtman, E. (2010). Electronic records of undesirable driving events. *Transportation Research Part F*, 13(2), 71–79. <https://doi.org/10.1016/j.trf.2009.11.001>

Tarko, A. P. (2018). Estimating the expected number of crashes with traffic conflicts and the Lomax Distribution – A theoretical and numerical exploration. *Accident Analysis & Prevention*, 113, 63–73. <https://doi.org/10.1016/j.aap.2018.01.008>

Tarko, A. P. (2019). Estimating safety with failure-related traffic conflicts: An example of road near-departures (Report No. 19-03682). *Transportation Research Board 98th Annual Meeting*.

Tarko, A. (2020). *Measuring road safety with surrogate events* (1st ed.). Elsevier.

Tarko, A. P., Ariyur, K. B., Romero, M. A., Bandaru, V. K., & Lizarazo, C. G. (2016). *TScan: Stationary LiDAR for traffic and safety studies—Object detection and tracking* (Joint Transportation Research Program Publication No. FHWA/IN/JTRP-2016/24). West Lafayette, IN: Purdue University. <https://doi.org/10.5703/1288284316347>

Tarko, A. P., & Lizarazo, C. G. (2021). Validity of failure-caused traffic conflicts as surrogates of rear-end collisions in naturalistic driving studies. *Accident Analysis & Prevention*, 149, 105863. <https://doi.org/10.1016/j.aap.2020.105863>

Tarko, A. P., & Lizarazo, C. G. (2023). Maximum likelihood method of estimating the conflict-crash relationship. *Accident Analysis & Prevention*, 179, 106875. <https://doi.org/10.1016/j.aap.2022.106875>

Tarko, A. P., Romero, M., Thomaz, J., Ramos, J., Sultana, A., Pineda, R., & Chen, E. (2016). *Updating RoadHAT: Collision diagram builder and HSM elements* (Joint Transportation Research Program Publication No. FHWA/IN/JTRP-2016/11). West Lafayette, IN: Purdue University. <http://dx.doi.org/10.5703/1288284316334>

## APPENDICES

**Appendix A. Guidelines for Traffic Encounters and Conflicts**

**Appendix B. TScan Engineering Applications Manual**

**Appendix C. TScan User Manual**

**Appendix D. Crash Expansion Factors**

## APPENDIX A. GUIDELINES FOR TRAFFIC ENCOUNTERS AND CONFLICTS

Table A.1 List of symbols

	<b>Observed or Computed Quantities</b>
$(x, y)$	Vehicle position in the system of coordinates $x$ and $y$
$(0, 0)$	Vehicle position in the system of coordinates $s$ and $b$
$\mathbf{v}$	Velocity (vector)
$v_x$	Magnitude of component velocity along axis $x$
$v_y$	Magnitude of component velocity along axis $y$
$v$	Longitudinal speed along axis $b$ ; lateral speed along axis $s$ is zero
$\Delta$	Differential velocity (vector) at collision of two road users, $\Delta = \mathbf{v}_1 - \mathbf{v}_2$
$\Delta_x$	Differential speed along axis $x$ at collision, $\Delta_x = v_{x1} - v_{x2}$
$\Delta_y$	Differential speed along axis $y$ at collision, $\Delta_y = v_{y1} - v_{y2}$
$\Delta$	Impact speed at hypothetical collision, $\Delta^2 = \Delta_x^2 + \Delta_y^2$
$\mathbf{a}$	Acceleration (vector)
$a_x$	Magnitude of component acceleration along axis $x$
$a_y$	Magnitude of component acceleration along axis $y$
$a_s$	Lateral acceleration rate (along axis $s$ )
$a_b$	Longitudinal acceleration rate (along axis $b$ )
$\mathbf{J}$	Jerk (vector)
$J_x$	Magnitude of component jerk along axis $x$
$J_y$	Magnitude of component jerk along axis $y$
$B$	Longitudinal jerk rate (along axis $b$ )
$S$	Sideway jerk rate (along axis $s$ )
$F$	Instantaneous time to collision
$T$	Time to collision
	<b>Determined or Tried Thresholds</b>
$F_m$	Maximum instantaneous time to collision
$\Delta_e$	Encounter minimum impact speed
$B_e$	Encounter minimum longitudinal jerk
$S_e$	Encounter minimum lateral jerk
$T_e$	Encounter maximum time to collision
$\Delta_c$	Conflict minimum impact speed
$B_c$	Conflict minimum longitudinal jerk
$S_c$	Conflict minimum lateral jerk
$T_c$	Conflict maximum time to collision
$\Delta_t$	Tried minimum impact speed
$B_t$	Tried minimum longitudinal jerk
$S_t$	Tried minimum lateral jerk
$T_t$	Tried maximum time to collision

## **A.1 Introduction**

Traffic encounters are a wide category of events that bring a possibility of crash. In a less frequent case of traffic conflicts, an error made by a road user leads to an unacceptably short time to collision and an uncharacteristically strong reaction from a road user to prevent a collision. In a more frequent case, an involved road user controls the situation; thus, the time left to a potential collision is longer than during a traffic conflict.

The existence of an error makes a traffic conflict etiologically consistent with a crash. It is because traffic conflicts and crashes are caused by errors. Furthermore, assuming that an error precedes a traffic conflict simplifies its analytical representation because it is easier to observe an error than to predict its occurrence. Estimating the probability of collision given an error is reduced to estimating the probability of a response time longer than the time remaining to a looming collision.

Although traffic conflicts, as defined here, are much more frequent than crashes, their frequency is still quite low. It takes days or even weeks to observe conflicts with advanced vehicle-tracking instruments, such as TScan, set at selected locations. This hurdle will be reduced or even eliminated in the future with the expected presence of tracking devices in automated vehicles and in the roadside to support operations of these vehicles. Until that time, safety analysts and engineers must reach for the mentioned special instruments and for more frequent traffic encounters to reduce observation periods. Traffic encounters supplemented with available traffic conflicts are the promising avenue to learn more and faster about safety than from crash reports.

Error-free non-conflict encounters reflect aggressive behavior of road users who seem to control the situation. Even then, a non-conflict encounter is not free of risk of crash. Due to a close proximity of other vehicles, any execution error (for example, faulty steering) or perception error (for example, insufficient pavement friction or obstructed sight of another vehicle) may convert the event into a traffic conflict or a collision.

A traffic encounter is closer to a collision (see pyramid of events in Figure A.1) and more relevant to safety than an unaffected passage of an intersection. Consequently, the volume of traffic encounters is a better measure of exposure to crash than the volume of vehicles entering an intersection. Furthermore, analyzing types of traffic encounters, their distribution in an intersection area, and circumstances under which they occur may help determine causes of these events and prompt corresponding safety countermeasures.

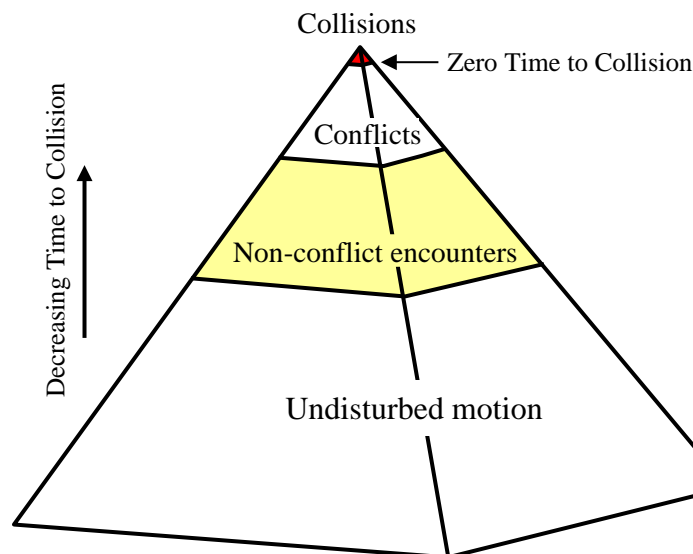


Figure A.1 Pyramid of traffic events.

This section introduces the definition of traffic encounters useful for automated detection and classification of these events at two levels of interaction severity: (1) traffic conflicts, and (2) non-conflict encounters. For the purpose of automated detection of traffic encounters, all road users visible in the field of view (FOV) must be tracked at a sufficient accuracy. Motion of vehicles and pedestrians is represented with their consecutive positions over time (trajectories) when passing through the FOV. Although the following description focuses on vehicles, it applies also to pedestrians, bicycles, and scooters.

A data collection period should be sufficiently long, and its length depends on the severity of encounters to be detected. The length of observation period depends also on the traffic itself. Low volume roads with fewer conflict points require longer periods. Past research indicates that an observation period may vary from several hours to several days or even several weeks.

The following sections introduce the counterfactual analysis applied to pairs of trajectories in potential encounters and more specific information is provided regarding these three quantities. Proper setting of thresholds of these quantities to detect traffic conflicts and non-conflict encounters are also discussed.

## A.2 Traffic Observations for Safety Analysis

Traffic Scanner (TScan) developed by the Purdue Center for Road Safety can track moving objects within its FOV (SPR-4102 report). In the applied tracking technique, a vehicle is represented with a bounding box—a tight rectangle that fully includes a tracked object. The center of the box  $(x, y, t)$  represents a position of the tracked vehicle on  $(x, y)$  plane at time  $t$ . The position is updated 10

times per second. The tracking results also provide three elements at each time  $t$ : velocity  $\mathbf{v}$ , acceleration  $\mathbf{a}$ , and jerk  $\mathbf{J}$  that characterize an event and help determine if the event is indeed an encounter.

Tracking provides object's velocity  $\mathbf{v}$  expressed through its component magnitudes along  $x$  and  $y$  axes:  $v_x$  and  $v_y$ , respectively. Acceleration  $\mathbf{a}$  is the rate at which velocity changes its magnitude and direction. Acceleration  $\mathbf{a}$  is represented through its component magnitudes:  $a_x =$  change rate of speed along  $x$ , and  $a_y =$  change rate of speed along  $y$ . Finally, jerk  $\mathbf{J}$  is the rate at which acceleration is changing and can be represented with its component magnitudes  $J_x$  and  $J_y$ .

Each of the three motion characteristics (velocity, acceleration, jerk) can be converted to its overall magnitude by combining its component magnitudes along  $x$  and  $y$  axes. The resulted overall magnitudes are values that do not carry the notion of direction. They describe road user's current overall speed  $v$  and road user's overall responses to current traffic situation via  $a$  and  $J$  values, but these values cannot be used to distinguish between swerving and braking maneuvers—actions most frequently used to avoid collisions. A more intuitive representation of the three motion characteristics is accomplished by representing them in a system of coordinates that are tied up with the current direction of motion: longitudinal ahead (axis  $b$ ) and lateral right (axis  $s$ ). The three motion characteristics decomposed onto the new ( $s, b$ ) axes. These two systems of coordinates and the corresponding decomposed motion characteristics are illustrated in Figure A.1.

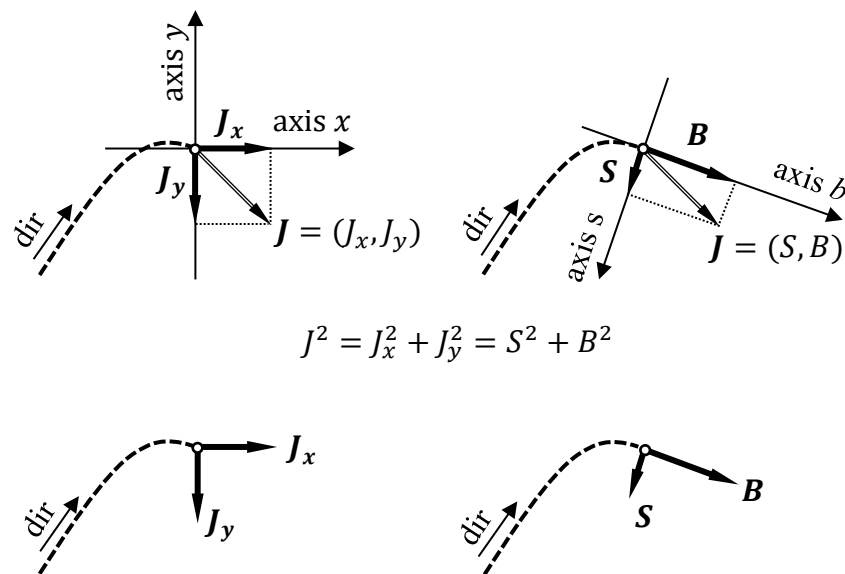


Figure A.2 Coordinates used in tracking moving objects.

Since velocity  $\mathbf{v}$  at any point determines vehicle's motion direction, longitudinal component  $v_b = v$  and lateral component  $v_s = 0$ . On the other hand, the magnitudes of component acceleration:  $a_s$  and  $a_b$  and magnitudes of component jerk:  $S$  caused by swerving and  $B$  caused by braking may take any values. According to Figure 1, positive acceleration  $a_b$  indicates increasing speed, while



negative  $a_b$  indicates braking. Positive acceleration  $a_s$  indicates turning (swerving) right and negative acceleration  $a_s$  indicates turning (swerving) left. Positive jerk  $B$  coupled with positive acceleration  $a_b$  indicates increasing acceleration. Non-zero speed  $v_b$ , negative constant acceleration  $a_s$ , and zero jerk  $S$  correspond to turning left along a circular curve. If in the previous case of turning left along a circular curve, jerk  $S < 0$ , then a vehicle is turning left along a spiral curve with growing curvature. Figure A.2 demonstrates a non-zero  $S$  component along axis  $s$  that indicates a right-turning along a spiral-like curve with a change of acceleration along the path.

### **A.3 Traffic Encounters**

#### *A.3.1 Conditions of Encounter Potential*

A traffic encounter occurs between two road users. Trajectories for any two vehicles simultaneously present in the FOV must be analyzed to determine if the two vehicles are involved an encounter between themselves. There may be secondary encounters caused by preceding (primary) encounters. These secondary events are considered independently from the primary ones.

Two road users can be involved in an encounter between themselves only if their paths cross or overlap one another. Thus, the potential for an encounter is determined to some extent by the interstation's geometry, number of lanes on its approaches and exits, and turning movements permitted. Another condition of an encounter between two road users is their simultaneous presence near a potential collision spot, heading towards the spot with a potential of arriving at the spot at approximately the same time. This temporal condition is reflected with an instantaneous time to collision.

These two necessary conditions of encounter: spatial and temporal are discussed in the next two sections.

#### *Spatial Characterization*

An intersection OD matrix specifies permitted one-way traffic connections between intersection entry lanes and intersection exit lanes. An encounter matrix, on the other hand, specifies pairs of one-way traffic connections that may be involved in traffic encounters between themselves in and around the intersection area. Consequently, any two vehicles passing through the intersection have assigned types of potential encounters and their general locations at the intersection. These types of encounters correspond to types of collisions: rear-end, side-swipe, right-angle, merging, etc.

#### *Instantaneous Time to Collision*

Any two road users simultaneously present in the FOV may be involved in their traffic encounter. To reduce the number of pairs for further time-consuming analysis, a vehicle is eliminated if it is well separated from all other vehicles during its appearance in the FOV. A vehicle is well separated from another vehicle at a certain instant if, so-called, instantaneous time to collision  $T$  is longer than maximum instantaneous time to collision  $T_m = 5$  s. The  $T$  value for any two vehicles is calculated repeatedly every 0.01 s. In this calculation, the current orientations, and speeds of the

two vehicles are assumed to remain constant for the remaining time of presence in the FOV.  $T$  takes a specific value at the moment when the two vehicles represented with their bounding boxes collide inside the FOV when moving under the assumed speeds and directions.

### *A.3.2 Indicators of Encounter Occurrence*

As mentioned in the introduction section to traffic encounters there are two general cases of encounters: (1) non-conflict encounters with aggressive and controlled behavior that brings a considerable risk of crash in case of an error, and (2) traffic conflicts where an error occurs and a last-moment response of one of the interacting road users eliminates the threat. Three indicators are used to identify these events: considerably quick response (jerk) to avoid collision, a short time to a hypothetical collision at the time of the jerk, and considerable impact speed at a hypothetical collision if the evasive maneuver (jerk) does not occur timely.

Road users who are too much separated from others to face any potential encounter (long instantaneous time to collision) are eliminated from further consideration to save processing time. Trajectories of road users, who survive the first elimination step, are checked for presence of a considerable jerk. For pairs of road users with at least one considerable jerk  $B$  or  $S$  observed, the imminence of hypothetical collision is checked, the time-to collision  $T$  calculated and the impact speed  $\Delta$  estimated. The traffic encounter is claimed if the potential of collision is confirmed, the  $T$  value is short, the  $\Delta$  is sufficiently high to expect a considerable harm produced by the hypothetical crash if realized.

#### *Jerk*

Driver's brake or swerve to avoid a collision with braking being more often used than swerving. Braking can be applied quicker, and it does not pose a danger of getting in a path of another vehicle, which is likely in dense traffic. Braking is a longitudinal negative acceleration, and swerving is a lateral acceleration to the left or right.

A jerk is the rate at which an acceleration is changing. Recent research indicated that a strong jerk is a better indication of a corrective action in a traffic conflict than a strong acceleration (braking). Strong acceleration (positive or negative) is frequently applied by aggressive drivers who tend to apply strong braking gradually. On the other hand, a rapid increase or decrease of acceleration (jerk) suggest an adjustment. A sudden burst of braking indicates correction - a quick reaction to likely overlooked hazard. Particularly obvious indication of correcting an error is a sudden change of motion direction implemented via a strong lateral jerk. Drivers avoid such behavior as it brings a risk of losing stability of a vehicle on the top of the mentioned concern about causing a secondary risk.

Attaining a considerable acceleration rate within a reasonable time while controlling the motion leads to a jerk weaker than in an emergency situation. For these reasons, jerk has been chosen to detect both cases of encounters: conflicts and non-conflict encounters.

In a case of several considerable jerks made by a driver or both drivers, the jerk which preserves a collision from happening is considered and the other ones are ignored. Such a jerk concurs with the locally shortest time to collision (explained in the next section), and it is followed by a rapid increase in the time to collision value.

#### *Time to Collision*

Time to collision  $T$  is the time remaining to a hypothetical collision. Since the collision is not observed, a counterfactual technique must be used that considers scenarios that are likely but do not occur. In the case considered here, a collision between two road users (vehicles) involved in interaction continue their motions without applying an evasive action, controlled or uncharacteristically rapid. This evasion is manifested via a considerable jerk in at least one of the two trajectories. The trajectory affected by the jerk is replaced with a plausible (counterfactual) alternative. Such a trajectory is looked among trajectories of other road users obviously not involved in any encounter. A replacement trajectory is plausible if the original and replacement trajectories are similar until the point of evasion, the two trajectories follow same intersection OD path, and the two vehicles are of same type. A replacement trajectory should occur close in time to the original trajectory to ensure similar external conditions (travel purpose, weather, pavement conditions, etc.). If a hypothetical collision is confirmed after replacing the original trajectory, the  $T$  value (counterfactual) is estimated. Otherwise, the event is not considered to be a traffic encounter.

#### *Impact Speed*

Impact speed  $\Delta v$  is the magnitude of difference between velocities of two vehicle at the time of hypothetical collision. Impact speed  $\Delta$  is estimated from trajectories of two involved vehicles: observed evasive-free trajectory and a counterfactual replacement trajectory of a vehicle that performed an evasive maneuver. Thus, the estimated impact speed is a counterfactual result and not observed one.

#### *A.3.3 Detecting Traffic Encounters*

Traffic encounters are observed with the TScan system, which has been developed at the Purdue Center for Road Safety and available to INDOT. It is a battery-operated trailer equipped with a telescoping mast and two LiDAR sensors attached to its top. Preparing files before the field activities, setting the trailer, and running the data collection in the field, and processing the data after returning to the office are described in TScan User Manual (Appendix C). Several user applications facilitate extracting and analyzing the collected data. Their manuals are available in Appendix B.

As explained in the previous sections of these guidelines, tracking vehicles and pedestrians with TScan produces data that are picked up by the end user applications for further processing. The first step is to set threshold values including the encounter maximum  $T$ , minimum longitudinal and lateral jerks, and minimum impact speed (Table A.2). Applying these threshold values to

trajectories vehicles, three files are generated: the applied threshold values, the encounter file (Table A.3), and the units file (Table A.4).

Table A.2 Thresholds used to detect traffic encounters

Symbol	Encounter Threshold Name	Recommended Value
$B_e$	Minimum longitudinal jerk	10 ft/s <sup>3</sup>
$S_e$	Minimum lateral jerk	6 ft/s <sup>3</sup>
$T_e$	Maximum time to collision	5 s
$\Delta_e$	Minimum impact speed	3 mi/h

*Thresholds and Descriptive Variables*

Table A.3 Variables that describe identified traffic encounters

Name	Description
EncounterID	Integer unique (in the observation period) to identify the encounter
DayOfWeek	Day of week
DateTime	Date and time when the conflict happened
LatitudeCollision	Location of the hypothetical collision measured as the angular distance north or south of the earth's equator
LongitudeCollision	Location of the hypothetical collision measured as the angular distance east or west of the meridian at Greenwich
MannerOfCollision	Hypothetical manner of collision: rear end, right angle, same direction sideswipe, opposite direction sideswipe
TimeToCollision ( $T$ )	Time to collision calculated for the encounter between the two listed objects in the units table
DeltaVAtCollision ( $\Delta$ )	Magnitude of the difference between two velocities at collision

Table A.4 Unit variables that describe involved road users and actions during encounters

<b>Name</b>	<b>Description</b>
LocationID	Location ID
EncounterID	Unique (in the observation period) integer to identify the encounter
UnitID	Unique (in the observation period) integer to identify the unit
UnitType	Non-heavy vehicle, Heavy Vehicle, Pedestrian, Bicyclist/scooter
TurningManeuver	Turning maneuver (left1, left2, through, right1, right2, U-turn)
EncounterWithID	ID of another unit involved in the encounter
EntryPolygonID	Entry polygon ID
ExitPolygonID	Exit polygon ID
UnitImpactType	Bullet or target
LatAtEvasion	Location where the evasion starts
LongAtEvasion	Location where the evasion starts
HeadBeforeEvasion	Heading before evasion
HeadAtCollision	Heading at hypothetical collision
SpeedBeforeEvasion	Speed at the beginning of evasion
SpeedAtCollision ( $v$ )	Speed at hypothetical collision
LongJerkAtEvasion ( $B$ )	Longitudinal jerk at evasion
LatJerkAtEvasion ( $S$ )	Lateral jerk at evasion
LatAccAtEvasion ( $a_s$ )	Lateral acceleration at evasion (plus to right, minus to left)
LongAccAtEvasion ( $a_b$ )	Longitudinal acceleration at evasion

Figure A.3 shows the user interface of the encounter detection tool. After the TScan results are accessed with the application, the intersection movement O/D matrix (interface area 1) is created as described in the user manual (Appendix B.11). Clicking the *Extract* button (area 3) in the *Encounter* window (area 2) starts the process of processing each potential encounter event. Only events that meet the encounter thresholds are considered as valid encounters and they are displayed in the *Encounter* window. The results of the analysis include two lists: the list of detected traffic encounters (area 4), and the list of road user actions (area 5).

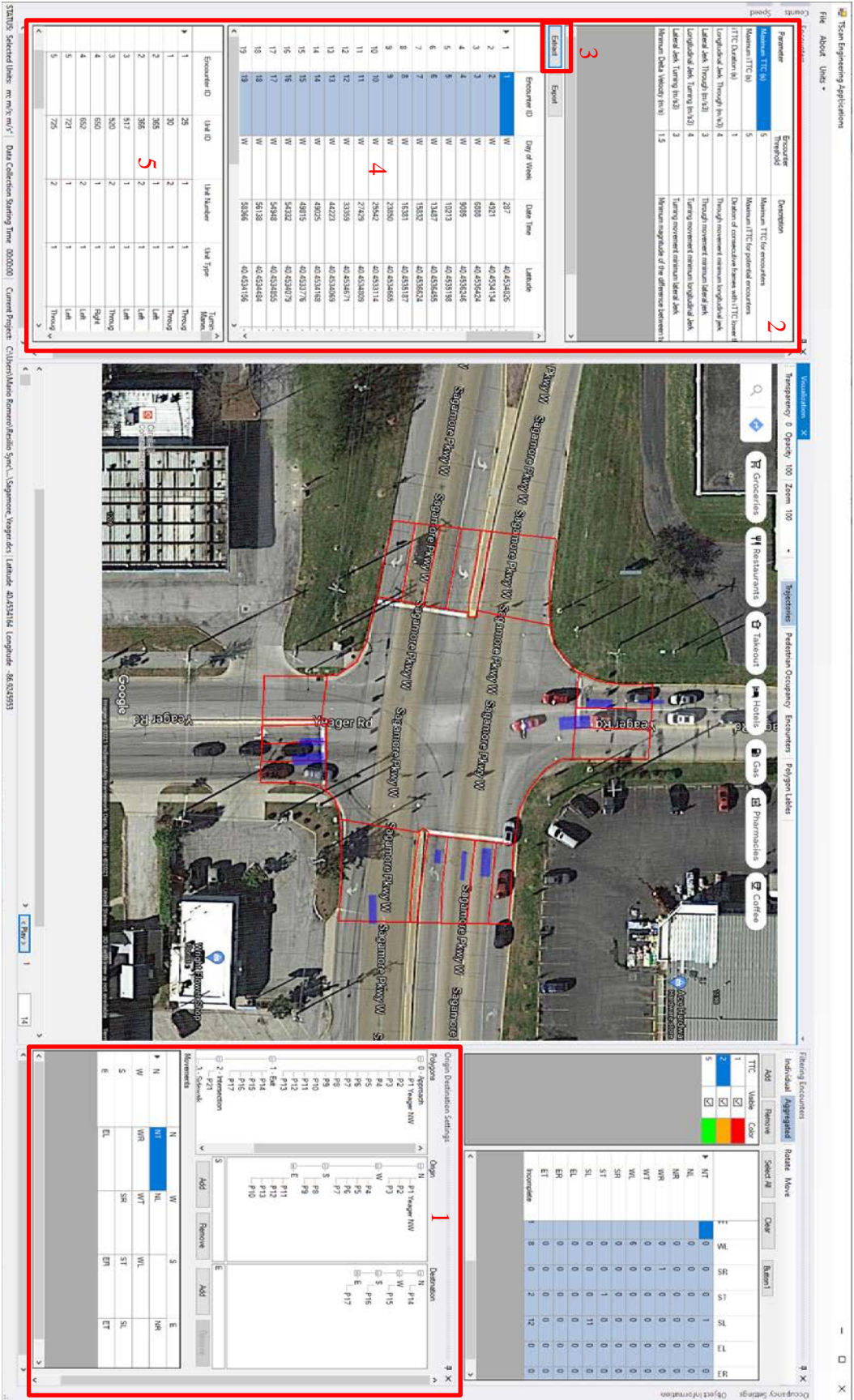


Figure A.3 Traffic encounters detection.

### *A.3.4 Safety Analysis with Traffic Encounters*

#### *Grouping Traffic Encounters*

Traffic encounters should be grouped based on pairs of conflicting paths.

#### *Encounter Diagram*

Once the *Encounter* mode (area 1) is selected in the visualization form the encounter diagram will be shown. The encounter diagram tool displays locations of hypothetical crashes (marked with dots) associated with traffic encounters. Another option is to display these hypothetical crashes grouped by pairs of movements of interacting vehicles and shown through bars. The height of bars in the *Aggregate* mode (area 2) is proportional to the number of traffic encounters. Both the dots and the bars are color-coded by severity measured with the Time to Collision (area 3).

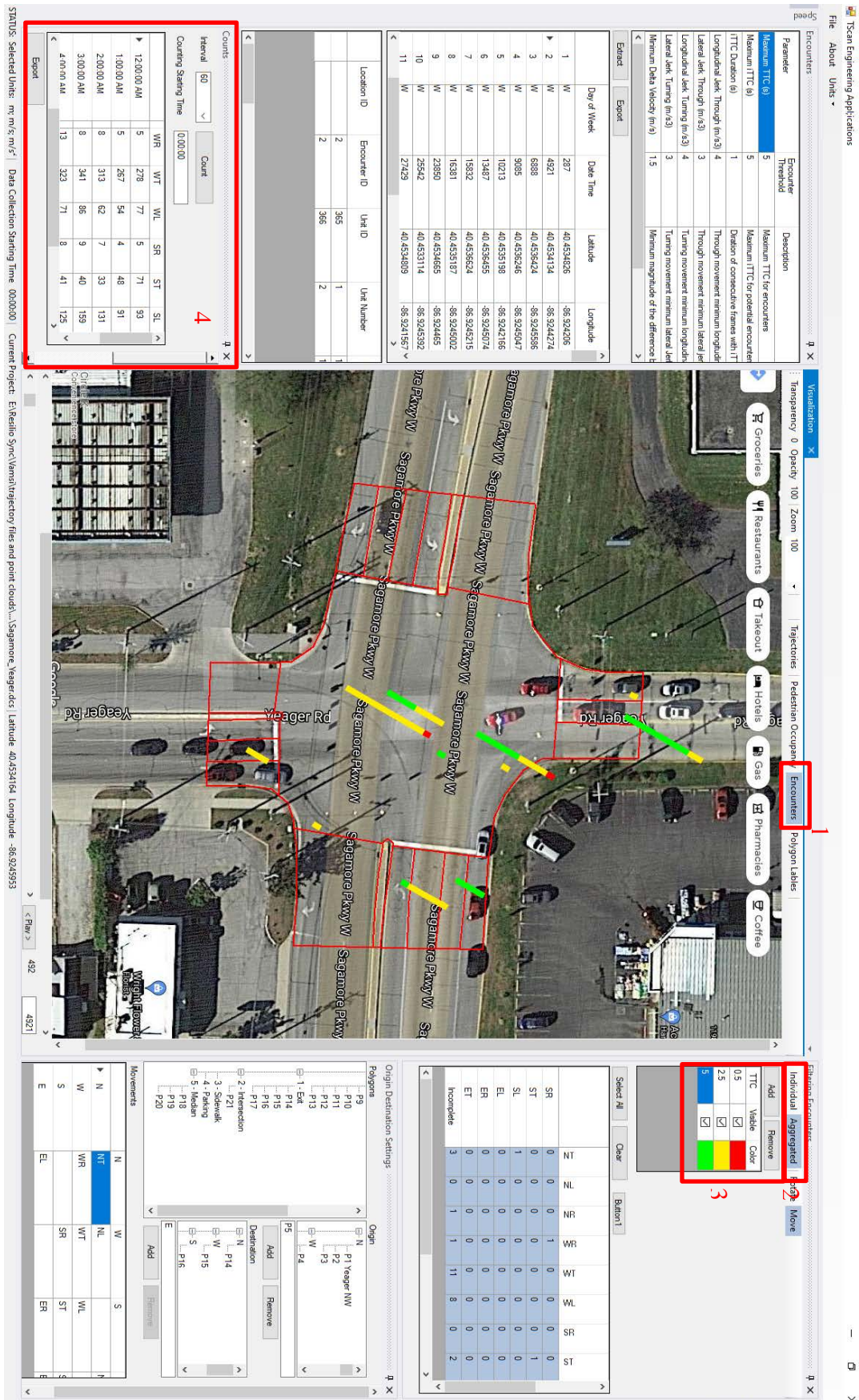


Figure A.4 Traffic encounters diagram.



### *Example Analysis of Traffic Encounters*

The traffic encounter analysis tool allows visualizing encounters at the entire intersection as shown in Figure A.4. The example diagram clearly shows that the left turn movements have the highest number of encounters among all the movements, and especially the southbound left and westbound left movements. This result is consistent with the intersection left-turn traffic volumes (area 4). Figure A.5 shows the traffic encounters diagram for a selected southbound left movement. It also allows to select any combination of movements selected by the user.

It is also important to highlight that the rear-end crashes that occur on approaches to the intersection within 200 ft from the intersection center are considered in Indiana as intersection crashes while the rear-end encounters shown in the diagram are limited to the field of view of the TScan system, which is typically shorter than 200 ft in the typical TScan field setting

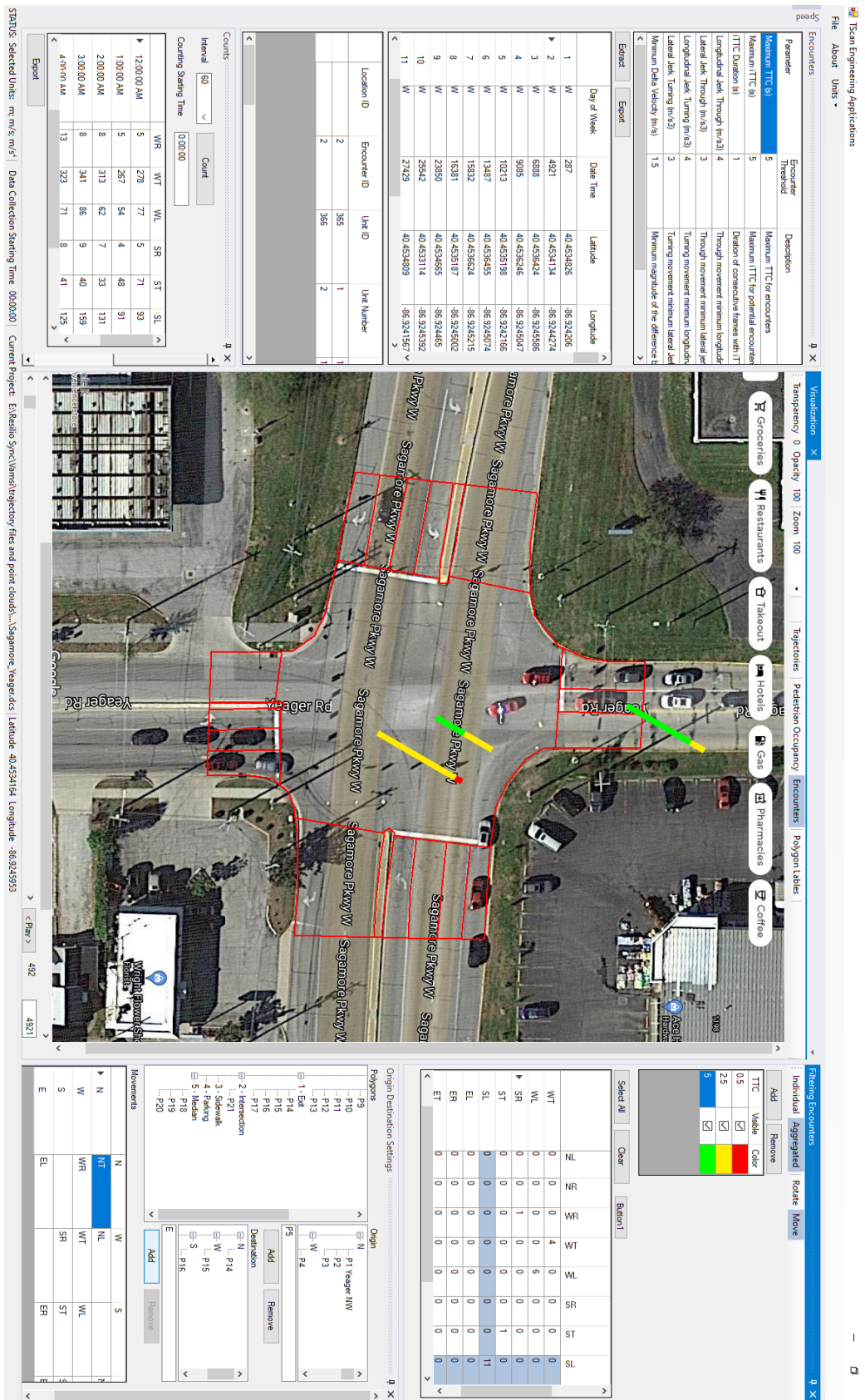


Figure A.5 Traffic encounters diagram for a selected movement.

## A.4 Traffic Conflicts

A conflict results from a human error or from other types of failure of vehicles or road such as pothole, traffic signals malfunctioning or shortage of sight distance. The resulted events are characterized with  $T$  values that are not acceptable even to aggressive drivers. Thus, the observed evasive maneuvers are stronger than ones performed by aggressive drivers who tend to control the risk. Consequently, conflict threshold  $T_c$  is unacceptably low and shorter than encounter's threshold  $T_e$ , while conflict jerk thresholds  $B_c$  and  $S_c$  are stronger than the corresponding encounter's thresholds.

Past research indicated that impact speed  $\Delta$  sufficiently high to report the collision to the police is a good choice for traffic conflicts threshold (Tarko & Lizarazo, 2021). Such a speed makes a potential outcome of the hypothetical event safety-relevant and makes behavior of involved road users indicative of being afraid of a potential collision. The same impact speed threshold  $\Delta v$  should be selected for safety-relevant encounters. The speed impact threshold may be tested when determining  $T_c$ ,  $B_c$ , and  $S_c$  thresholds explained in the next sections.

Traffic conflicts should be grouped based on the types of corresponding collisions and the number of observed and available events. The following calculations and analysis apply to each group of conflicts.

### A.4.1 Equivalent Number of Crashes

The method of determining the best  $(T_c, B_c, S_c)$  triplet can be summarized as estimating multiple times the number of crashes expected in the encounter observation period under various triplets and searching for the best triplet. The best triplet is characterized with the largest number of observed conflicts used to obtain the consistent  $Q_c$  estimate that does not experience any trend (increasing or decreasing) in multiple estimates of this quantity obtained by each time gradually reducing threshold  $T_c$  while keeping the other thresholds at their best values. The following formula is applied to encounters to obtain the initially calculated  $Q_c$  estimate (Tarko, 2020):

$$Q_c = n \cdot 2^{-k}$$

where  $n$  is the number of observed events (encounters and crashes) and  $k$  is calculated with equation (Tarko, 2023):

$$k = \frac{n - c}{\sum_{i=1}^{n-c} \ln(1 + T_i/T_e) + c \ln(2)}$$

where:

$n$  = number of encounters and crashes,

$c$  = number of crashes,

$T_e$  = maximum time to collision used to detect encounters,

$T_i$  = observed time to collision in encounter  $i = 1..n - c$ .

#### A.4.2 Determining Traffic Conflicts

Traffic encounters should be grouped based on the types of corresponding collisions and with consideration of the number of observed and available events. Clustering more cases may be required if the number of events in already formed groups are too small.

Values of the conflicts criteria must be determined for each group of traffic conflicts by analyzing encounters detected based on their own encounter criteria listed in Table A.5. The traffic conflicts criteria must be determined based on crash estimation results and their trends using the encounter data.

The shorter is the  $T$  during an encounter, the more plausible is traffic conflict caused by an error. It prompts using  $T_c$  threshold as short as practical. On the other hand, short threshold  $T_c$  may lead to ignoring real traffic conflicts with longer  $T$  values. Although this omission does not lead to an incorrect estimation of the expected number of crashes, the opportunity to improve estimation confidence by using a greater number of traffic conflicts is not fully utilized. Thus, the choice of  $T_c$  threshold should be done carefully.

Figure A.6 presents a typical sequence of estimated expected number of crashes resulted from applying a sequence of different tried thresholds  $T_t$  to the same set of observed encounters. The profile of the crash estimates includes a portion that is sloped and a portion that is flat. The attempt is to determine a breakpoint between the two portions which is the searched for  $T_c$ .

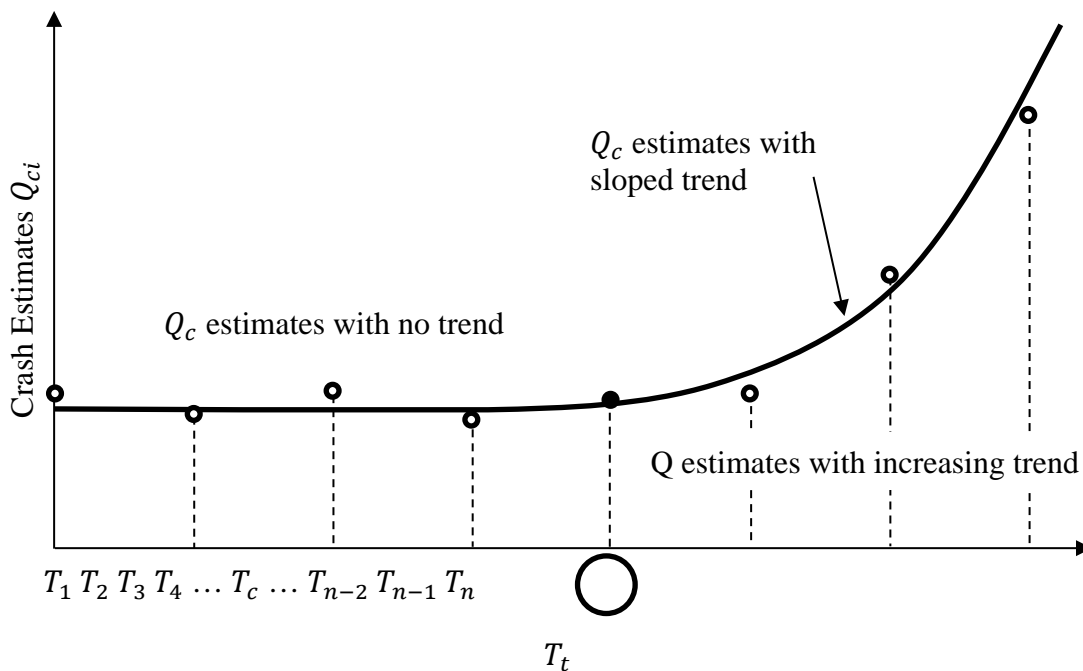


Figure A.6 Profile of crash estimates obtained for a sequence of tried  $T_t$  thresholds; the identified breakpoint is  $T_c$

Let us consider  $n$  estimates of expected number of crashes  $Q_c$  obtained for threshold separations  $T_t$ ,  $t = 1..n$ , and currently tried jerk and impact speed thresholds:  $B_t, S_t, \Delta_t$  (Figure A.6). The task is to determine the longest separation threshold  $T_t$  that belongs to the sequence of estimates exhibiting no

trend. If the flat portion of the profile is confirmed, then the current threshold  $T_t$  is set at  $T_c$  as it produces unbiased  $Q_c$  estimates under the assumed other thresholds  $B_t, S_t, \Delta_t$ . The above procedure of finding the  $T_c$  threshold for traffic conflicts under assumed  $(B_t, S_t, \Delta_t)$  triad may be extended to find the optimal  $B_c, S_c$  and  $\Delta_c$  values by starting with  $B_t = B_e, S_t = S_e, \Delta_t = \Delta_e$  and repeating the procedure for gradually tightened thresholds  $B_t, S_t, \Delta_t$ .

Each one-dimensional solution under assumed thresholds  $(B_t, S_t, \Delta_t)$  yields  $T_c$  estimate. Thus, the best solution from all the obtained solutions should be selected as the final one. The final solution is the one that delivers the lowest Akaike measure, and the number of traffic conflicts is at least the minimum (10 has been used successfully in the past). If multiple solutions with similar Akaike values are obtained, the solution with the largest number of conflicts in the flat part of the profile is selected. The proposed method is robust if the tried jerk and speed thresholds exhaustively cover the range of threshold values applied to observe traffic encounters.

Name	Description
$TTC_c$	Conflict maximum $T$
$J_c$	Conflict minimum longitudinal jerk (initial)
$S_c$	Conflict minimum lateral jerk
$\Delta V$	Minimum magnitude of difference between two velocities at collision for conflicts (initial)

Table A.5 Traffic conflicts criteria

In some cases, estimates of the expected number crashes do not stabilize by the time the number of events reach their minimum. This situation may indicate that the conflicts observation period was too short. Another possibility is that the observed conflicts are applied to too dissimilar types of collisions. For example, rear-end and right-angle crashes, and corresponding traffic conflicts, are quite different in many ways. Combining them together may be questionable. There is no formalized knowledge to propose any guidance other than good judgment

## A.5 Analyzing Road Safety with Traffic Conflicts

### A.5.1 Conflict Diagram

The developed encounter engineering applications program allows the end user specify thresholds of traffic events that include only traffic conflicts that are also encounters but much more severe and convertible to equivalent number of crashes. Conflicts are a subset of the encounters with stronger jerks, higher impact speeds, and shorter  $T$  values as defined for traffic conflicts. These events require much longer data collection periods than encounter to make them useful for follow up safety analysis. Expected crashes in the observation period

$$k = \frac{n - c}{\sum_{i=1}^{n-c} \ln(1 + T_i/T_c) + c \ln(2)}$$

$$Q_c = n \cdot 2^{-k}$$

Observing safety-relevant events under threshold  $T_c$  set at zero yields only crashes ( $n = c$ ). Indeed, the equations return:  $k = 0$  and expected number of crashes  $Q_c = c$ .

*Expanding Crash Estimates to Annual Frequencies*

The TScan system can be used continuously at one intersection for up to four days with a 12-hour observation period per day due to the system’s limited battery capacity and human operation requirements. This implies that traffic conflicts can only be observed for a few days at a given location. A reliable method is thus needed to convert the collected conflict data to a standard road safety measurement namely annual crash frequency. Thus, the need for an expansion factor, defined as the ratio of annual expected crash frequency and expected crash frequency for the observation period.

Traffic events at an intersection are classified into three types based on the intersection legs from where two involved vehicles approach the intersection before colliding one with another. Namely, Same-Approach (SA) pairs, Opposite-Approach pairs (OA), Corner Approach (CA) pairs. A typical four-legged intersection would have ten different approach pairs. As shown in the table below.

Table A.6 Approach pairs at a four-legged intersection

<b>Approach 1</b>	<b>Approach 2</b>	<b>Type</b>
W	W	SA
E	E	SA
N	N	SA
S	S	SA
W	E	OA
N	S	OA
W	N	CA
W	S	CA
E	N	CA
E	S	CA

To get the desired expansion factors between short and long periods, the crash probability would be analyzed at a very disaggregated level. According to current safety practice and the resolution of potential data at their sources, an hour is a proper unit for estimating crash probability and calculating the expansion factors afterward. Considering the limited traffic conflict data collection time by TScan every day, an hourly level analysis could allow more flexibility in the final expansion factors calculation.

The probability that a crash happens during an observation time interval (hour) is expected to be modeled with various explanatory variables regarding traffic, speed, road characteristics, and weather conditions. The total expected crash frequency is then the sum of the predicted crash frequency (one multiplied by the estimated crash probability) for each hourly observation categorized by each type of approach pair.

To calculate expansion factors, input datasets must be assembled for target intersections. The data processing and assembling are similar to preparing sample for modeling (Sections D.3 and D.4). The total number of records  $n$  in this dataset is  $24 \cdot 365 \cdot m$  where  $m$  is the number of approach pairs at the target intersection. A typical four-legged intersection would have 87600 observations. Using these observations and the logistic regression models developed for each type  $k$  of approach movement pairs (Section D.5) the corresponding expansion factors from a short period  $H$  with  $h$ -indexed hourly intervals to year  $Y$  with  $y$ -indexed hourly intervals are calculated with the following equation:

$$EF_{ATY} = \frac{\sum_{h \in \{Y\}} P_h}{\sum_{h \in \{T\}} P_h}$$

where:

$EF_{ATY}$  = crash expansion factor for approach pair  $A$  from short period  $T$  to annual period  $Y$ ,

$\{Y\}$  = set of hour indices in one year,

$\{T\}$  = set of hour indices in short period  $T$  that may include several disjoint subperiods,

$P_h$  = probability of crash hour  $h$ .

#### *Splitting Annual Crash Estimates by Severity*

Once the annual number of crashes are estimated, they can be split by severity level of their outcome by applying existing equations in the current safety management system. Indiana safety management system maintains safety performance functions  $SPF_s$  for three levels of crash severity: fatal and incapacitating injuries  $SPF_{KA}$ , non-incapacitating and possible injuries  $SPF_{NI}$ , and property-damage-only  $SPF_{PD}$ . Thus, the obtained conflict-based annual crash estimates and aggregated to the entire intersection may be split in proportion to the values obtained with the existing safety performance crashes.

#### *Using the Results in Safety Management*

At this point of safety analysis, the results obtained with analyzing traffic conflicts have the format of traditional input to the existing Indiana safety management process. Thus, the exiting components of that process including the Indiana crash modification factors for safety countermeasures and the lifecycle benefit-cost analysis may be used.

### **A.6 References**

Tarko, A. (2020). *Measuring road safety with surrogate events* (1st ed.). Elsevier.

Tarko, A. P. (2023). Maximum likelihood method of estimating the conflict-crash relationship. *Accident Analysis and Prevention*, 179, 106875.

Tarko, A. P., & Lizarazo, C. G. (2021). Validity of failure-caused traffic conflicts as surrogates of rear-end collisions in naturalistic driving studies. *Accident Analysis and Prevention*, 149, 105863.

## **APPENDIX B. TSCAN ENGINEERING APPLICATIONS MANUAL**

### **B.1 Preface**

The TScan project aims to develop a data processing module for a novel LiDAR-based traffic scanner to collect highly accurate microscopic traffic data at road intersections. TScan uses light detection and ranging (LiDAR) technology that can detect and track various types of road users including buses, cars, pedestrians, and bicycles and, unlike video detection, LiDAR data has one-to-one correspondence with the physical world. Hence, it is possible in principle to produce the positions and velocities of road users in real-time as needed for traffic and safety applications, with errors of estimation dependent only on the resolution and accuracy of the LiDAR sensor.

The Engineering Application Program include trajectory visualizer, counting vehicles, display pedestrian occupancy, evaluate vehicle's speed in the intersection area, and evaluate traffic encounters. This toolset was developed by the Center for Road Safety. The User Manual includes the user interface to upload information from both the initial setup and the data collection, to enter the required information for the different types of analysis and for displaying trajectory files generated at a given intersection. The TScan Engineering Application toolset was developed as part of the Stationary LiDAR for Traffic and Safety Applications–Vehicles Interpretation and Tracking (TScan) project at the request of the Indiana Department of Transportation, and its development was supported through the Joint Transportation Research Program of Purdue University and the Indiana Department of Transportation.



## B.2 TScan Engineering Applications Overview

TScan engineering applications are some applications developed to use TScan results.

Figure B.1 shows TScan framework in which TScan Engineering Applications is included.

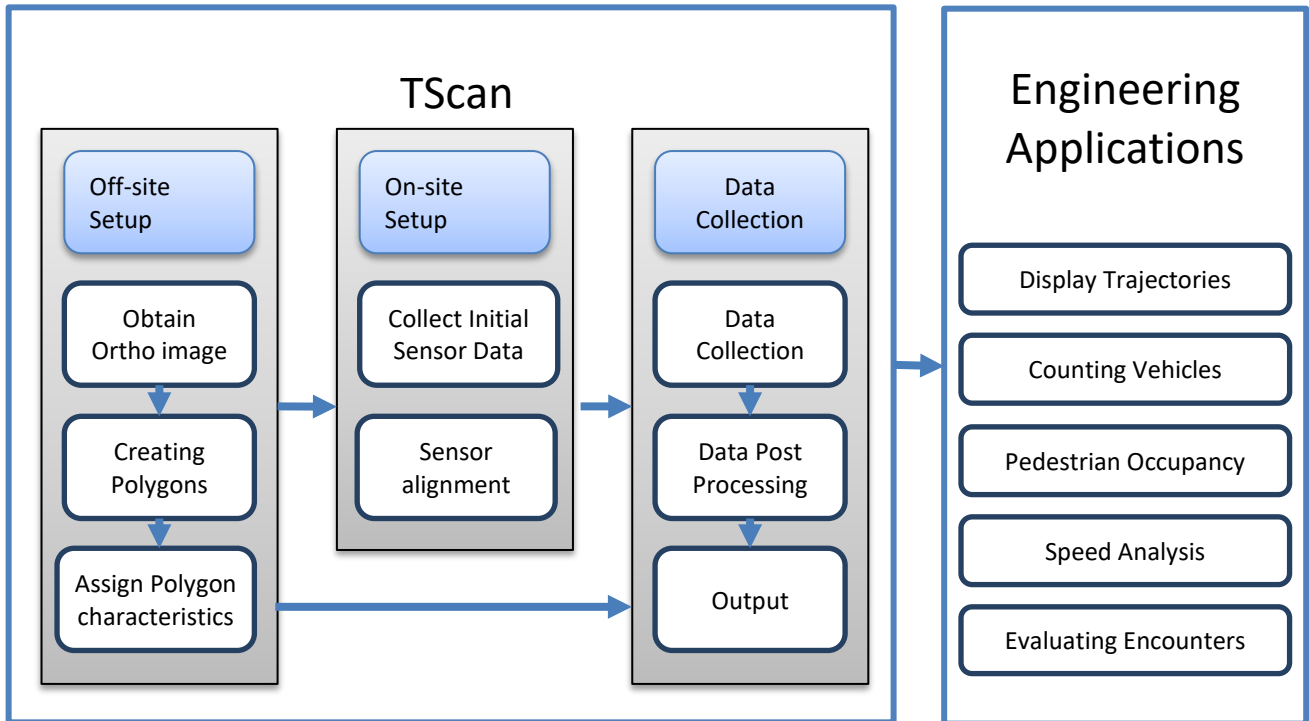


Figure B.1 TScan overview.

TScan Engineering Applications toolset is a computer application that allows the user to display vehicle's trajectories, counting turning movements, display pedestrian occupancy, evaluate vehicle's speeds in the intersection area, and evaluate traffic Encounters based on TScan data collection output.

## B.3 Installation

TScan Engineering Applications is compatible with Windows 10. In order to run TScan Engineering Applications, the MS .NET 4.5.2 Framework or later component must be installed.

If the MS .NET 4.5.2 Framework is not present during the installation, TScan Engineering Applications will attempt to install this component if the PC is connected to the Internet.

To install the TScan Engineering Applications interface, follow the steps given below.

1. Extract the contents of the archived file to your local drive.
2. The installation process is initiated by clicking on the *setup.exe* file.
3. Step by step instructions that explains the installation process is found in *readme.exe*.

*Note:* The user should always read the *readme.txt* file included in the installation package which includes the most up-to-date installation instructions.

After checking that the program is working, the user may delete the unzipped files in the folder with the *setup.exe* file to save disk space. The user should save the zipped/compressed file in case it is needed to reinstall the program.

## **B.4 Launching TScan Engineering Applications**

The TScan Engineering Applications program can be launched using any of the following methods after installation:

### Method 1

1. Double click the shortcut on the desktop.

### Method 2

1. Press *Start* button.
2. TScan Engineering Applications should appear in the list of installed programs.
3. Single click on the shortcut.

### Method 3

1. Press *Start* button.
2. Start typing '*TScan Engineering Applications*', the program shortcut should appear in the search results
3. Single click on the shortcut

### Method 4

1. Open *My Computer*.
2. Browse to the location where the software was installed. Typically, '*C:\Program Files (x86)\TScan Engineering Applications /*'.
3. Click on *TScan Engineering Applications.exe*.

The main interface appears within a few seconds (Figure B.2).

## B.5 TScan Engineering Applications Interface

The TScan Engineering Applications interface window is filled with interactive panels. The layout of the interface window can be changed by resizing and moving the panels. For instance, double clicking on the top bar of a panel, that panel becomes a floating object. To dock a panel, click and hold a panel's title bar while dragging it, and the interface will show where it can be docked.

Clicking on the vertical pin in the upper right corner of a panel closes the panel. The closed panel is represented with a tab placed along the vertical border of the interface window. Hovering the cursor above the tab opens the panel as long as the cursor is placed over the tab. Moving the cursor over the horizontal pin in the upper right corner of the open panel and clicking on the pin make the panel open permanently.

The main interface includes a command bar in the first row and status bar in the bottom (Figure B.2). The command bar facilitates operations on files and selection of the units. The floating windows allows user to select among the different applications.

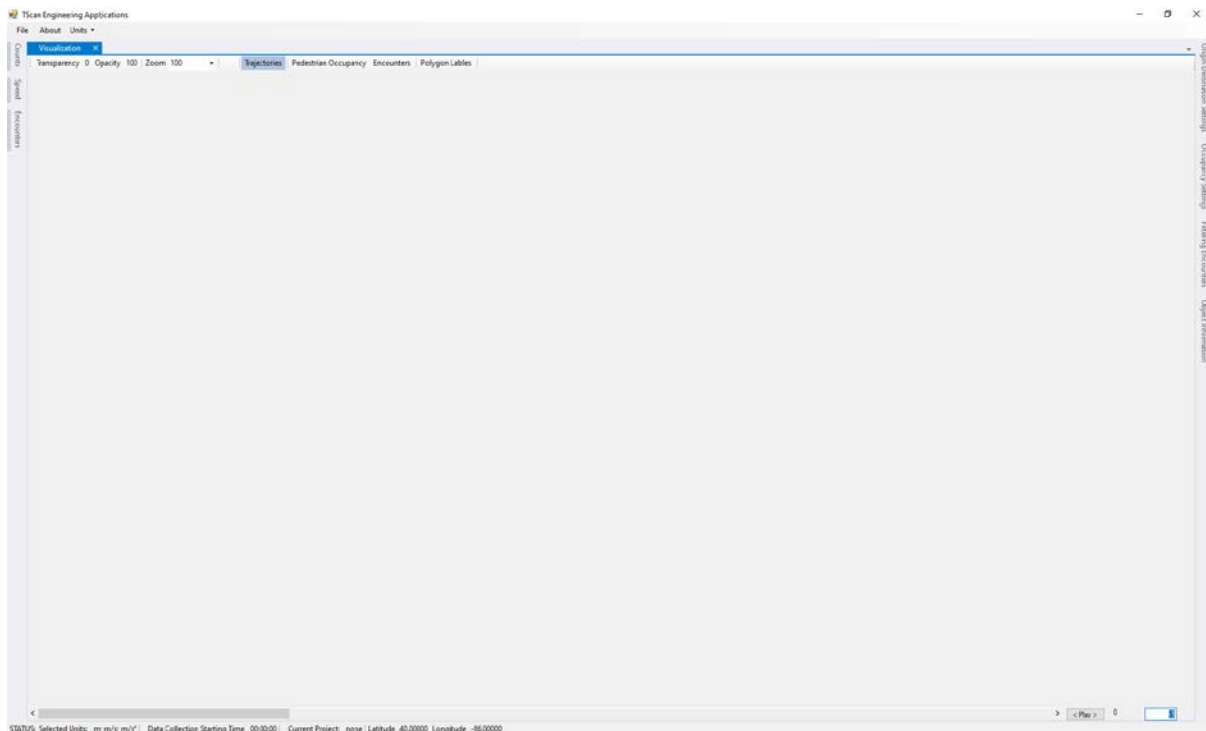


Figure B.2 TScan engineering applications main interface.

The *Visualization* window displays the detected objects trajectories, the Pedestrian Occupancy Rate and the Traffic Encounters Diagram. The *O-D Settings* tab allow user to create vehicle's movements groups, input the origin-destination matrix for directional counting, speeds and encounter detection. The *Counts* window and the *Speed* tab allows users to select intervals, starting time and to display and save the results. The *Occupancy Settings* window is used to select the graphic parameters to represent the pedestrian occupancy rates. The *Encounters* and the *Filtering Encounters* window allow user to enter the detection, and filtering parameters, and to display the Encounter results. The Object Information window shows the vehicle's parameters such as size, location, speed, acceleration. The following chapters describe each activity in detail.

## Upload Data

Even TScan generates a set of files that are required for the engineering applications, all files are managed by the collection settings file <Location Name>.dcs. All files should be kept in the same folder to maintain the data integrity.

To open the data collection settings file obtained after post processing the data, click on the *Open* button to open the Windows file selection window shown in Figure B.3, where the appropriate file can be selected. Once selected click on the *Open* button to upload the image and the corresponding data. The orthographic image is uploaded automatically since its location is saved on the dcs file.

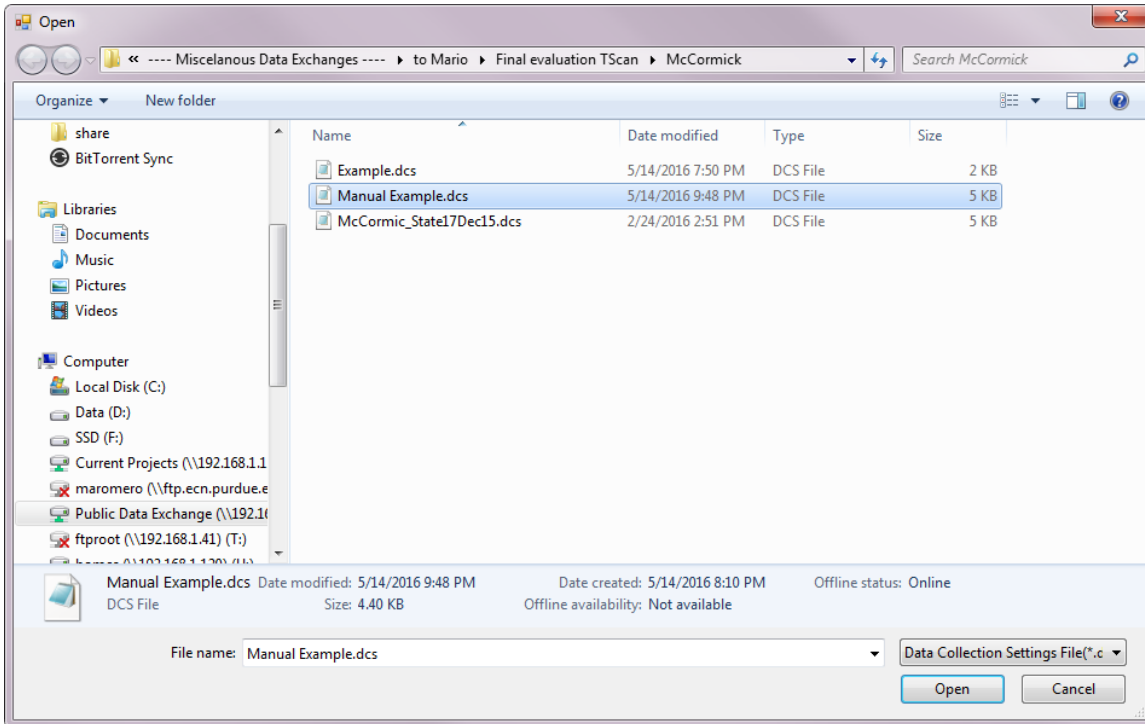


Figure B.3 Upload data collection settings file.

## B.6 Trajectory Visualizer

Select the Trajectory Visualization mode (1) to graphically inspect the TScan tracking results frame by frame or by selecting a vehicle for detail inspection. The intersection layout is shown, along with the moving objects (vehicles, bicycles, or pedestrians) which are represented as a blue rectangle (Figure B.4).

The visualization is done frame by frame thus can be presented as a video using the *play*, *pause* button (2). A particular frame can be selected either by using the scroll bar at the bottom or by typing the desired frame number (3). Also, the zoom level of the image can be adjusted by selecting it in the dropdown list (4), as well as the distance, speed, and acceleration units (5). Finally, a particular vehicle can be selected for a more detailed analysis by clicking on the vehicle's box or by typing in the vehicle ID (6) (Figure B.4).

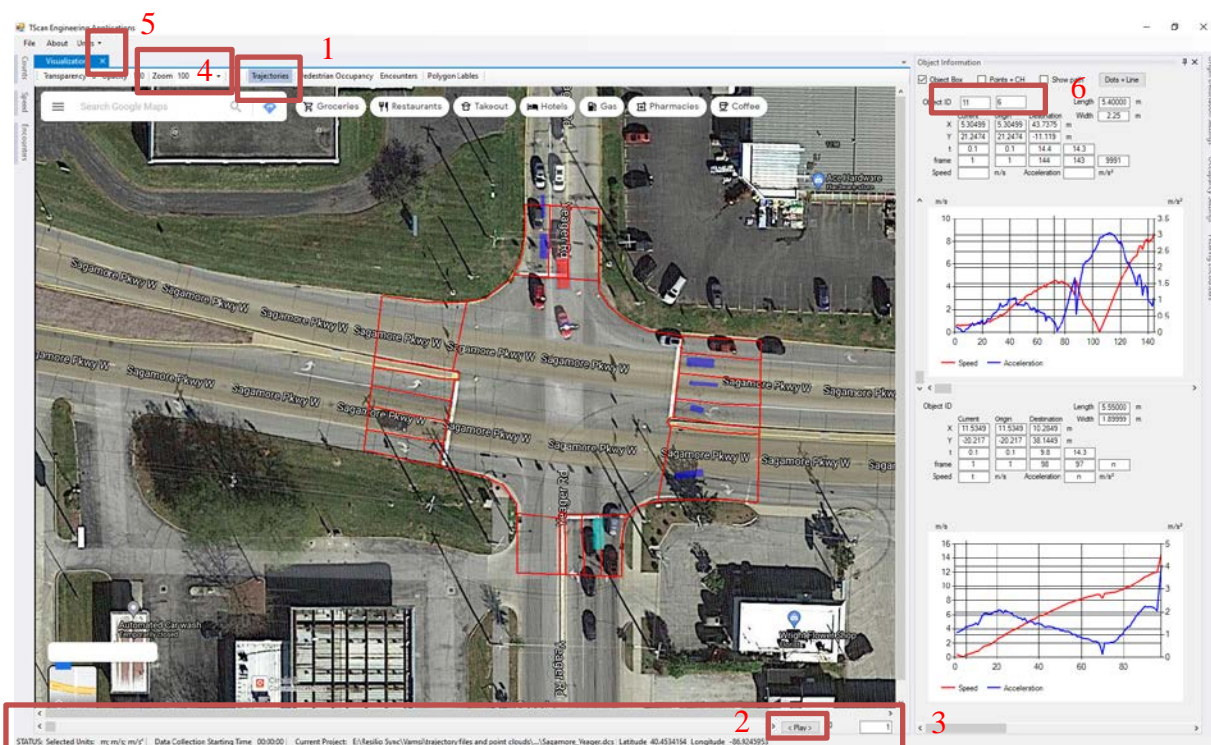


Figure B.4 Visualization and object information windows.

Once a vehicle is selected, its location, speed and acceleration in the current frame is displayed (1) as well as a chart that represents its entire trajectory (2). The representation of the objects can be

changed to show the LiDAR points that compose it, its bounding box, and its path by selecting the corresponding checkboxes (3) (Figure B.5)

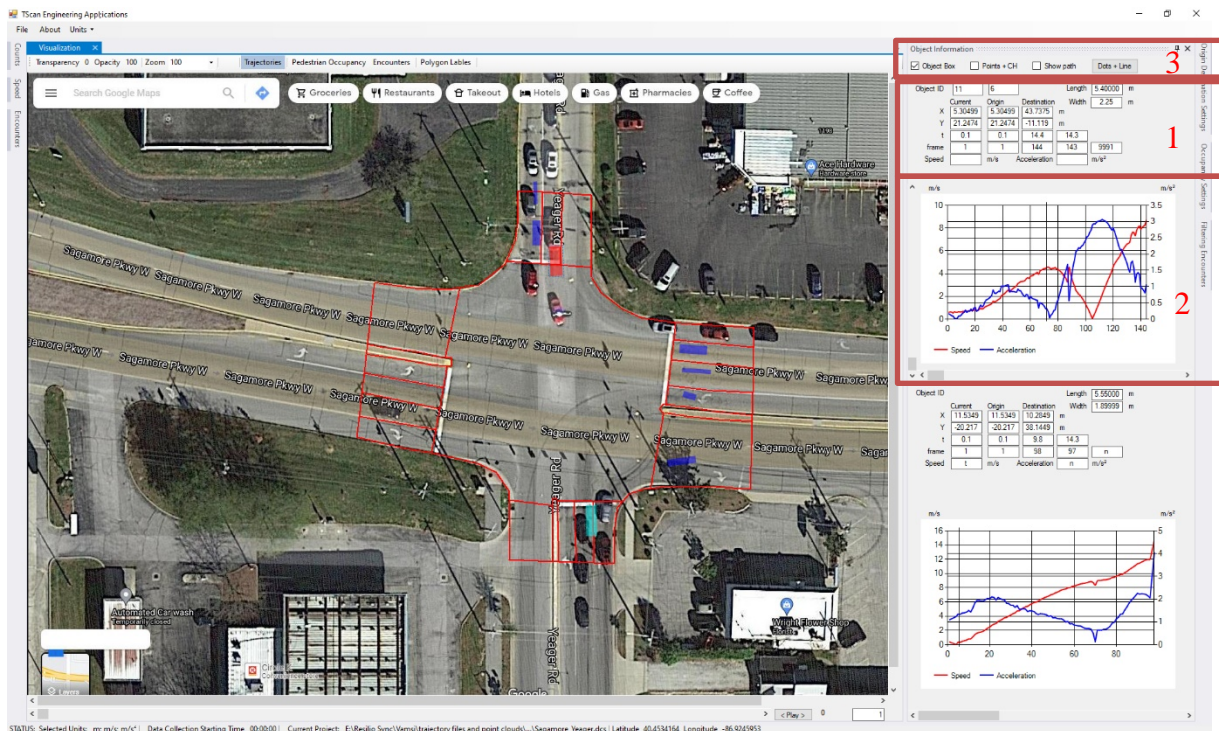


Figure B.5 Vehicle inspection.

## B.7 Movements Setup

Both the Counting Vehicles and the Speed Analysis Module requires the user to review the uploaded data, select the time interval, set the starting time, and create the origin-destination matrix to define the movements, based on the created reference polygons of the intersection.

Each of the steps are explained in further detail below.

### *Review Uploaded Data*

Once the data collection settings file is open, the data collections starting time is displayed. The list of polygons IDs of the intersection area are set as well as the lists of polygons arranged by polygon type. The background image is displayed. The polygons IDs and the polygons edges are also displayed on the image. Review the uploaded values and images to check the intersection setup.

### *Creating Origin-Destination Matrix*

To count turning movements it is necessary to set the origin approaches and the destination areas by assigning polygons to each origin and destination. One single origin or destination can include several polygons. The require steps are explained below.

### Creating Origin Approaches

To create an origin approach. Select the current origin textbox. Enter the origin name and click on *Add* button. This action will create an origin place holder and selects it as shown in Figure B.6. Then add the corresponding polygons to the selected origin

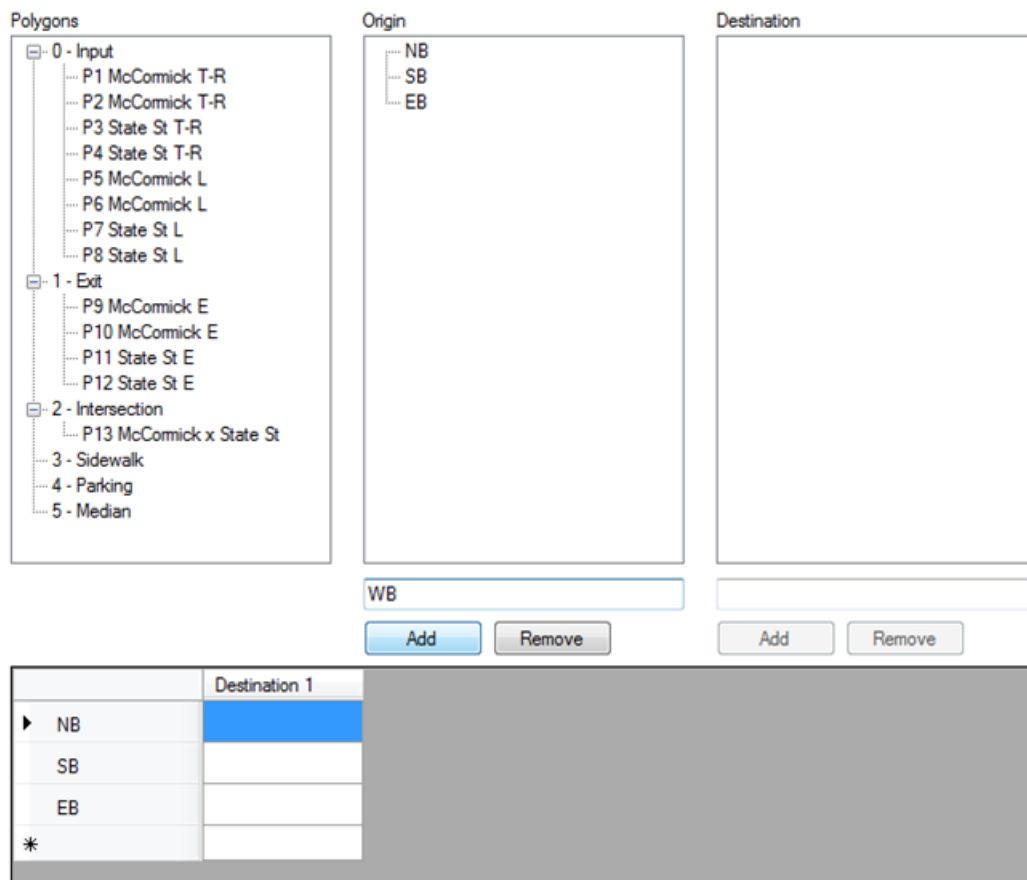


Figure B.6 Creating origin approaches.

### Creating a Destination Area

To create a destination area. Select the current destination textbox. Enter the destination name and click on *Add* button. This action will create a destination area place holder and it becomes the selected destination as shown in Figure B.7. Then add the corresponding polygons to the selected destination.

**Polygons**

- [-] 0 - Input
  - P1 McCormick T-R
  - P2 McCormick T-R
  - P3 State St T-R
  - P4 State St T-R
  - P5 McCormick L
  - P6 McCormick L
  - P7 State St L
  - P8 State St L
- [-] 1 - Exit
  - P9 McCormick E
  - P10 McCormick E
  - P11 State St E
  - P12 State St E
- [-] 2 - Intersection
  - P13 McCormick x State St
- 3 - Sidewalk
- 4 - Parking
- 5 - Median

**Origin**

- [-] NB
- [-] SB
- [-] EB
- [-] WB

**Destination**

- [-] NB E
- [-] SB E
- [-] EB E

WB      WB E

Add    Remove      Add    Remove

	NB E	SB E	EB E	
▶ NB				
SB				↖
EB				
WB				
*				

Figure B.7 Creating destination areas.



*Assigning Polygons to an Origin Approach or to a Destination Area*

To assign a polygon to a selected origin or destination, select the origin/destination on the corresponding list by clicking on the name. The origin/destination name will be set as the selected one. Then double-click on the polygon list to add the polygon as shown in Figure B.8.

The interface consists of three main panels: **Polygons**, **Origin**, and **Destination**.

- Polygons:** A list of polygon categories and their associated polygons.
  - 0 - Input: P1 McCormick T-R, P2 McCormick T-R, P3 State St T-R, P4 State St T-R, P5 McCormick L, P6 McCormick L, P7 State St L, P8 State St L
  - 1 - Exit: P9 McCormick E, P10 McCormick E, P11 State St E, P12 State St E
  - 2 - Intersection: P13 McCormick x State St
  - 3 - Sidewalk
  - 4 - Parking
  - 5 - Median
- Origin:** A list of origin approaches.
  - NB: P1 McCormick T-R, P5 McCormick L
  - SB: P2 McCormick T-R, P6 McCormick L
  - EB: P4 State St T-R, P8 State St L
  - WB: P3 State St T-R, P7 State St L
- Destination:** A list of destination areas.
  - NB E: P10 McCormick E
  - SB E: P9 McCormick E
  - EB E: P12 State St E
  - WB E: (highlighted)

Below the Origin and Destination panels are input fields and buttons:

- Origin input field: (empty)
- Origin buttons: Add, Remove
- Destination input field: WB E
- Destination buttons: Add, Remove

At the bottom is a table with the following structure:

	NB E	SB E	EB E	WB E	
▶ NB					
SB					
EB					
WB					
*					

Figure B.8 Assigning polygons to an origin approach or to a destination area.

*Removing Polygons from an Origin Approach or a Destination Area*

To remove a polygon from an origin approach or a destination area, select the polygon on the area to be removed and click on *Remove* button as shown in Figure B.9.

The interface consists of three tree views: **Polygons**, **Origin**, and **Destination**.

- Polygons:**
  - 0 - Input
    - P1 McCormick T-R
    - P2 McCormick T-R
    - P3 State St T-R
    - P4 State St T-R
    - P5 McCormick L
    - P6 McCormick L
    - P7 State St L
    - P8 State St L
  - 1 - Exit
    - P9 McCormick E
    - P10 McCormick E
    - P11 State St E
    - P12 State St E
  - 2 - Intersection
    - P13 McCormick x State St
  - 3 - Sidewalk
  - 4 - Parking
  - 5 - Median
- Origin:**
  - NB
    - P1 McCormick T-R
    - P5 McCormick L
  - SB
    - P2 McCormick T-R
    - P6 McCormick L
  - EB
    - P4 State St T-R
    - P8 State St L
  - WB
    - P3 State St T-R
    - P7 State St L
- Destination:**
  - NB E
    - P10 McCormick E
  - SB E
    - P9 McCormick E
  - EB E
    - P12 State St E
  - WB E
    - P11 State St E
    - P13 McCormick x State St** (Selected)

Below the Destination tree, a text box contains "P13 McCormick x State St". Below this are two pairs of "Add" and "Remove" buttons.

	NB E	SB E	EB E	WB E	
▶ NB					
SB					
EB					
WB					
*					

Figure B.9 Removing polygons from an origin approach or a destination area.

*Removing an Origin Approach or a Destination Area*

To remove an origin approach or a destination area, select the origin approach or the destination area to be removed and click on *Remove* button as shown in Figure B.10.

The interface consists of three main panels: **Polygons**, **Origin**, and **Destination**. Each panel contains a list of items with expandable/collapsible icons. Below each panel is a text input field and two buttons: **Add** and **Remove**.

- Polygons:**
  - 0 - Input
    - P1 McCormick T-R
    - P2 McCormick T-R
    - P3 State St T-R
    - P4 State St T-R
    - P5 McCormick L
    - P6 McCormick L
    - P7 State St L
    - P8 State St L
  - 1 - Exit
    - P9 McCormick E
    - P10 McCormick E
    - P11 State St E
    - P12 State St E
  - 2 - Intersection
    - P13 McCormick x State St
  - 3 - Sidewalk
  - 4 - Parking
  - 5 - Median
- Origin:**
  - NB
    - P1 McCormick T-R
    - P5 McCormick L
  - SB
    - P2 McCormick T-R
    - P6 McCormick L
  - EB
    - P4 State St T-R
    - P8 State St L
  - WB
    - P3 State St T-R
    - P7 State St L
- Destination:**
  - NB E
    - P10 McCormick E
  - SB E
    - P9 McCormick E
  - EB E
    - P12 State St E
  - WB E (highlighted in blue)
    - P11 State St E
    - P13 McCormick x State St

Below the panels are two sets of controls. The first set has an empty text field and **Add** and **Remove** buttons. The second set has a text field containing "WB E" and **Add** and **Remove** buttons.

	NB E	SB E	EB E	WB E	
▶ NB					
SB					
EB					
WB					
*					

Figure B.10 Removing an origin approach or a destination area.

### Setting Movement Names

Once the origin approaches and the destination areas are set it is necessary to name the movements corresponding to the origin destination pairs as shown in Figure B.11. Only the named movements are counted in the counting process.

**Polygons**

- 0 - Input
  - P1 McCormick T-R
  - P2 McCormick T-R
  - P3 State St T-R
  - P4 State St T-R
  - P5 McCormick L
  - P6 McCormick L
  - P7 State St L
  - P8 State St L
- 1 - Exit
  - P9 McCormick E
  - P10 McCormick E
  - P11 State St E
  - P12 State St E
- 2 - Intersection
  - P13 McCormick x State St
- 3 - Sidewalk
- 4 - Parking
- 5 - Median

**Origin**

- NB
  - P1 McCormick T-R
  - P5 McCormick L
- SB
  - P2 McCormick T-R
  - P6 McCormick L
- EB
  - P4 State St T-R
  - P8 State St L
- WB
  - P3 State St T-R
  - P7 State St L

**Destination**

- NB E
  - P10 McCormick E
- SB E
  - P9 McCormick E
- EB E
  - P12 State St E
- WB E
  - P11 State St E

	NB E	SB E	EB E	WB E	
NB	NB-T		NB-R	NB-L	
SB		SB-T	SB-L	SB-R	
EB	EB-L	EB-R	EB-T		
▶ WB	WB-R	WB-L		WB-T	
*					

Figure B.11 Setting movement names.

## B.8 Counting Vehicles

The Counting Vehicles Module allow the user to obtain directional counting statistics based on the TScan post processing output files using user defined counting periods. Once the counting interval is selected, the counting starting time is set and the origin-destination matrix is created the counting process can be initiated.

### B.8.1 Set the Starting Time

The starting time is set by default as the data collection starting time. To change the starting time for counting or for speed analysis, enter the appropriated time on the starting time text box as shown in Figure B.12.

Vehicles entering the intersection polygon before the starting time are not considered in the counts or in the speed analysis.

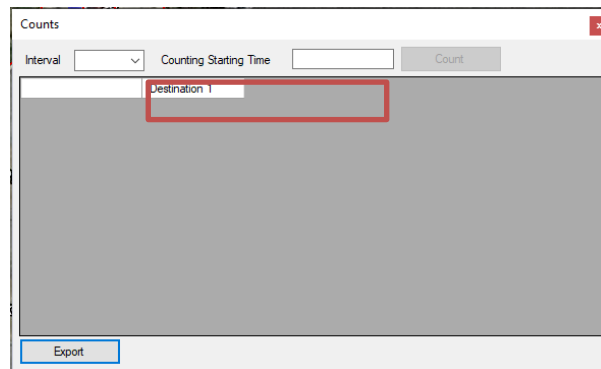


Figure B.12 Setting the starting time.

### B.8.2 Select Time Interval

The selected time interval governs both the counting and the speed analysis aggregation level. Select the interval from the combo-box as shown in Figure B.13. The interval options are 1, 3, 5, 10, 15, 20, 30, and 60 minutes.

Note: Only full intervals are included in the results i.e., if the data set's time period is not a multiple of the selected interval, then the last portion is ignored.

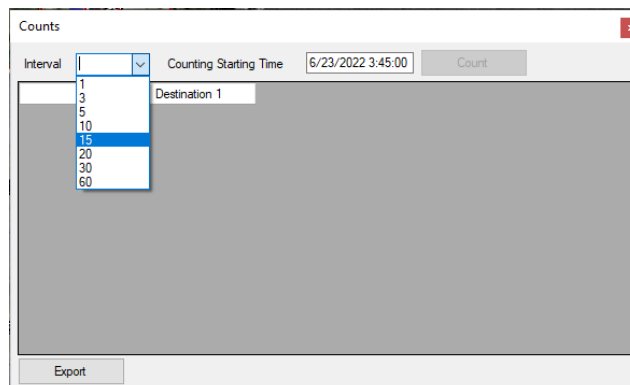


Figure B.13 Interval selection.

### B.8.3 Counting Process

Click on the *Counting* button to initiate the counting process. Once completed the *Statistics* tab is automatically selected to display results as shown in in Figure B.14.

	NB-T	NB-R	NB-L	SB-T	SB-L	SB-R	EB-L	EB-R	EB-T	WB-R	WB-L	WB-T
10:35:00 AM	4		1	11	1	6	10		1		2	
10:40:00 AM	12	1	1	18	4	8	15		7		4	1
10:45:00 AM	7		1	32	2	3	17	1	6		8	
10:50:00 AM	8	2	1	14		7	19		5		2	
10:55:00 AM	10		4	10	1	6	14		10		5	1
11:00:00 AM	17		4	19	1	1	21		4		11	2
11:05:00 AM	8	1	1	16	3	7	26		2		5	
11:10:00 AM	13	1	5	22	4	5	17	1	6	1	10	1
11:15:00 AM	5		1	10		2	8		3			
11:20:00 AM	1			6	1	3	1					
11:25:00 AM	1		2	3	3		2		1		2	
→*	2			6		3	1					

Export

Figure B.14 Counting results.

The results are organized in a table. Each row corresponds to a time interval and each column correspond to a named movement from de origin-destination matrix.

### B.8.4 Export Results

To save the shown result table into a comma separated values file format, click on the *Export* button. The user is asked to provide a file name and a folder to which the results file is to be saved as shown in Figure B.15.

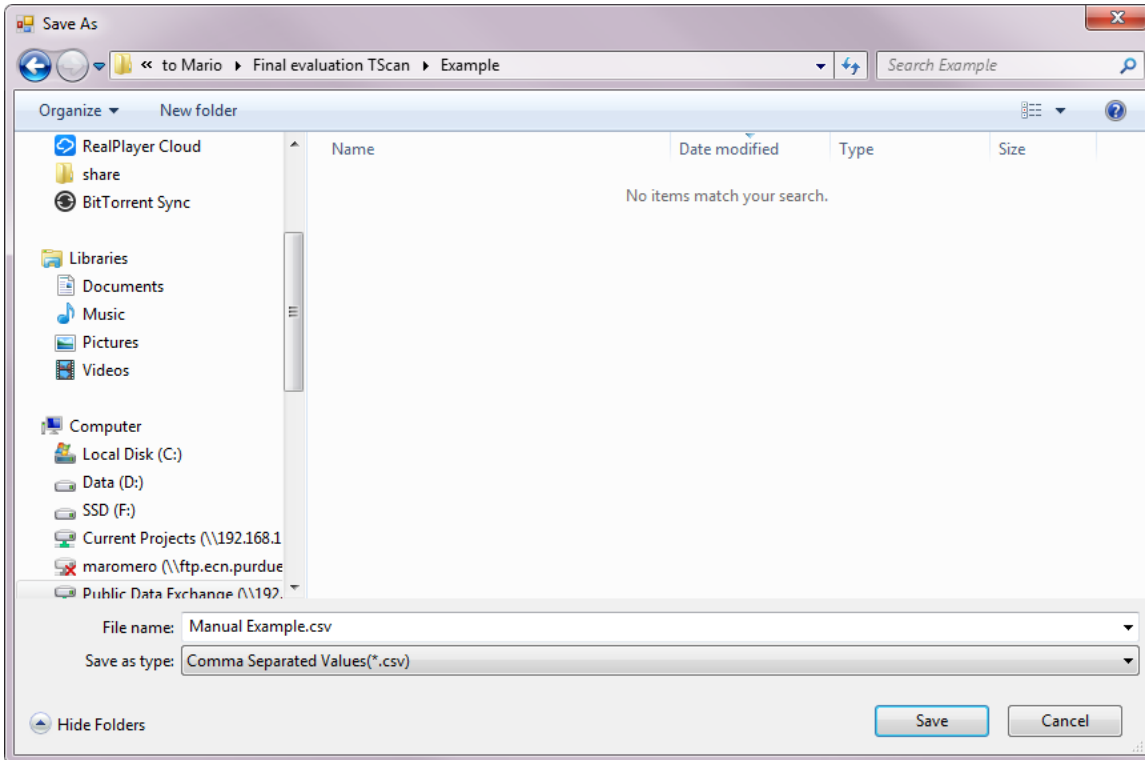


Figure B.15 Export results.

## B.9 Pedestrian Occupancy Rate

The pedestrian occupancy rate graphically presents the time a tyle is occupied by a pedestrian during a given period. The more intense the color, the higher the occupancy rate (Figure B.16).

First, select the Pedestrian Occupancy (1) display mode in the visualization window. Then, to define the aggregation interval, select from a pulldown list for 1, 5, 15, 30 or 60 minutes (2).

The color saturation level can be controlled depending on the percentage of occupancy time. The upper limit determines the point at which the color is fully saturated while the lower limit represents the point at which the saturation is set to zero (3).

Finally, it can be defined if the representation will be on color or on saturation scale and the saturation/color distribution (4).

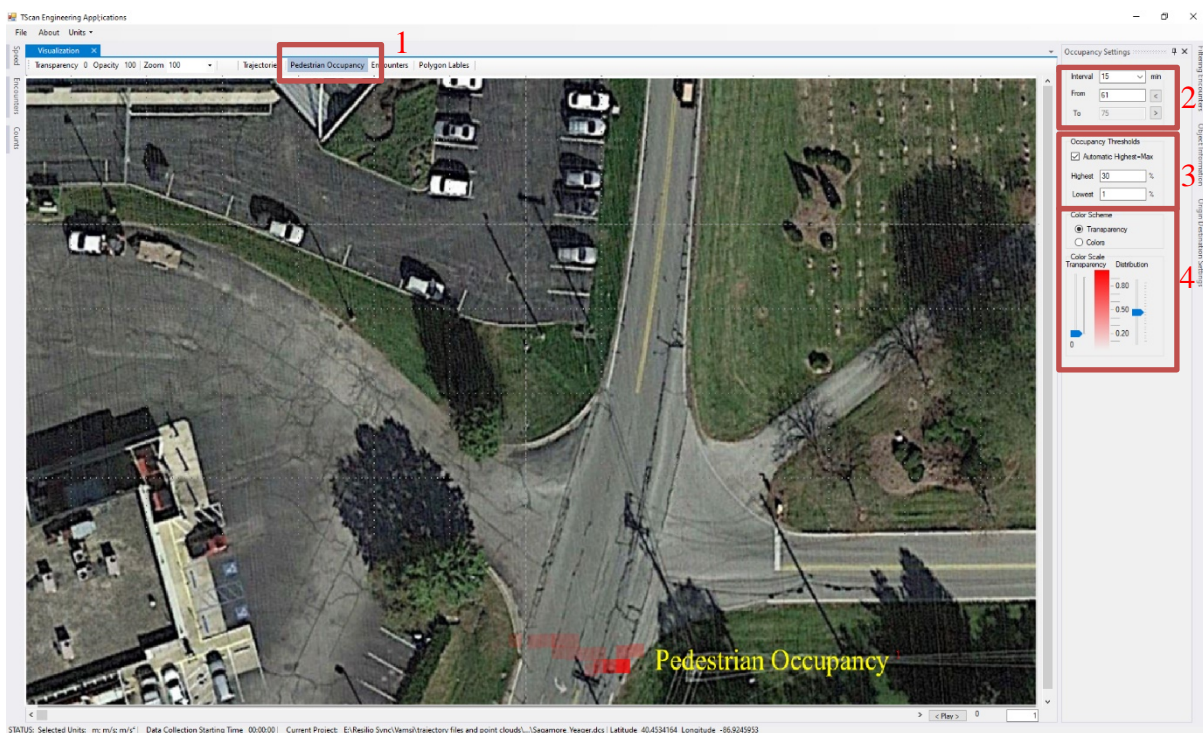


Figure B.16 Pedestrian occupancy rate.

## B.10 Speed Analysis

The objective of the speed analysis is to calculate a user selected percentile of the average speed of the vehicles during their passage through the intersection zone for each of the movements defined in the origin-destination matrix. Information on how to create the origin-destination matrix can be found in section B.7.2

First, enter the speed threshold parameter (1) (Figure B.17). The units of this value are shown after the text box and depends on the selected units in the main command bar. The threshold is used to filter out those vehicles that at some point during their passage through the intersection have circulated at a speed lower than this threshold. It is important to note that this threshold does not apply to the average speed of the vehicle but to the spot speeds during its passage through the intersection zone.



In addition, the time interval (2), the desired speed percentile (3) and the starting time (4) must be selected using the pulldown lists or typing on the corresponding textbox. Once these parameters have been selected, click on the *Calculate* button (5) to obtain the results.

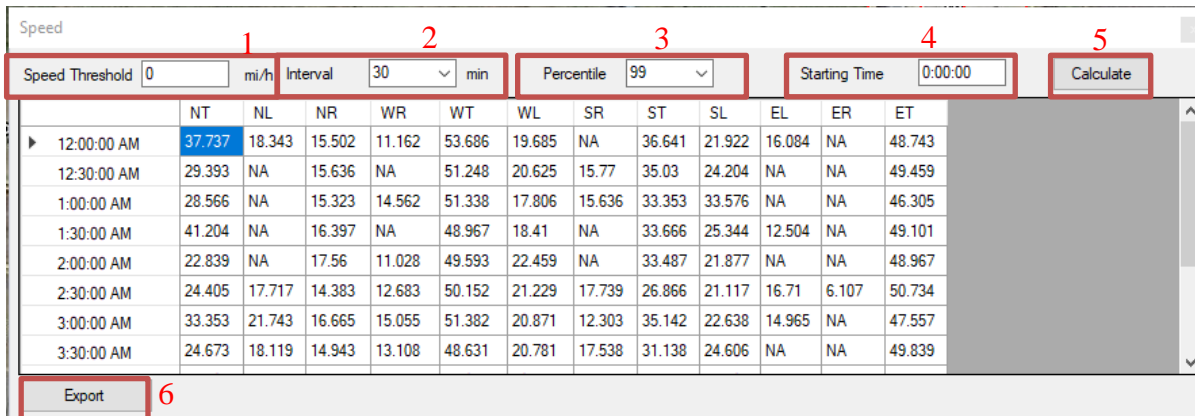


Figure B.17 Speed analysis interface.

If a particular movement doesn't have enough traffic to calculate the speed percentiles, the result value is NA indicating that it's not available.

Finally, click on the *Export* button (6) to save the results as a comma separated values .csv file. The user is asked to provide a file name and a folder to which the results file is to be saved as shown in Figure B.15.

## B.11 Traffic Encounters

TScan post-processing generates a list of interactions between vehicles that could potentially be considered as traffic encounters. The user must filter this list using the parameters that determine whether an action taken by a vehicle is an evasion and therefore it is a potential encounter or not (1) (Figure B.18). Default values are provided but the user can modify them according to their experience.

### B.11.1 Encounter Detection

Figure B.18 shows the encounter detection interface. Once the parameters are set, click on the *Extract* button (2) to obtain the list of encounters. The program first filters the potential encounters according to the parameters selected by the user, then it detects the evasive maneuver and replaces the evasive part of the vehicle's trajectory by an encounter-free path that resembles the one of the vehicles making the evasive maneuver, then, the program checks if a collision happens using the non-evasive maneuver. If so, the time to collision for that interaction is obtained.

The list of detected traffic encounters (3) and the list of road user actions (4) are presented in a grid and can be inspected in the viewer by selecting the Encounter in the grid. Click on the *Export* button (5) to save the results as a comma separated values .csv file. The user is asked to provide a file name and a folder to which the results file is to be saved as shown in Figure B.15.

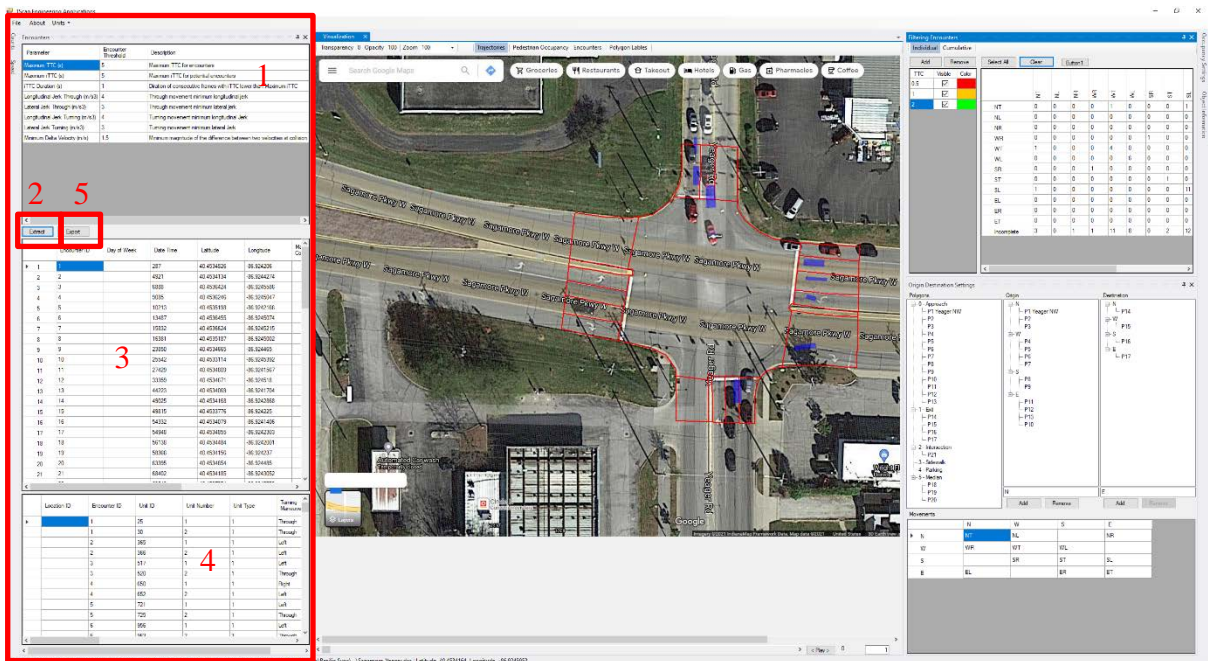


Figure B.18 Traffic encounters interface.

### B.11.2 Encounter Inspection

The encounters with incomplete trajectories due to obstructions are highlighted for further inspection by the user.

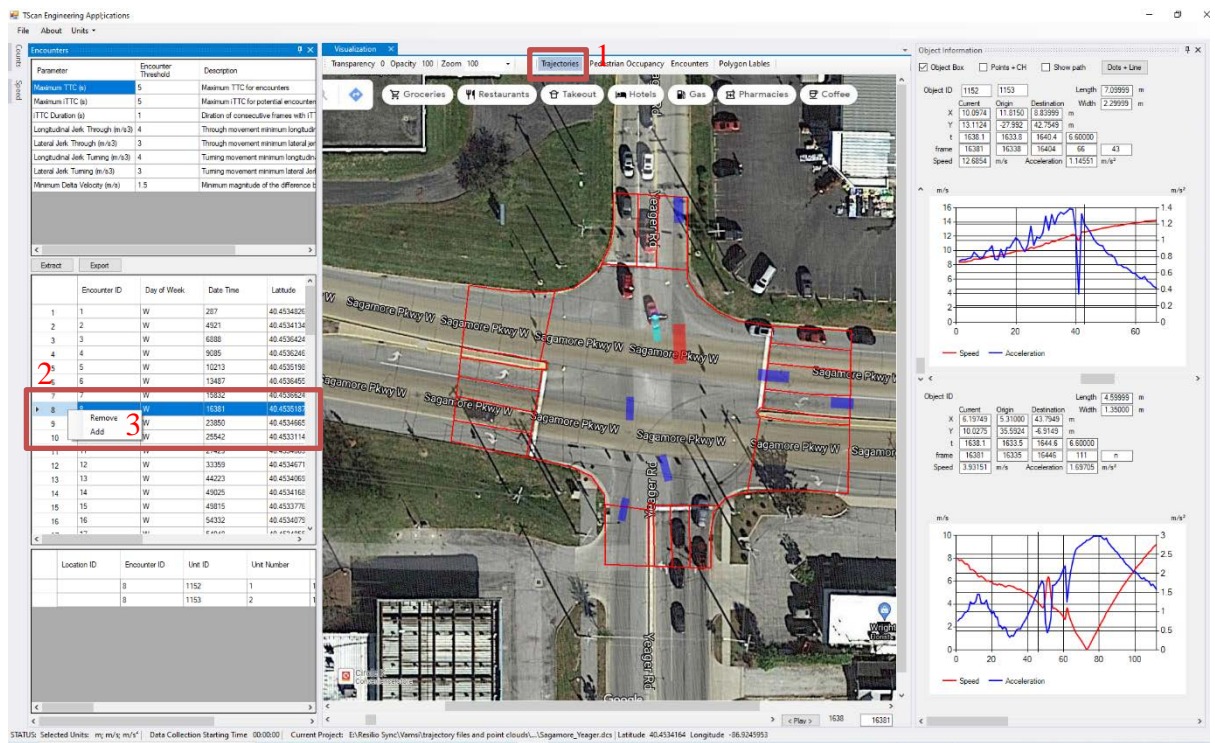


Figure B.19 Inspecting traffic encounters.

To inspect individual encounters, first select the display trajectories mode<sup>1</sup> (Figure B.19). Then, select the encounter to be inspected from the list of detected traffic encounters.<sup>2</sup> The program automatically displays the two vehicles involved in the traffic encounter and at the time instant in which the conflict was detected. If further inspection is needed the user can check the video clip and supplement the missing information on the grid. If the user decide that the encounter is invalid, the encounter can be tagged to remove it for further analysis. To do so, right click on the encounter and select *Remove* on the pop-up menu.<sup>3</sup> User can undo this operation by right clicking on the encounter and select *Add* on the pop-up menu<sup>3</sup>.

### B.11.3 Encounter Diagram

The encounter diagram allows to visualize the locations the hypothetical crash location of the individual encounters or grouped by the vehicles' movement (Figure B.20). Change the display mode to *Encounters*<sup>1</sup> to visualize the encounter diagram. The height of the bars in the *Aggregated*<sup>2</sup> mode is proportional to the number of encounters. Both the individual encounters and the bars are color coded by severity depending on the Time to Collision.<sup>3</sup> To create new Time to Collision categories, click on the *Add* button<sup>3</sup>, then enter the *T* level. The *T* table will be automatically sorted, and the colors will be adjusted accordingly. To remove a particular *T* category, select it and click the *Remove* button.<sup>3</sup>

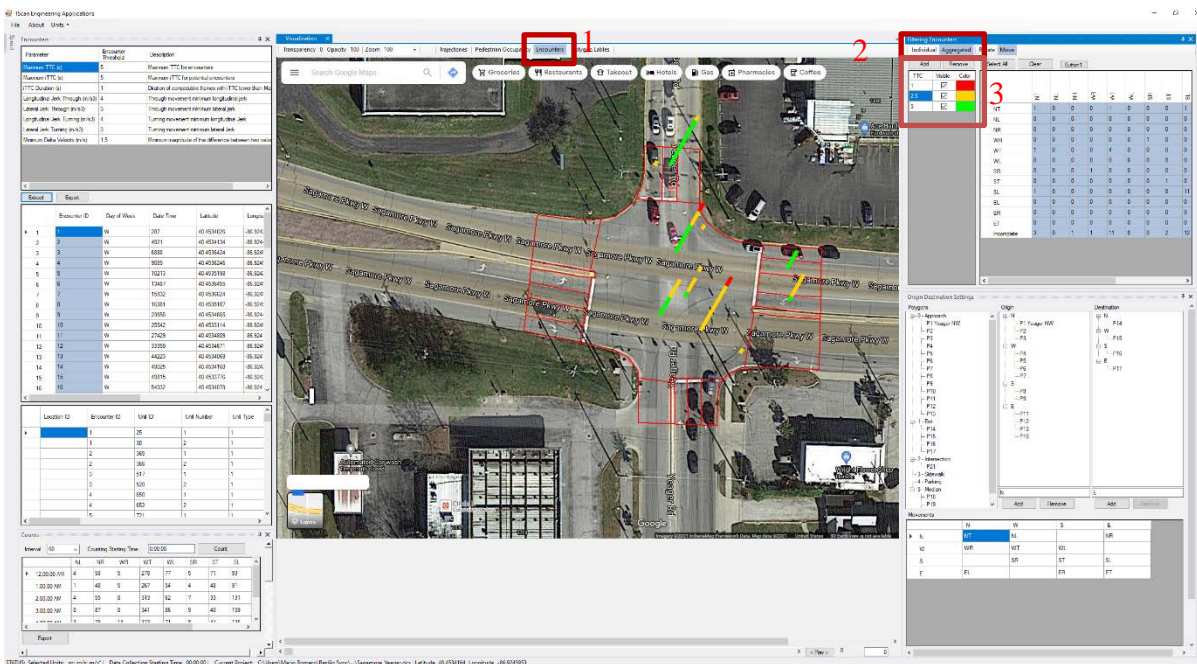


Figure B.20 Traffic encounters diagram.

## Filtering Encounters to Display

To analyze the encounters of a particular movement, the Encounter Diagram can be filtered by selecting in the interaction matrix<sup>1</sup> the entire grid, a particular movement or a single interaction type by selecting a column, or a particular cell in such grid (Figure B.21). Clicking in a row or a column header will toggle the selection of all cells on the row and column. The same way, clicking on a single cell will toggle the selection of that particular cell. Only the highlighted interactions will be displayed.

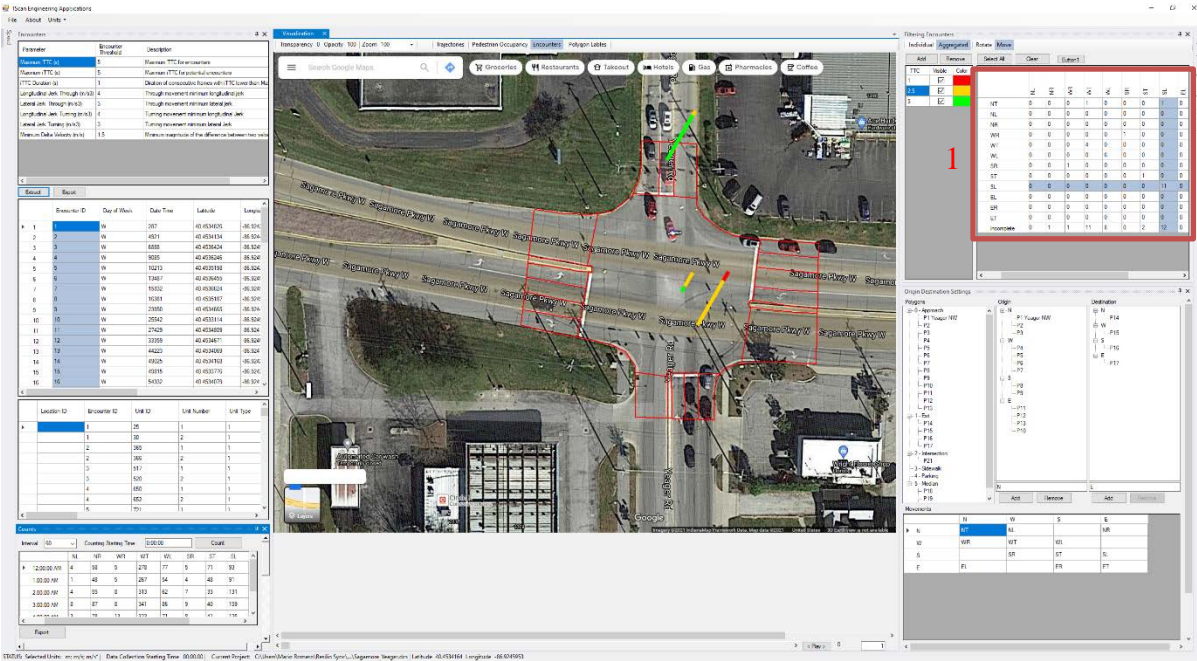


Figure B.21 Filtering encounters to display.

## Moving Encounter Bar

The bars are placed by default on the intersection of the two different trajectories involved in the encounter. Since those locations can be very close one to another, User can move the encounter bars in two different ways, rotating the bars or translate them.

To rotate the bars, select the *Rotate* button<sup>1</sup> to enable this action (Figure B.22). Then, dragging the mouse around the center of the image with the left mouse button adjusts both the direction and the height of the bars.

To move an individual bar, first select the *Move* button<sup>2</sup>, then click on the bar to be move and finally click on the new location of the bar.

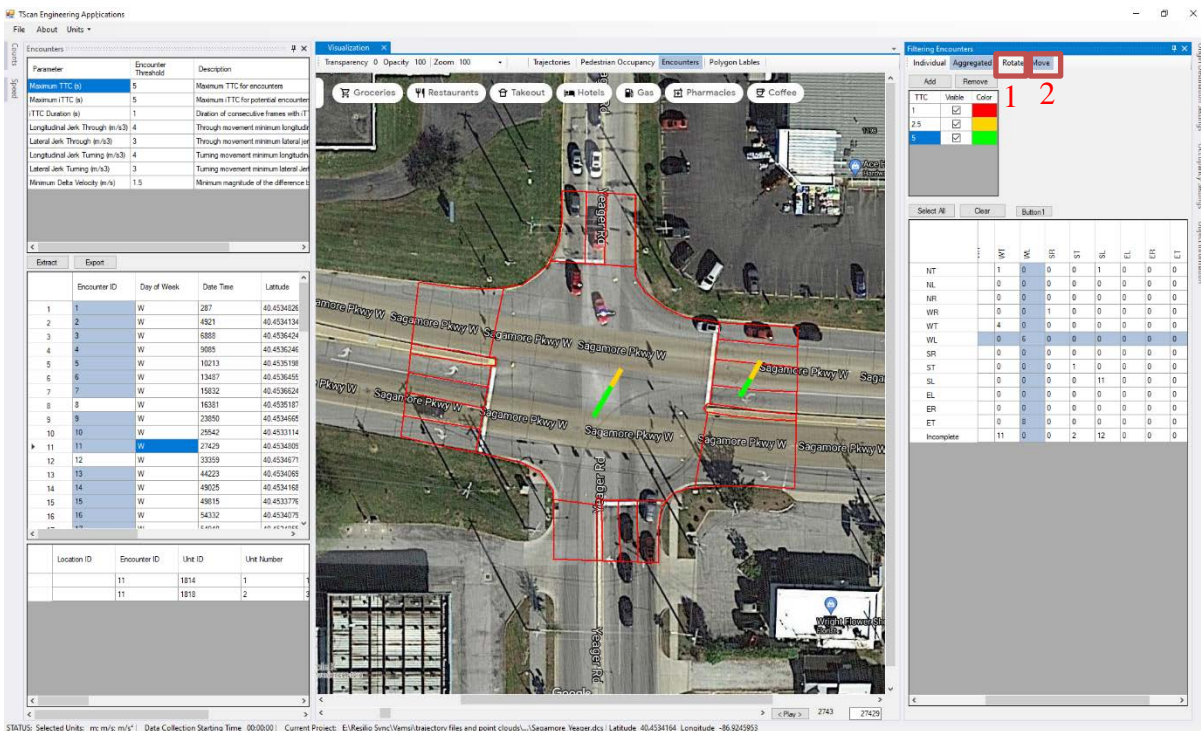


Figure B.22 Moving encounters bars.

## APPENDIX C. TSCAN USER MANUAL

### C.1 Introduction

TScan uses LiDAR technology that can detect and track various types of road users including buses, cars, pedestrians, and bicycles. Unlike video detection, LiDAR data has a one-to-one correspondence with the physical world. Hence, it is possible in principle to produce the positions and velocities of road users in real-time as needed for traffic and safety applications, with the errors of estimation dependent only on the resolution and accuracy of the LiDAR sensor.

The TScan prototype was developed by the Purdue Center for Road Safety as part of a research project funded by the Indiana Department of Transportation through the Joint Transportation Research Program. This user manual describes the overall process of preparing the system and then it focuses on the initial preparations executed in an office (off-site setup).

Setting TScan includes the following two phases (Figure C.1):

- off-site preparation, and
- on-site setup.

Once TScan data are collected, processed, and retrieved on site, they can be further processed off site with engineering applications. This first volume of the manual introduces the software TScan-Offsite Setup and guides the user in creating, saving, and retrieving the intersection's characteristics to transfer it later to the field data collection. Part II: TScan Onsite Setup discusses the required steps to collect data on the field.

The off-site process allows the user to enter, save, and retrieve the intersection's characteristics in a graphic environment to be used by the onsite setup, the data collection and processing program and other engineering applications. The information required by TScan data collection and processing program is a list of polygons that define the road lanes, intersection areas, parking areas, sidewalks, and medians along with the corresponding intended maneuvers. In order to create the required polygons, the user selects or uploads an orthographic image that will be used to draw the polygon edges. This can also be used in other engineering applications. Although it is not necessary to create a new folder, it is advisable to do so in order to have all the study information regarding a particular site in a single folder.

In the off-site preparation, the user enters, edits, and saves intersection characteristics that are later transferred to the TScan Onsite Setup and data collection. *TScan Offsite Setup* software was developed to facilitate the following operations: (1) Selecting and uploading orthographic images of intersections, (2) Drawing polygons that represent the intersection areas and (3) Entering polygon properties such as polygon type, and traffic maneuvers executed in the approach polygons.

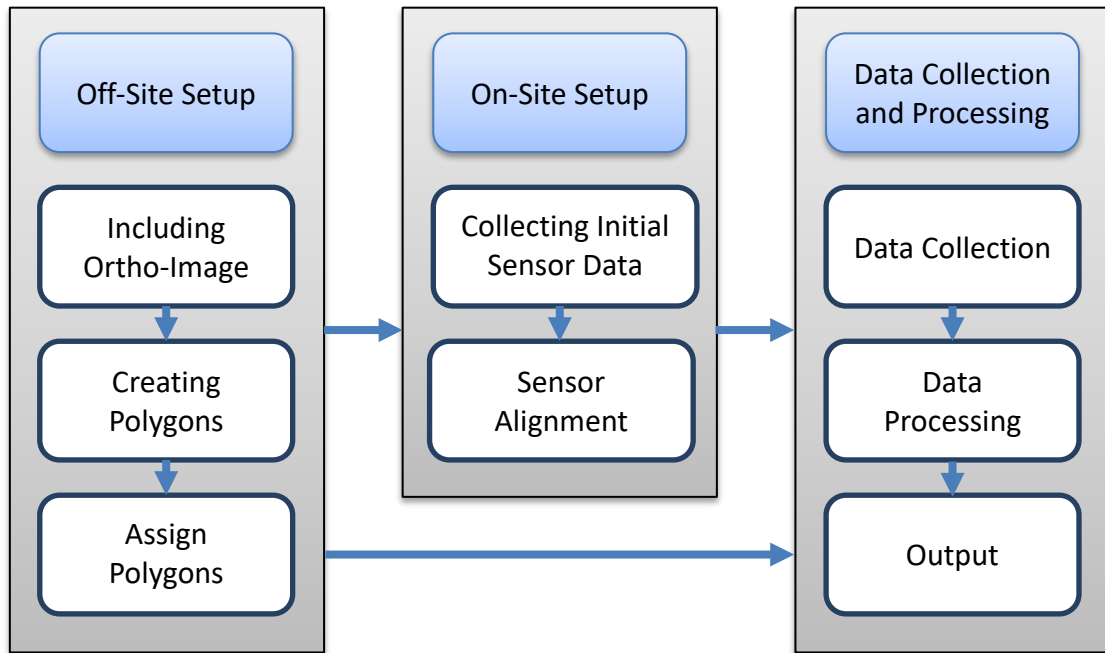


Figure C.1 TScan process overview.

*TScan Offsite Setup* software helps the user prepare site information needed to set the system in the field. The off-site preparation produces a setup file with the intersection polygons and other site characteristics. This file is needed in the on-site setup to establish a single spatial reference shared by the TScan sensors and the orthographic image of the intersection to relate the collected data to the intersection and the GIS coordinates. It is accomplished by overlaying the LiDAR-collected data with the orthographic image of the intersection. Characteristic elements of the intersection recognizable on the image and in LiDAR data are aligned by the user in the field with help of the on-site software. The alignment information is saved and exported to the TScan data collection and processing module.

After the setup information is obtained, the data collection and processing module receives the data from the sensors, processes it, and produces an output file. No user intervention is required while collecting data unless there is an indication of malfunctioning.

The output file produced as the result of data collection and on-site processing is useful for many engineering studies including traffic volume and speeds, traffic performance (queues and delays), capacity analysis, warrants for control devices (signalization, stop signs, etc.), pedestrian studies, red signal compliance, traffic conflicts, etc.

## C.2 Intersection Representation

The intersections in TScan are represented as a set of functional areas that determine both the potential users, the possible movements, and the operational function. These areas can be approaches, exits, intersection areas, parking areas, medians, or pedestrian areas. It is important to note that the areas must be interconnected.

- Approaches are areas that cover one vehicle arrival lane to the intersection. Each of these lanes is assigned the possible movements that vehicles can make, such as left or right turn, or through movement. The approach zone should be at least the length of one vehicle but the length of three vehicles is preferable.
- An exit area is the set of lanes after vehicles leave the intersection area. In general terms, it is not necessary to divide the exit area by lanes. The exit area should be similar to the adjacent approach area.
- Parking area is a lane adjacent to and parallel with the travel lane of a roadway that is used for parking vehicles.
- The intersection area is the zone shared by different vehicular movements or pedestrians. Usually starting at the stop lines.
- A median area is the portion of the roadway separating opposing directions of the roadway.
- Pedestrian areas are zones adjacent to the roadway that are reserved for pedestrian-only. Usually, pedestrian areas are defined adjacent to the intersection polygon and an approximate width of two meters (7 ft.) is used.

This intersection representation is important for tracking objects. They define the areas of interest to select the LiDAR points for further analysis. Points outside of the defined areas are not considered. It also helps to define the intersection background and surfaces, to classify objects and to check if an object tracking is completed.

To define the intersection, an orthographic image is obtained on which the edges of the areas of interest are drawn. Then, the zones are selected, and the characteristics should be completed.



### **C.3 TScan Offsite Setup**

TScan off-site configuration software helps the user to prepare the site information required to configure the system in the field. Off-site Setup produces the configuration files corresponding to the data collection site that include the intersection representation for TScan, its characteristics, and an orthographic image.

#### *C.3.1 Software Installation*

*TScan Offsite Setup* is compatible with Windows 10. In order to run the program the MS .NET 4.5.2 Framework or later component must be installed. If the MS .NET 4.5.2 Framework is not present during the installation, TScan will attempt to install this component if the PC is connected to the Internet.

To install *TScan Offsite Setup*, follow the steps given below.

1. Extract the contents of the archived file to your local drive.
2. Click on the *setup.exe* file to initiate the installation.

After checking that the program is installed and working, the user may delete the unzipped files in the folder with the *setup.exe* file to save disk space. The user should save the zipped/compressed file in case it is needed to reinstall the program.

### C.3.2 Launching TScan Offsite Setup

Once installed, the *TScan Offsite Setup* program can be launched using any of the following methods.

Method 1: Double click the shortcut on the desktop.

Method 2

1. Press *Start* button.
2. *TScanUI* should appear in the list of installed programs.
3. Single click on the shortcut.

Method 3

1. Press *Start* button.
2. Start typing *TScanUI*, the program shortcut should appear in the search results
3. Single click on the shortcut

Method 4

1. Open *My Computer*
2. Browse to the location where the software was installed. Typically, '*C:\Program Files (x86)\TScanUI*'
3. Click on *TScanUI.exe*

### C.3.3 Offsite Setup Interface

The main interface appears within a few seconds (Figure C.2).

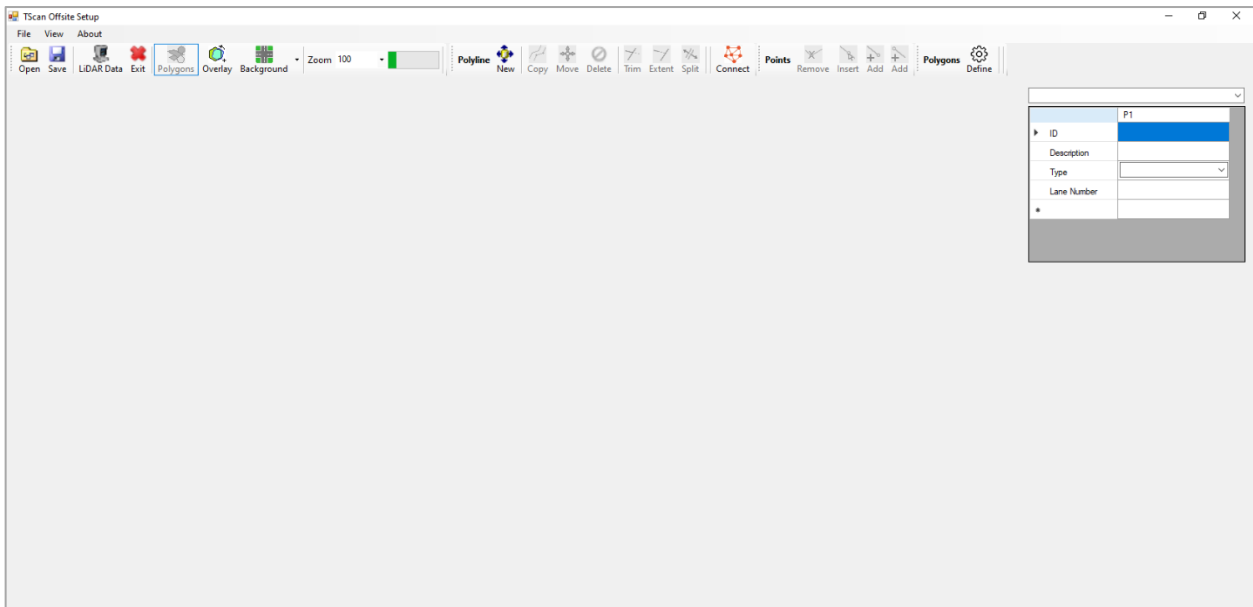


Figure C.2 TScan Offsite Setup process main interface.

The program interface includes a menu and command bar (Figure C.3). The menu includes tabs: *File*, *View*, and *About*. The *File* tab facilitates the process of saving or opening existing projects as well as exiting the program. The *View* tab allows the user selecting the option to visualize the command bar (second row) with only *icons*, *icons + text* or only *text*. The default configuration of the software provides the option of *icons + text*. The commands facilitate the operations on files, polylines, points, polygons, and LiDAR alignment (Figure C.3).

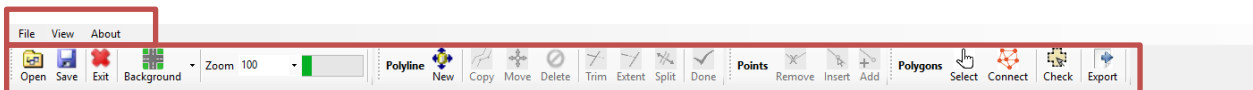


Figure C.3 Menu and command bars in the interface.

### C.3.4 Background Image

A background image is needed to help the user develop an electronic representation of the intersection. *TScan Offsite Setup* offers three options of bring the background image: (1) capturing from Google maps, (2) capturing from Bing maps, or (3) uploading from a file.

When capturing the images from Google or Bing maps, the original scale of the image must be preserved. To meet this condition, the computer *Display* settings must have the scale value in the *Scale and layout* option set at 100% (Figure C.4).

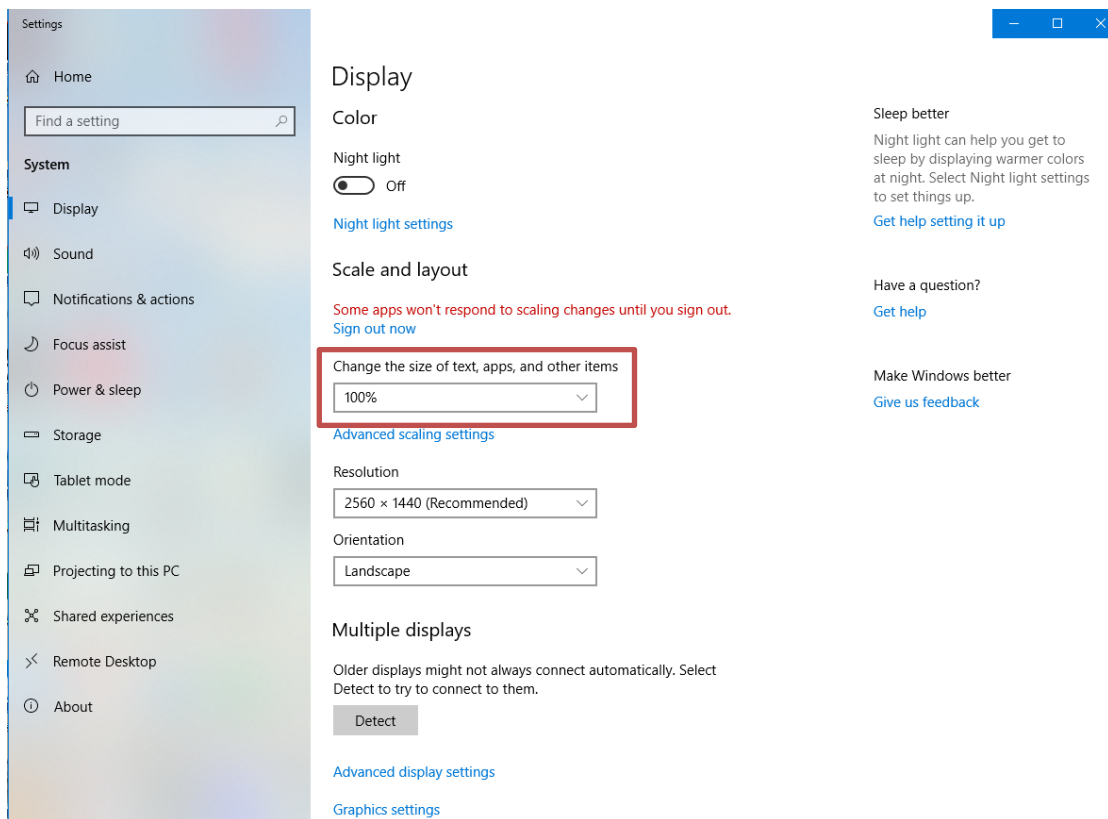


Figure C.4 Display settings: scale and layout 100%.

Select the desired source of the image from the drop-down menu in *Background<sup>1</sup>* of the *TScan Offsite Setup* (Figure C.5). Once the source is selected and the connection with either Google or Bing maps is established, look for the intersection of interest. Use the *Capture<sup>2</sup>* command button to store the

background image as shown in Figure C.5. The user can quickly find the intersection by entering the latitude and longitude<sup>3</sup> of the intersection and pressing the *Go to*<sup>4</sup> button.

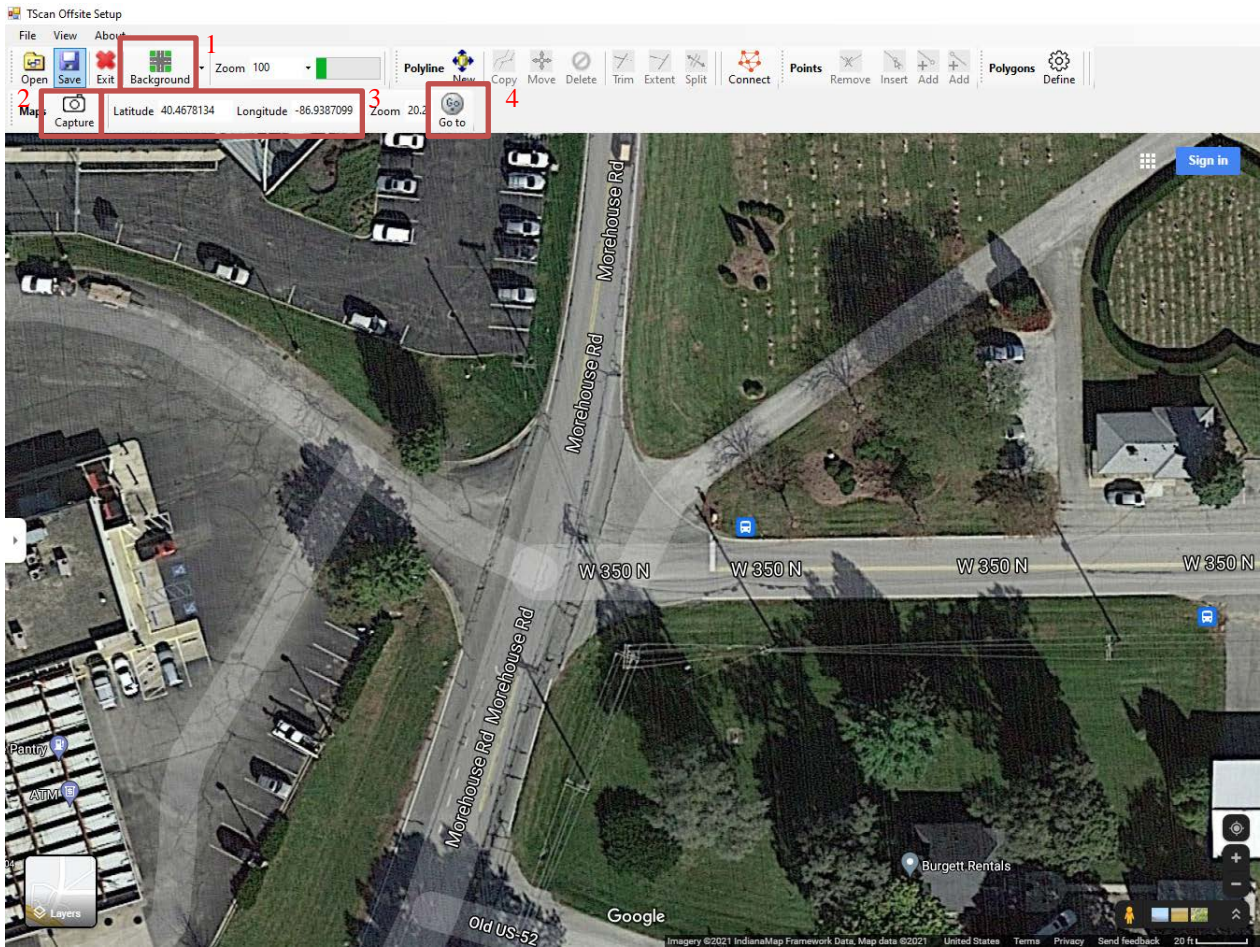


Figure C.5 Capture of orthogonal image to generate intersection layout.

The user has the additional option of uploading an image from a file. Select the option *From File* on the *Background* drop-down menu to open the Windows file selection window shown in Figure C.6 and select the folder where the file can be found and select the file.

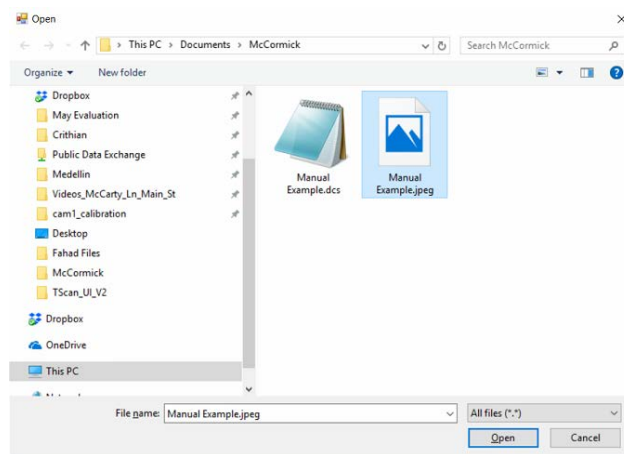


Figure C.6 Open background image from file.

It is important that when capturing the image some easily distinguishable nearby elements such as poles or buildings are visible in the image.

Once the image is uploaded, the scale of the displayed image can be adjusted by selecting one of the available zoom factors (Figure C.7).

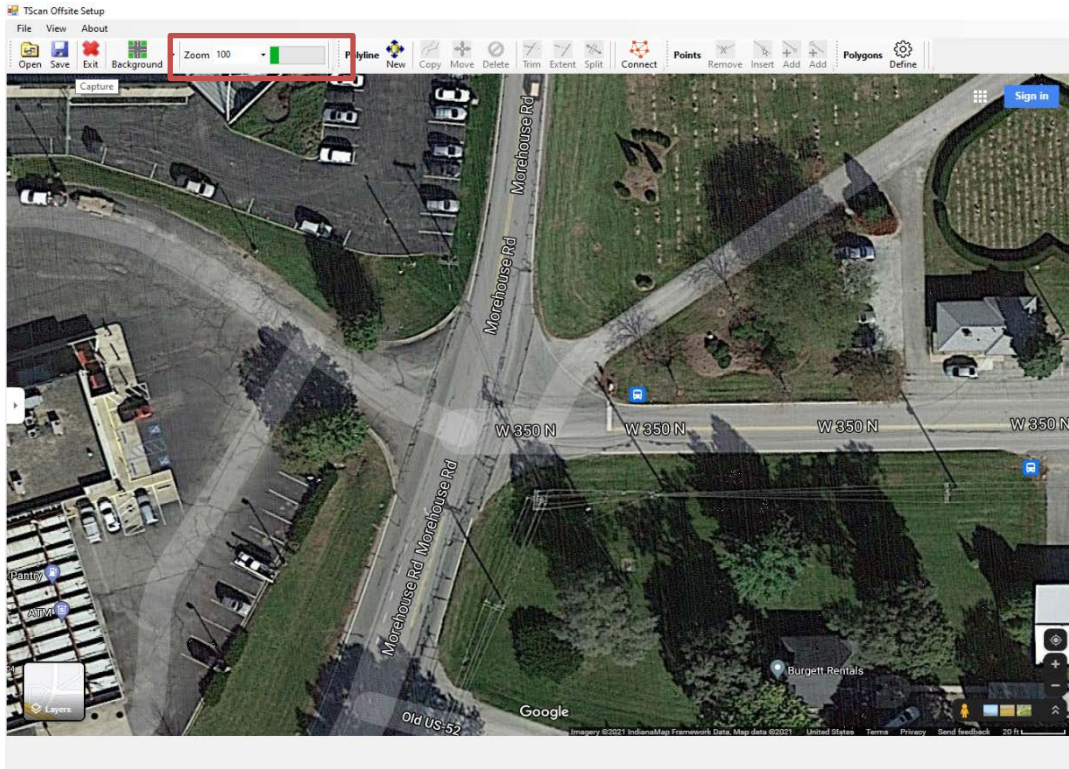


Figure C.7 Adjust background zoom factor.

### C.3.5 Intersection Layout

Certain elements of the intersection need to be defined to help TScan detect objects moving within the intersection. Convenient drawing tools are available for this task in the polyline command bar and the point command bar. Table C.1 shows the supported operations including drawing polylines, adding points, and defining polygons.

Table C.1 List of supported operations for various polygon-related features

Feature	Supported Operations
Points	Insert Add Remove
Polylines	Add Delete Copy Move Trim Extend Connect
Polygons	Define

### *Polyline Commands*

Polyline is useful to represent the intersection layout. It is a continuous line composed of one or more straight segments. The polyline command bar includes tools for adding, deleting, copying, moving, trimming and, extending polylines as shown in Figure C.8.

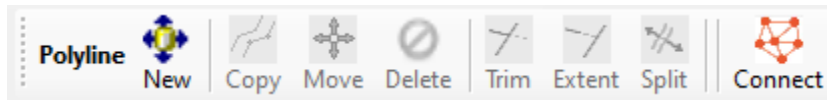


Figure C.8 Polyline tools in the command bar.

### *Adding Polylines*

The first step of drawing the intersection layout is adding polylines. You can create a polyline by clicking on the *New* button and then specifying the vertices of the polyline by clicking at the desired points on the orthographic image. Once all vertices are created, right-click on the image to finalize adding points to the polyline. The vertices of the new polyline will be deemphasized. Figure C.9 shows an example of adding polylines.

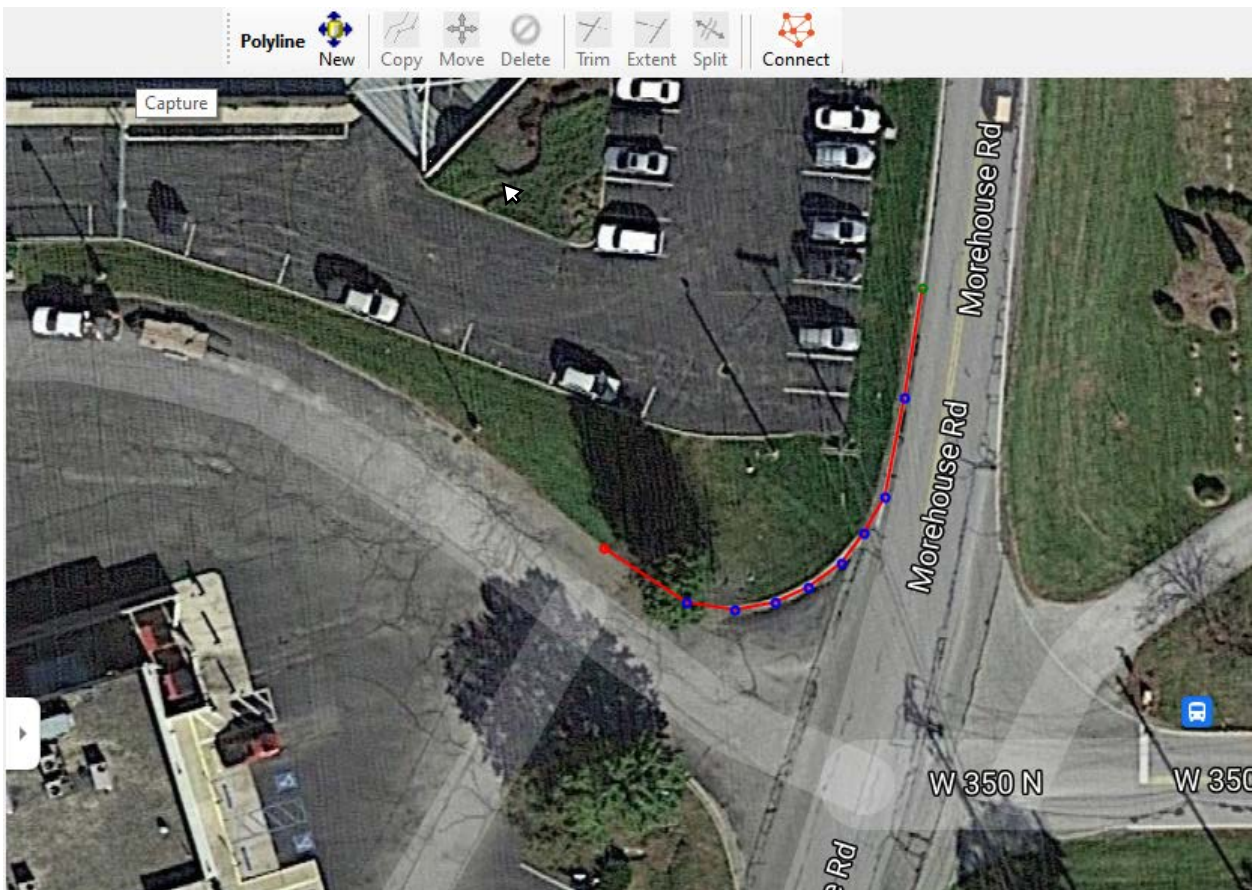


Figure C.9 Adding polylines.

**Polylines should not intersect themselves.** The program does not support drawing closed polylines. Closed areas can be created by allowing consecutive polylines intersect each other (Figure C.10).



(a) Incorrect–Closed Area with a single polyline.



(b) Correct–No closed areas with a single polyline.

Figure C.10 Avoid closed areas within a single polyline and ensure polylines to intersect.

### Deleting Polylines

To delete a polyline, select the polyline by clicking on it. The vertices are displayed to confirm that a polyline is selected and ready for editing. Then, click on the *Delete* button to remove it from the image. Figure C.11 shows an example of deleting polylines.

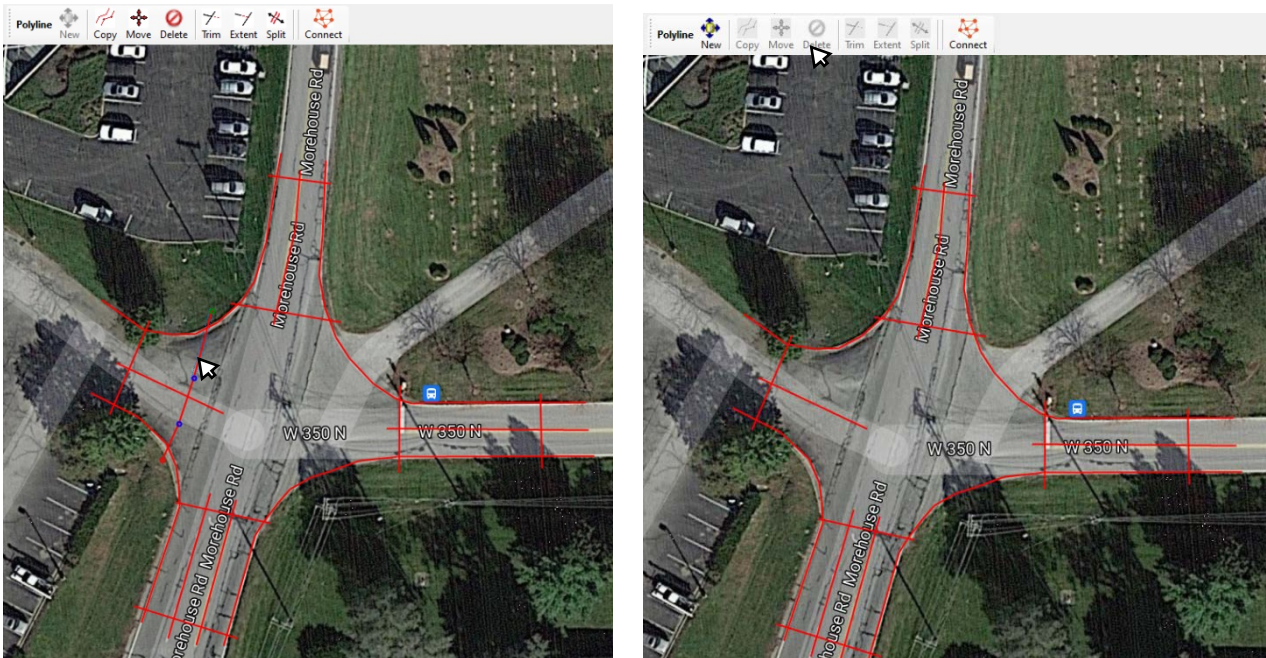


Figure C.11 Deleting polylines.

### Copying Polylines

To create several parallel polylines, click on a polyline (vertices are emphasized), click on the *Copy* button, and then click on the image at the point where the new polyline is supposed to begin. Each next click generates another copy of the polyline. Right-click on the image to end copying polylines. Figure C.12 shows an example of a copied polyline.



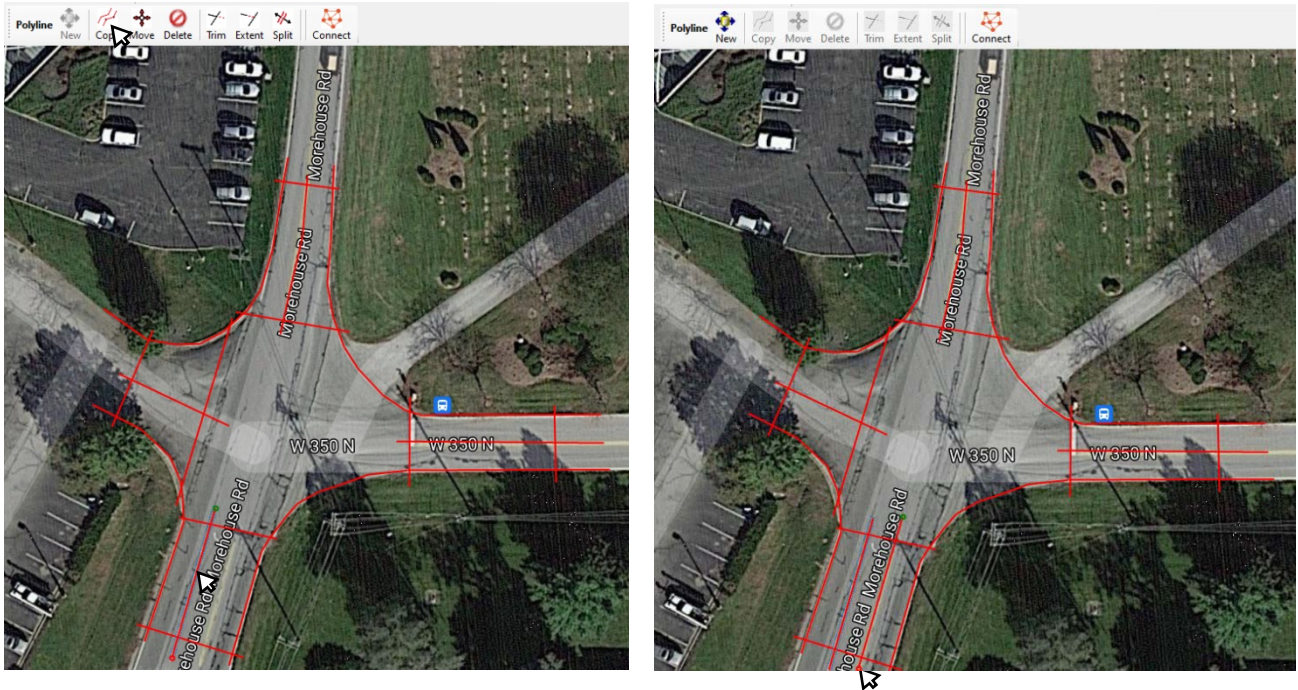


Figure C.12 Coping a polyline.

### Moving Polylines

Select the polyline to be moved to highlight it. Then, click on the *Move* button and pressing the mouse left button, move the polyline to the desired location and release the mouse left button to finish. Right-click on the image to end moving the polyline. Figure C.13 shows an example of a moved polyline.

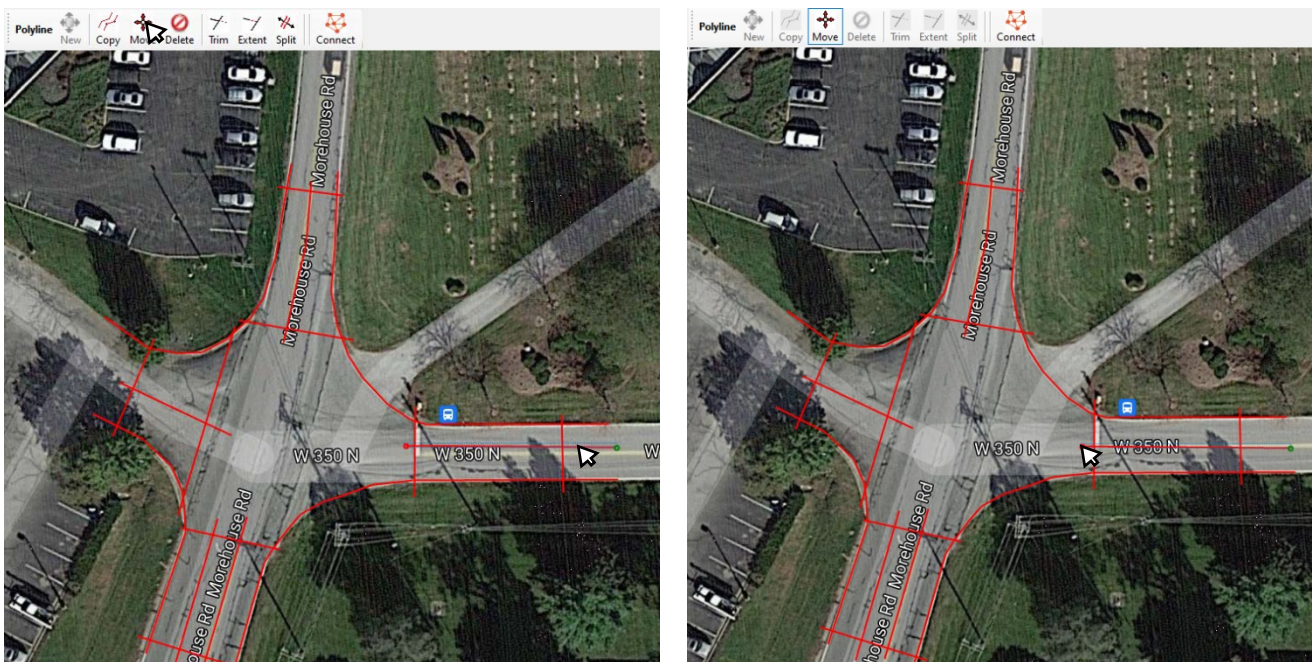


Figure C.13 Moving a polyline.

### Trimming Polylines

To trim a polyline at the crossing point with another polyline, select the polyline, which will remain untrimmed (vertices will be emphasized). Click on the *Trim* button and the selected base polyline will change its color to yellow. Click on an intersecting polyline on the side to be removed. Repeat this operation to trim other intersecting polylines. Finally, right-click on the image to end the trimming operation. Figure C.14 shows an example trimmed polyline.

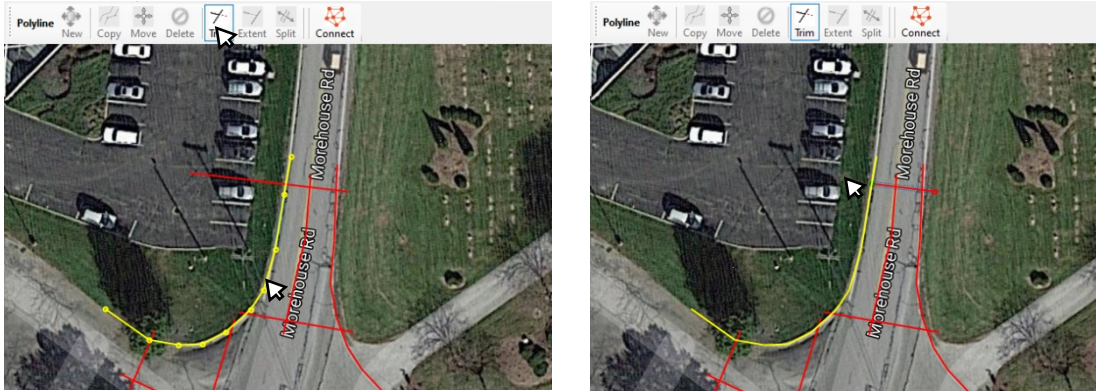


Figure C.14 Trimming a polyline.

### Extending polyline to Intersection with Another Polyline

To extend a polyline to an intersection with another polyline, it is necessary to select the other polyline, which is to remain unchanged. After selecting that polyline, its vertices become emphasized. Click on the *Extend* button. The selected base polyline will change to yellow. Then click on the polyline to be extended on the side where the extension needs to be made. Repeat this operation to extend other polylines. Finally, right-click on the image to end extending the polylines. Figure C.15 shows an example of an extended polyline.

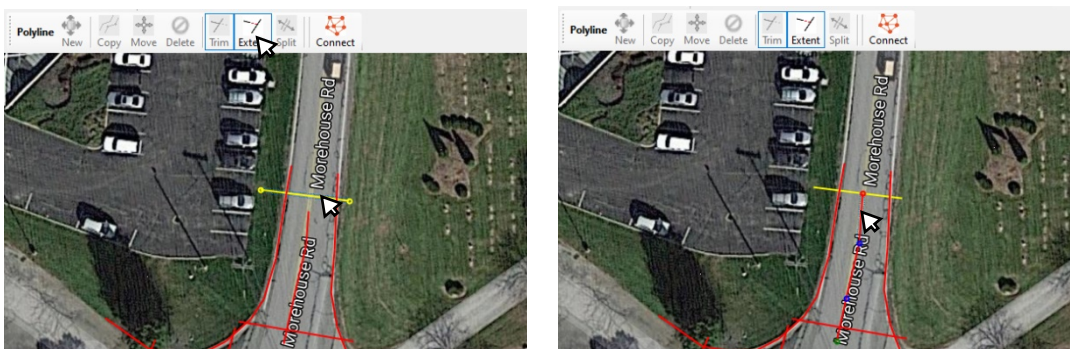


Figure C.15 Extending polylines.

### Vertex Commands

Another option to edit the polylines is to edit their vertices. The point command bar tools include functions for editing, removing, inserting, and adding polyline's vertices as shown in Figure C.16.

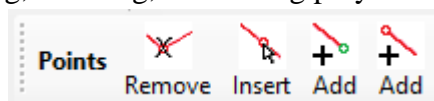


Figure C.16 Points tools in the command bar.

### Edit Vertices

To edit the vertices, first select the polyline by clicking on it. After selecting, the vertices are displayed on the screen. To change the vertex's location, simply click on the vicinity of the vertex, drag it to the new position, and release the mouse button. Once all the changes of vertices' locations are executed, right-click on the image to finalize editing the vertices. Figure C.17 shows an example of moving vertices.

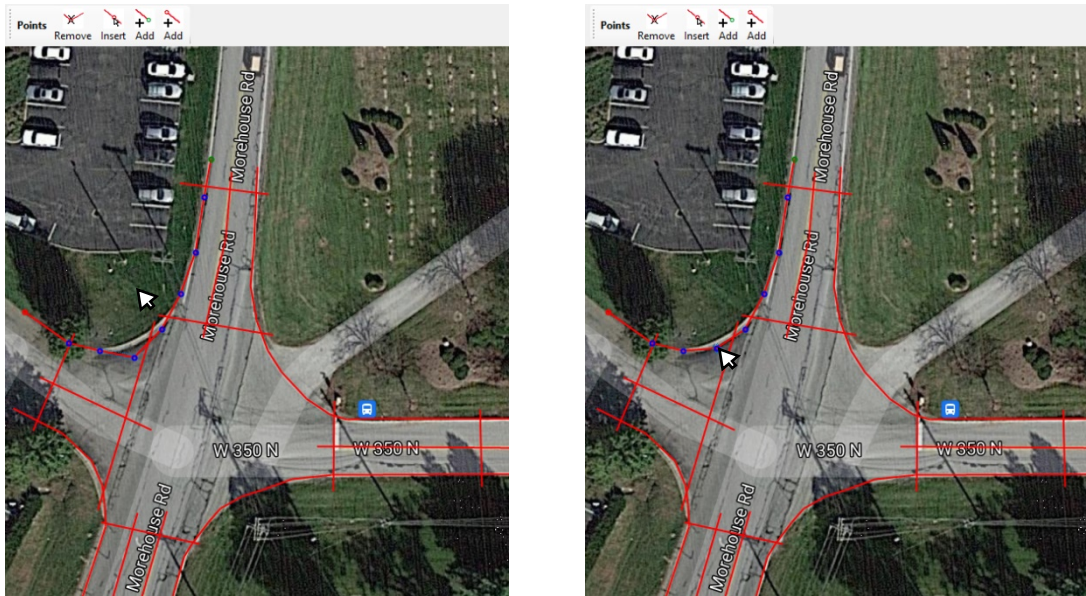


Figure C.17 Editing vertices.

### Insert Vertices

To insert vertices, first select the polyline by clicking on it; the vertices are highlighted. Click on the *Insert* button to activate the function. Then, click on the polyline where the new vertices should be added. To end adding vertices, right-click on the image. Figure C.18 shows an example of inserting vertices.

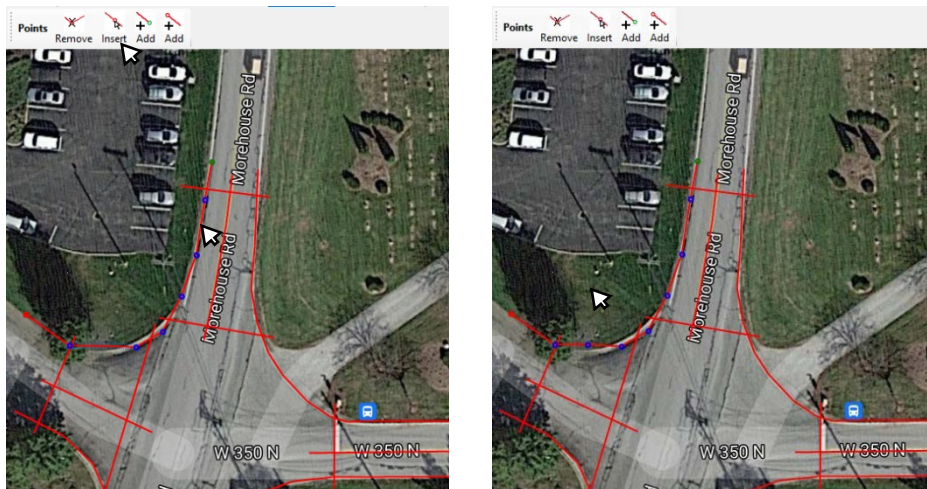






Figure C.18 Inserting vertices.

### Extending Polyline by Adding a Vertex

The difference between inserting and adding vertices is that inserting adds points between two consecutive vertices while adding vertices creates an extra vertex at an end of the polyline. To add vertices, select the polyline by clicking on it. After selecting, the vertices are displayed on the screen, the extreme vertices of the polyline are shown one in red and the other in green. The click on the *Add* with red dot button to activate the function that add vertices at the red end of the polyline or click on the *Add* with green dot button  to activate the function that add vertices at the green end of the polyline. To create a new  vertex,  click on the location of the new vertex. Right-click on the image to finalize adding  vertices. Figure C.19 shows an example of adding vertices.

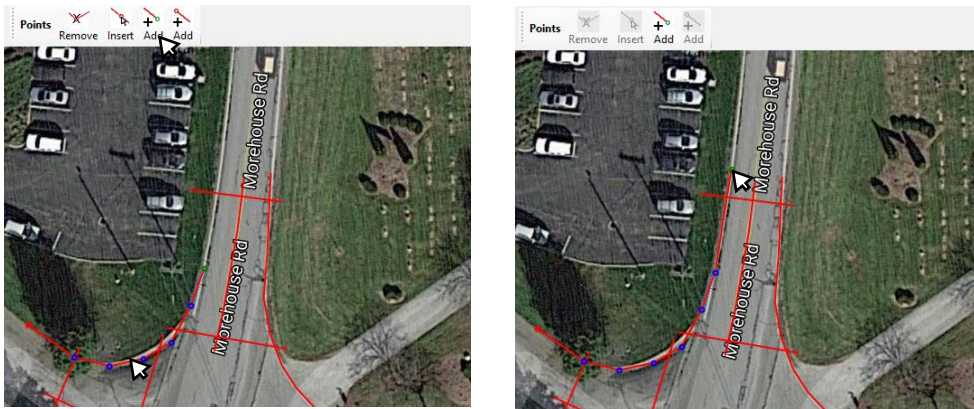


Figure C.19 Adding vertices.

### Remove Vertices

To remove vertices, select the polyline by clicking on it to highlight its vertices. Click on the *Remove* button to activate the function. Click on the vertex to be removed from the selected polyline. Right-click on the image to finalize removing vertices. Figure C.20 shows an example of removing vertices.

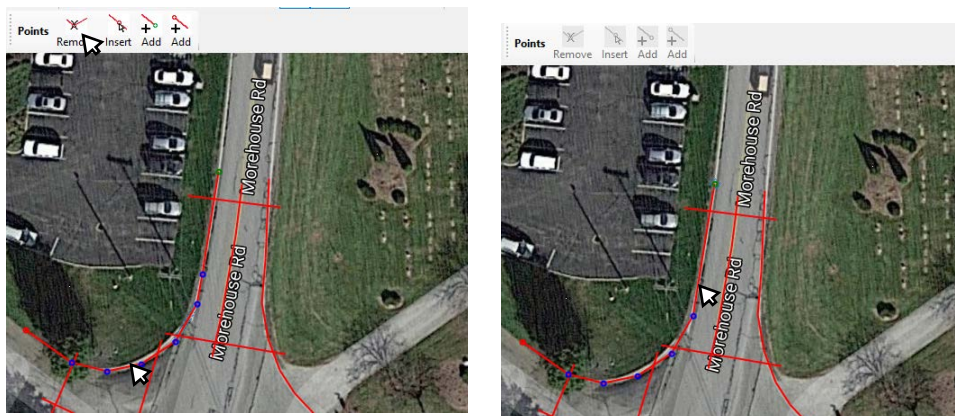


Figure C.20 Removing vertices.

### Connecting Polylines

Use this function to trim all segments of the polylines and to add vertices at the intersection of any segment pairs. This command also removes all unconnected segments. Figure C.21 shows the result of the *Connect* function. The *Connect* polyline's function facilitates the process of drawing the intersection layout by allowing drawing boundaries of the intersection and lanes without paying attention to corners and then integrating and trimming redundant lines. The obtained intersection areas have limits and corners that meet the requirements needed to track objects inside the intersection effectively.

Before using this function, make sure that all polylines have at least two intersections with other polylines. Polygons that do not have at least two intersections will be deleted as a result of this process.

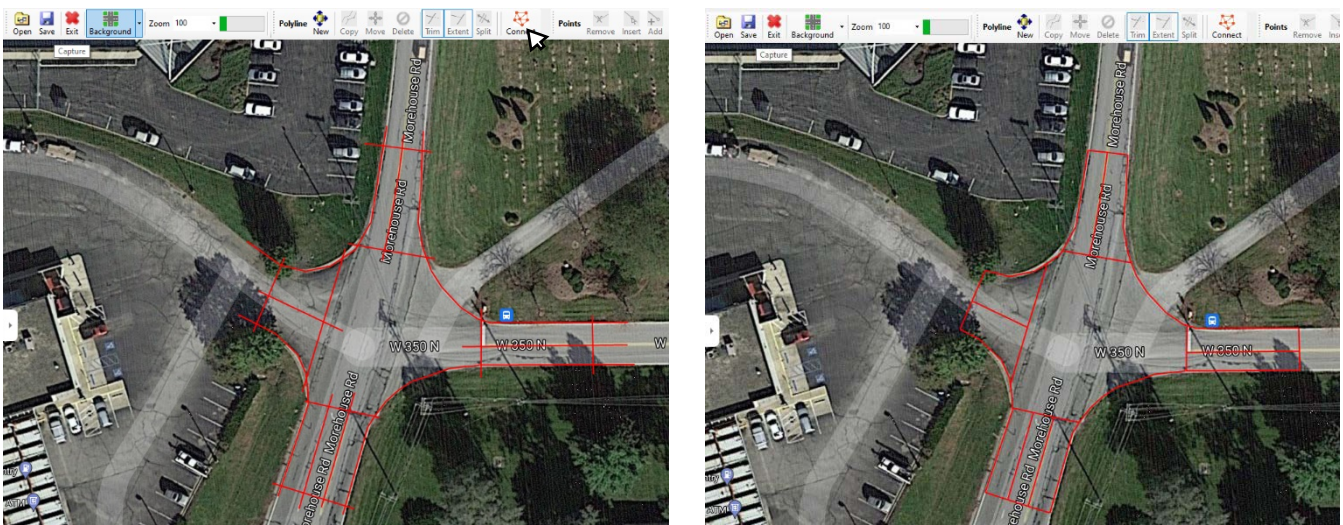


Figure C.21 Connect function before/after.

### Creating Polygons

Once the intersection layout is drawn, the next step is to create the polygons and set their characteristics. The polygon commands provide an easy way to convert polylines into closed polygons by executing several polyline operations at once.

### Define Polygons

Use the *Define* command to select a polygon and to define its properties. After activating the *Define* command, the polygon definition bar appears (Figure C.22). Commands in this bar allows defining or redefining the function of the selected polygon. The definition includes type of polygon: approach, exit, intersection, parking, median and sidewalk. An approach polygon has associated traffic maneuver allowed from the polygon (typically a single lane): left, through or right. An exit polygon includes lanes after crossing the intersection (central area). A parking polygon is a lane used for curb parking. Sidewalk polygon is reserved for pedestrians and possibly for bicycles. An intersection polygon is the common area where vehicles leaving the approach polygons cross each other paths and leave this polygon by entering the exit polygons.

Once the command is activated, click on the image inside the polygon. The polygon area is highlighted. If the polygon already exists, the polygon information is displayed, otherwise, it has just

been labeled by clicking on it is ready to be defined by entering its properties. Select one of the polygon definition command options to assign the function to the polygon.

Other polygon characteristics, such as zone ID, description, and lane number, if applicable, can be entered in the right-hand pane of the interface window. (Figure C.22).

The polygons can be selected either by clicking in the image inside the polygon or by selecting it from the pulldown list at the top of the right-side panel. Note that only polygons selected at least once by the user are listed in this panel. Right-click on the image to end defining polygons.

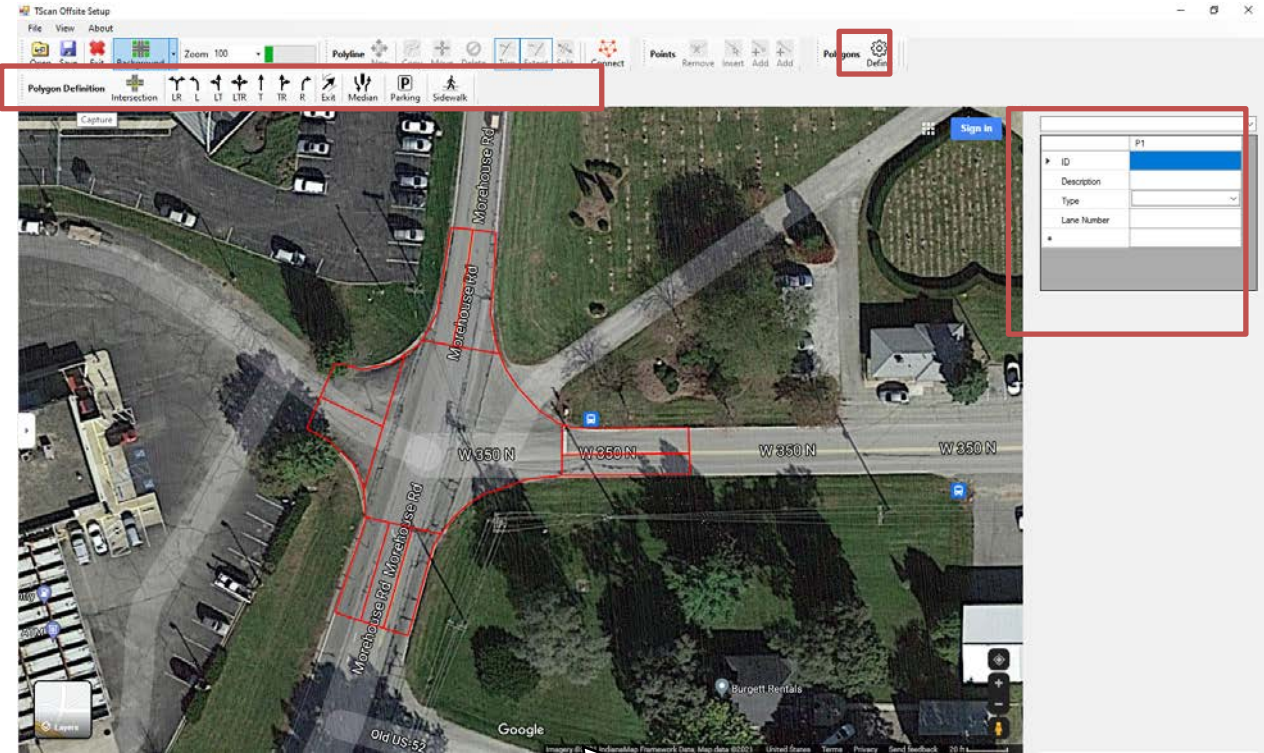


Figure C.22 Defining polygons.

### Saving and Retrieving Layouts

TScan Offsite Setup allows saving the orthographic-image of the intersection as well as a “.dcs” file with the polygons layout and properties that are used during the on-site setup of the TScan system. Table C.2 shows the list of files produced and saved.

Table C.2 Files produced with the TScan Offsite Setup

File	Format	Description
<Orthographic-image>	Any image format such as bmp, jpg, jpeg, gif, png, tif, etc.	User can capture this image from the online mapping options or can provide a top view image of the intersection saved in image format.  This file is used as a reference to mark entry lanes, exit lanes, intersection area, parking areas, medians and sidewalks.
<Location Name>.dcs	Text format with extension dcs	TScan generated file that contains all user information regarding the data collection settings using the orthographic-image coordinates system.

### Saving Layouts

After clicking on the *Save* button, the user is asked to provide a file name and a folder to which the layout is to be saved (Figure C.23).

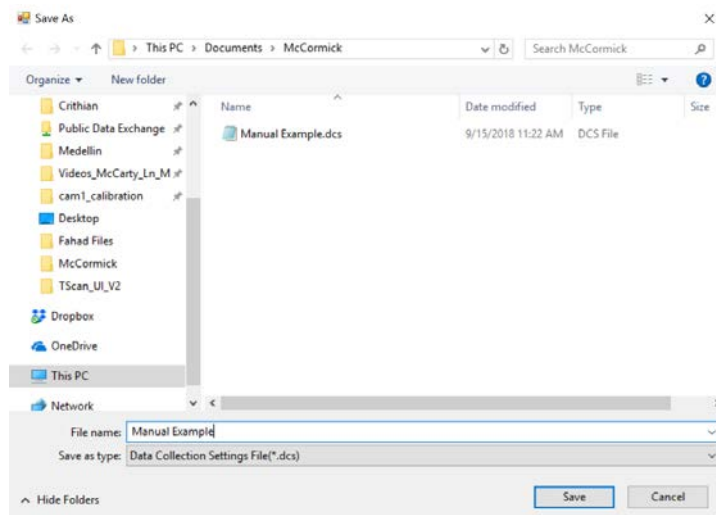


Figure C.23 Save layout.

### Retrieving Layouts

After clicking on *Open*, the user is asked to select the file to be opened as shown in Figure C.24. If the orthographic-image path is not found, it only retrieves the polylines and the polygon information so the user should open the background image as well.

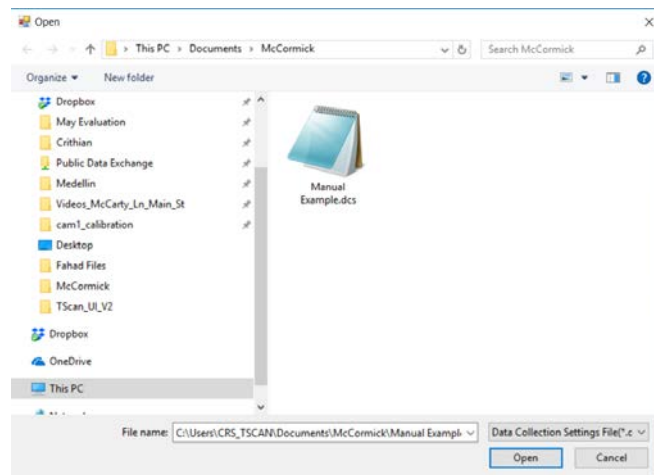


Figure C.24 Open layout.

### *Offsite Setup Final Remarks*

The .dcs file generated by the *TScan Offsite Setup* program contains all the user information regarding the orthographic image, the polygons information, and the associated coordinates. Please keep and take this file with you for the field data collection. This information is required in the next step of the TScan setup. The aforementioned step includes setting a common reference system for the TScan sensor and the orthographic image in order to transfer the polygon characteristics to the TScan processing module. This requires an appropriate setup of the TScan hardware in the field.

## **C.4 TScan On-Site Setup**

The *TScan On-site Setup* software assists the user in preparing the site information needed to set up the system in the field. The *TScan On-site Setup* program reads the files prepared in the Off-site Setup which includes both the orthoimage and the intersection layout and characteristics. This file is needed to establish a single spatial reference shared by the TScan sensors and the ortho image of the intersection to relate the collected data to the intersection and GIS coordinates.

Since the TScan Head computer has no monitor or keyboard, the computer and program are controlled via remote access using the trailer's ethernet connection.

After the setup information is obtained, the data collection and processing module receives the data from the sensors, processes it, and produces an output file. No user intervention is required while collecting data unless there is an indication of a malfunction.

The output file produced as the result of data collection and on-site processing is useful for many engineering studies.

### *C.4.1 Remote Desktop*

After setting up the TScan hardware, connect the user laptop to the one located on the TScan head. To do so, connect the laptop to the WiFi Network TScan with password tscan. Once the laptop is connected to the specified network, launch the Remote Desktop Connection program in the laptop. This program can be launched by following these steps:



1. Press start button.
2. Start typing *Remote Desktop Connection*, the program shortcut should appear in the search results.
3. Single click in the shortcut

In the dialog box of the program type *TScanMaster* under the Computer Tab. Once this computer is selected click connect. Enter TScan as the user credentials. When the connection is established, you will be able to visualize the desktop of the computer located on the TScan head and the main window of the *TScan On-site Setup* software.

#### *C.4.2 Launching TScan On-Site Setup*

*TScan On-Site Setup* software helps the user prepare site information needed to set the system in the field. The software is already installed in the TScan Head computer and is launched automatically when the system is powered up.

In case the *TScan On-site Setup* program is not running, it can be launched using any of the following methods.

Method 1: Double click the shortcut on the desktop.

Method 2

1. Press *Start* button.
2. *TScanOnSite* should appear in the list of installed programs.
3. Single click on the shortcut.

Method 3

1. Press *Start* button.
2. Start typing *TScan*, the program shortcut should appear in the search results.
3. Single click on the shortcut

Method 4

1. Open *My Computer*.
2. Browse to the location where the software was installed. Typically, '*C:\Program Files (x86)\TScanOnSite /*'.
3. Click on *TScanOnSite.exe*.

### C.4.3 On-Site Interface

Figure C.25 shows the main interface.

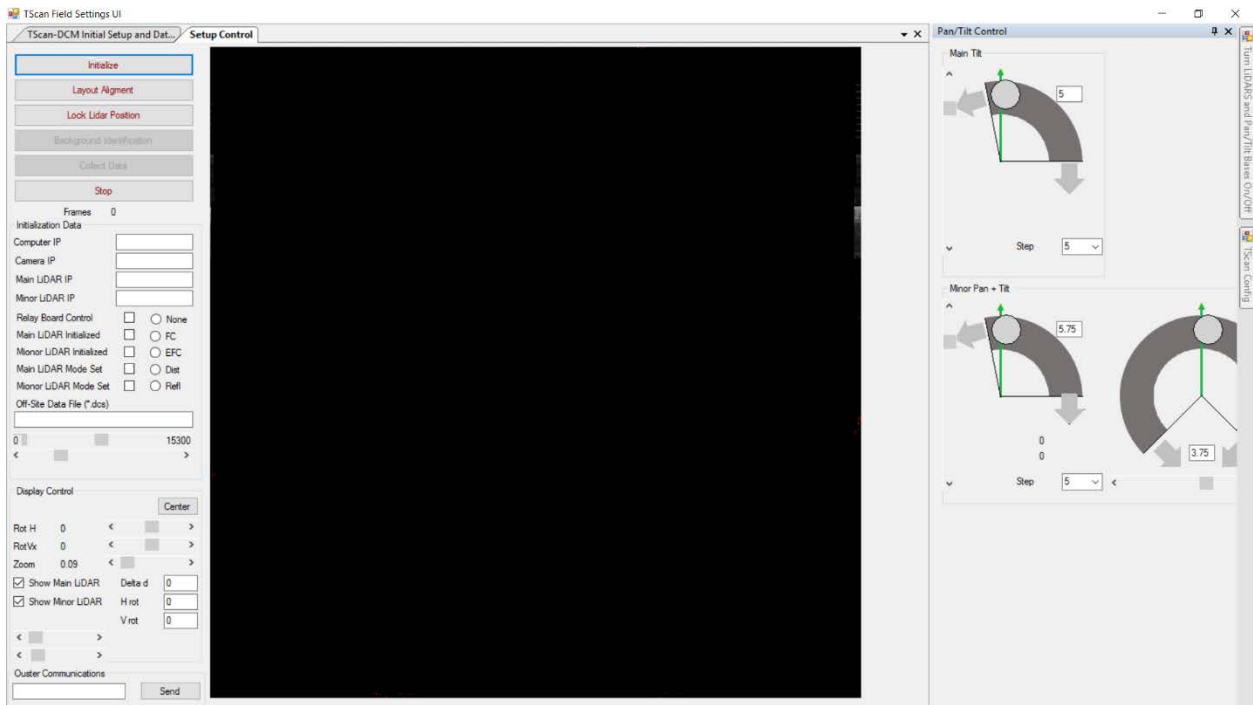


Figure C.25 TScan on-site setup main interface.

The program includes several windows. The control and live display window is used to execute the different program functions and to display the sensor readings in order to ensure the intersection coverage; the orthoimage alignment window is used to overlay the orthoimage with the sensor readings; and the configuration window is used to indicate the ports for the different hardware elements in the TScan head.

### C.4.4 Sensor Initialization

The first step is to initialize the sensors. To do so, click on the *Initialize* button to start the process (Figure C.26). This process can take few minutes. The IP addresses are displayed in the control panel as well as the checklist of the sensors that are initialized.

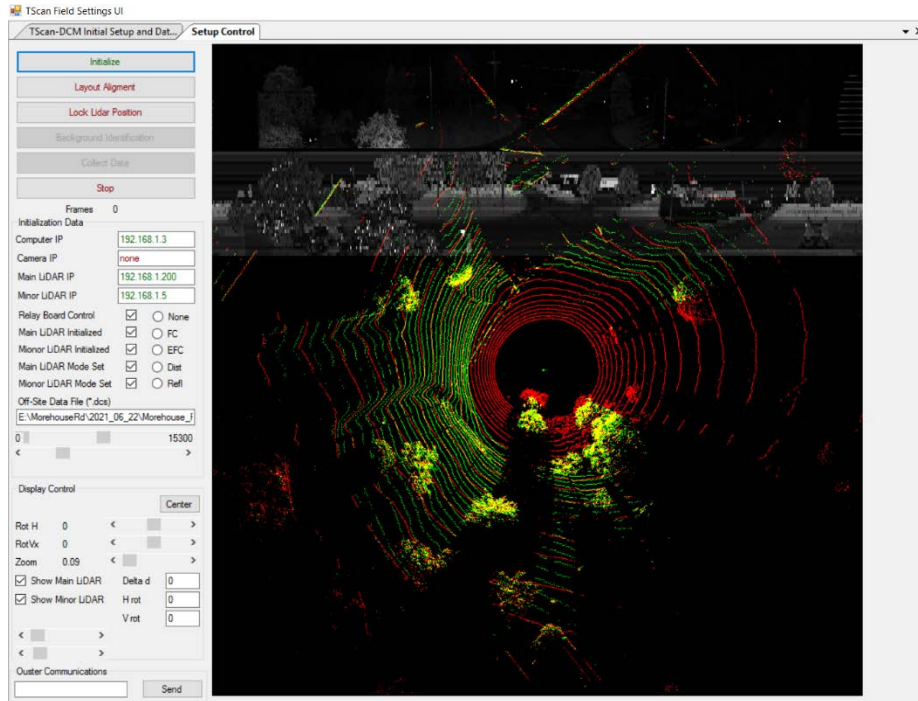


Figure C.26 TScan on-site setup control and live sensor readings display.

Once the sensors are ready, the program ask for the location of the .dcs files prepared during the off-site setup.

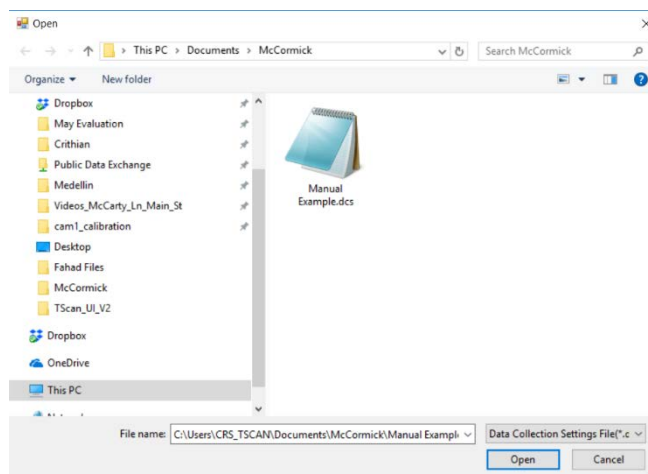


Figure C.27 Retrieving the off-site information.

Once the files are retrieved, the user can visualize the point cloud collected with the main sensor (located on top) in red color. The point cloud collected with the auxiliary sensor is shown in green color. A black and white image of the sensor's readings are also displayed.

### C.4.5 Sensors Orientation

After initializing the sensors, the next task for setting up the system is to find an optimal position for the LiDAR sensors. The Pan/Tilt Control Tab allows the user to define the position of the main LiDAR (located on top of the TScan Head) and the Auxiliary or minor mechanism (located in the lower position). The user can control the pan/tilt mechanism using the scroll option or by defining as input the desired angle (Figure C.28). The reference zero is shown in green.

The appropriate position of the LiDAR involves maximizing the coverage at the intersection provided by the two sensors.

To maximize coverage, set up the mast and the head mechanism pointing at the center of the intersection. Tilt the main sensor that allows collecting points from vehicles from approximately 150 ft. The auxiliary LiDAR provide a pan and tilt mechanism in case the user requires a better detection of a specific approach of the intersection. If the user is interested in better detection on the intersection polygon, the pan mechanism should be positioned at the zero reference. Make sure to interlace the point cloud between the two sensors when selecting the position of the pan mechanism for the auxiliary sensor.

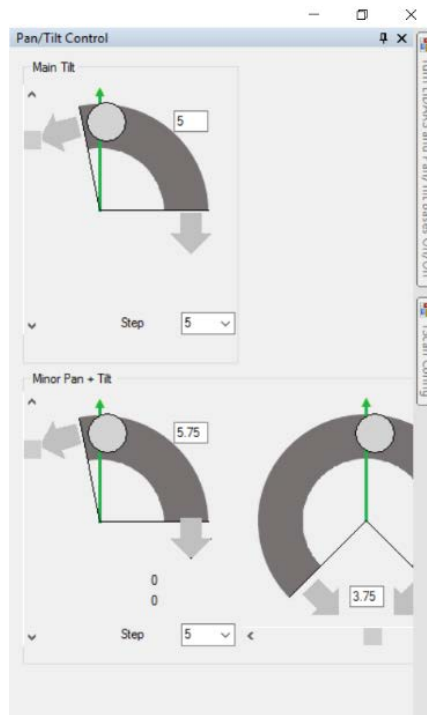


Figure C.28 LiDAR orientation window.

### C.4.6 Layout Alignment

The step aims to set a common reference system for the TScan sensor and the orthographic image in order to transfer the polygon characteristics to the TScan On-Site Setup. The initial alignment can be performed semi-automatically based on user defined reference points. Then fine manual adjustments can (also) be made.

The information created off-site is retrieved and displayed, then an initial sensor data is collected and displayed over the orthographic image as shown in Figure C.29. Then, the interface allow user to move and rotate the TScan points in order to properly align them and to perform further adjustments. Once the user is satisfied with the alignment the information is saved and exported to the data

collection and processing module. The orientation of the LiDAR should not be changed after this point. This includes rising and lower the mast, changing the Pan-Tilt base setting or leveling the unit. If the .dcs file was not opened after initialization, then the .dcs file containing the information regarding the polygons needs to be opened. Press the button *open* and select the .dcs file corresponding to the layout of the intersection to analyze.

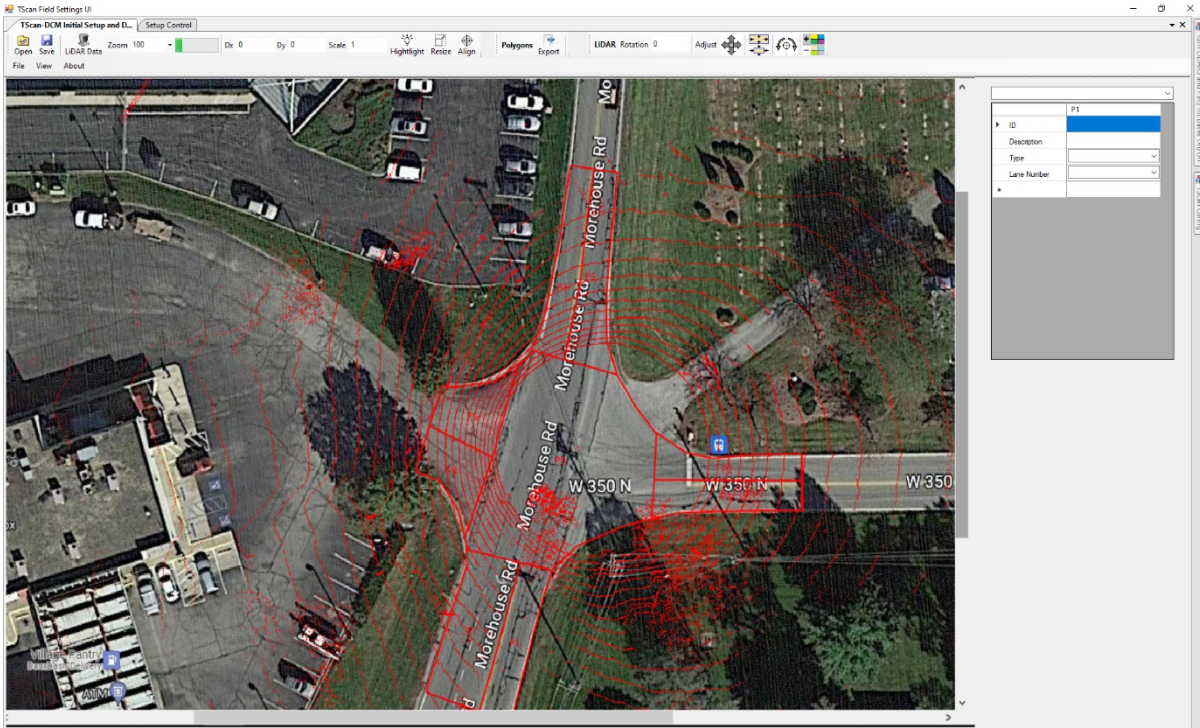


Figure C.29 Initial sensor data.

### *LiDAR Alignment*

The orthographic-image and the TScan sensor must have a common reference system therefore aligning them is necessary. The initial alignment is performed by the user by translate and rotating the LiDAR's points. Then fine adjustments can (also) be made.

### *Initial Alignment*

To perform the initial alignment, select the *Align* button<sup>1</sup> to enable the translation and rotation of the LiDAR's points. To translate drag the LiDAR's points over the orthographic image using the left mouse button. To rotate drag the LiDAR's points using the right mouse button around the selected translation point. As a first attempt, drag the center of the point cloud to the estimates mast location on the image and then rotate the point cloud around the point until the points align to the orthographic image. Repeat the process using a clearly distinguish reference points until the align in satisfactory.

The user should take into account that some images might not be completely perpendicular to the terrain so the closer the reference points to the surface the better.

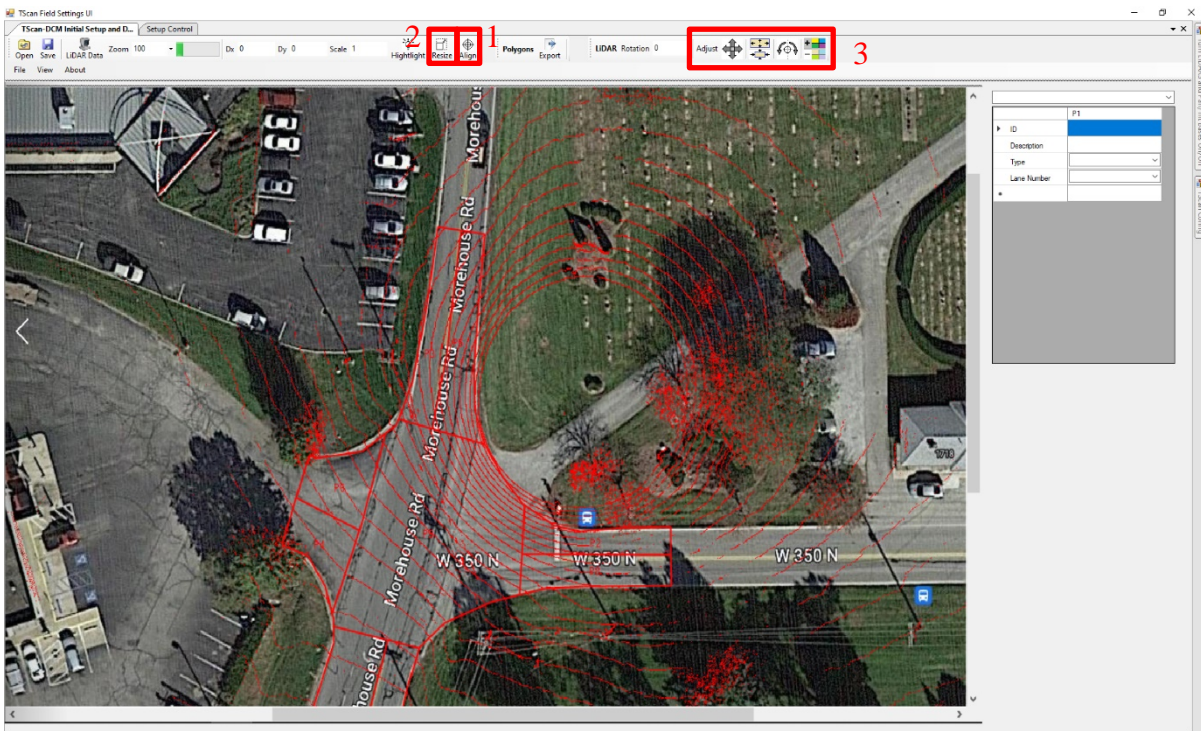


Figure C.30 TScan and orthographic-image alignment.

The created .dcs file brings an automatic scale factor for the orthographic image. In case change in scale is needed, select the *Resize* button<sup>2</sup> and the resize will happen at the same time as the rotation.

#### *Manual Alignment (Adjustments)*

The alignment can be adjusted manually by modifying the four alignment parameters: rotation, scale, vertical shift, and horizontal shift. The alignment command bar<sup>3</sup> in Figure C.30 provides tools to adjust the parameters.

#### *Rotation Adjustment*

Click on *Rotate* to adjust the rotation angle by 0.1 degrees or right click to adjust the rotation angle by 0.01 degrees with the location of the click determining the rotation direction (Figure C.30). The overlay will be updated after clicking. Press and hold the mouse button to repeat the operation 5 times per second and release it to stop.

#### *Scale Adjustment*

Click on *Scale* to adjust the scale factor between the orthographic-image and the TScan data by 0.01 or right click on *Scale* to adjust the scale factor between the orthographic-image and the TScan data by 0.001 with the location of the click determining if the scale will be increased or decreased (Figure C.30). The new overlay will be updated after Press and hold the left mouse button to repeat the operation 5 times per second and release it to stop.

#### *Horizontal and Vertical Shift Adjustment*

Click on *Shift* to adjust the horizontal or vertical shift by 0.1 points per meter or right click *on Shift* to adjust the horizontal or the vertical shift by 0.01 points per meter with the location of the click on the button determining the shift direction (Figure C.30). The overlay will be updated after clicking. Press and hold the left mouse button to repeat the operation five times per second and release it to stop.

### Background Transparency Adjustment

To facilitate the visualization a button to change the background transparency is available. Click on the *Transparency* button by 10% or right click on the *Transparency* button by 5% to adjust the orthographic image, the location of the click on the button will determinate if the transparency will be increased or decreased (Figure C.30). The new overlay will be updated after clicking. Press and hold the left mouse button to repeat the operation 5 times per second and release it to stop.

### Exporting aligned polygons

Once the alignment is completed, click on the *Export*<sup>1</sup> button to save the polygons in the TScan reference system (Figure C.31). This process may take some time.

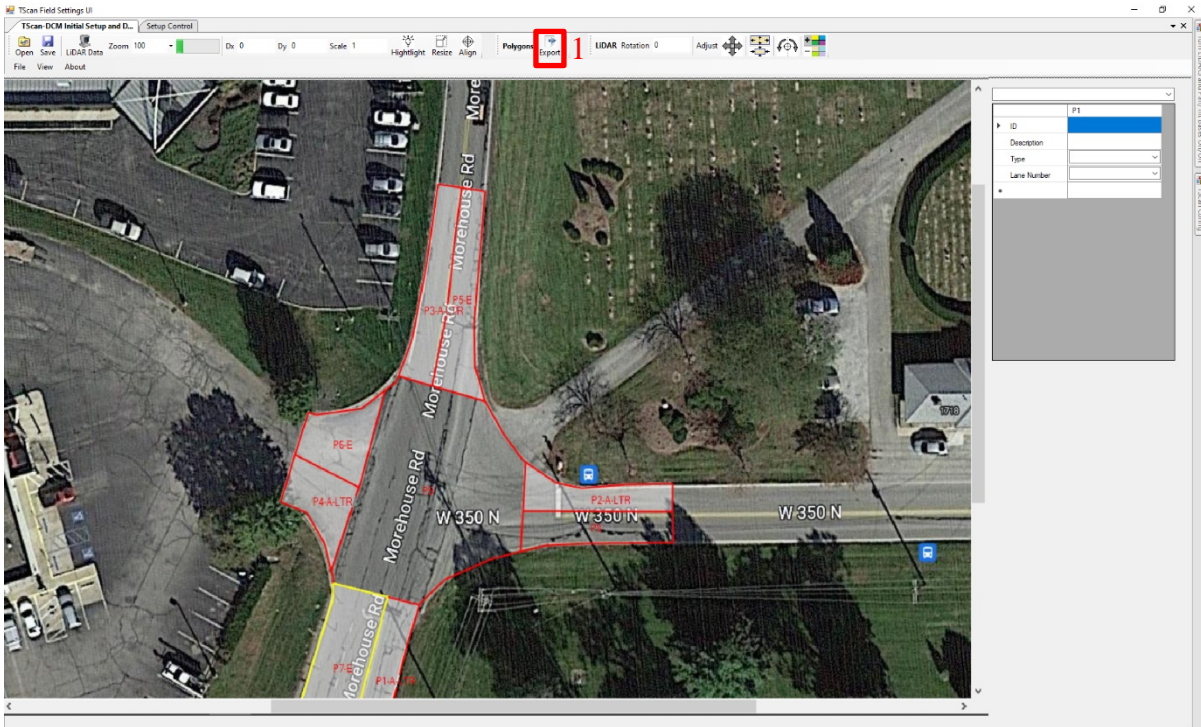


Figure C.31 Display polygons and export.

### C.5.7 Background Identification

Select the control window tab to continue with the data collection process. The software indicates the correct transfer of the polygons by showing a green convection in the buttons. Click on Background Identification to remove the background and isolate only the points belonging to objects. This process takes approximately 20 minutes. The progress bar informs the user when the background is properly identified.

### C.5.8 Data Collection

Click on *Collect Data* to launch the data collection process that does not require user intervention while collecting data. Click on *Stop* to finalize the data collection. The program will continue running for several minutes until the final batch data are processed. A message box will be displayed to inform the user that is safe to exit the program as shown in Figure C.32.

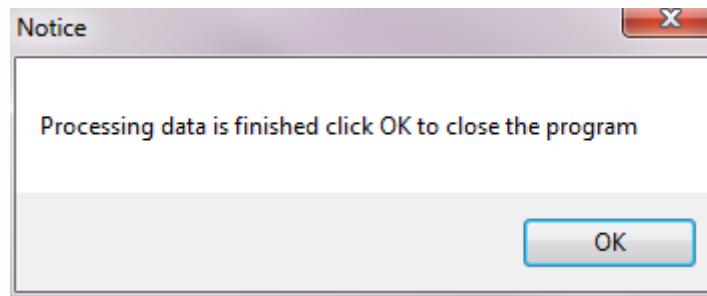


Figure C.32 End calculations.

### *C.5.9 Configuration Window*

Each TScan head is preconfigured when the On-Site Setup program was installed. The configuration panel displays the hardware Ids and the communication port of the main components of the TScan head including:

- Main Tilt
- Pan Minor
- Tilt Minor
- Inertial Measurement Unit
- Power Relay Controller

No user action is required.



## APPENDIX D. CRASH EXPANSION FACTORS

This chapter explains the method to expand the expected crash frequency from the observation period estimate to the estimated annual crash frequency.

### D.1 Introduction to the Concept of Expansion Factors

The TScan system can be used continuously at one intersection for up to four days with a 12-hour observation period per day due to the system's limited battery capacity and human operation requirements. This implies that traffic conflicts can only be observed for a few days at a given location. After recharging the batteries, observations can be repeated multiple times. All the elements of the method are applicable to multiple disjoint periods that may be analyzed together. Even then, the overall observation period is much shorter than a year. A reliable method is thus needed to convert the collected conflict data to a standard road safety measurement namely annual crash frequency. Thus, the need for an expansion factor, defined as the ratio of annual expected crash frequency and expected crash frequency for the observation period.

### D.2 Methodology

This section explains the key concepts and methodology used in estimating expansion factors.

#### D.2.1 Pair of Approaches at Intersections

Crashes at intersections can be categorized in multiple ways. Crashes involving two vehicles are often classified by their manner of collision, traffic maneuvers, or traveling directions related one to another. Possible types of accidents may include rear-end crashes, sideswipe crashes, left-turn crashes, and right-angle crashes. Another categorization method is to separate crashes by their locations concerning intersections. In this case, crashes can be categorized as within-intersection and within-approach, which could be useful in causative analysis. To better fit the method of traffic conflict data collection, which determines a traffic conflict by investigating both the actual and hypothetical trajectories of two involved vehicles, this study adopts a variation of the latter classification method. Instead of focusing on the relative location of the crashes in terms of intersections, this study focuses on the intersection legs from where two involved vehicles approach the intersection before colliding one with another. This manner of classifying crashes better fits the modeling effort by aggregating data into larger samples and still allowing a causative analysis.

Although there are no directly recorded approaches of the two vehicles in the source crash dataset, it is possible to extract the information based on pre-collision traveling directions, pre-collision movements, and the crash narratives of each record. Furthermore, a new crash-type variable named pair of approaches is introduced after the vehicle approaches are identified. According to the bearing of each intersection approach, the pairs of approaches are categorized into three types: same-approach pair, opposite-approach pair, and corner-approach (right-angle) pair. The detailed descriptions of each type of approach pair are as follows:

*Same-approach pairs* refer to the pair of approaches that are the same at one intersection. An example of a typical same-approach pair is illustrated in Figure D.1a

*Opposite-approach pairs* refer to the pair of approaches that belong to the same road but have opposite bearings. The requirement of opposite bearings does not include a geometrically 180-degree difference. As long as the two approaches can be considered opposite from the perspective of

intersection design, they would be claimed as an opposite-approach pair in the frame of this study. An example of a typical opposite-approach pair is shown in Figure D.1b

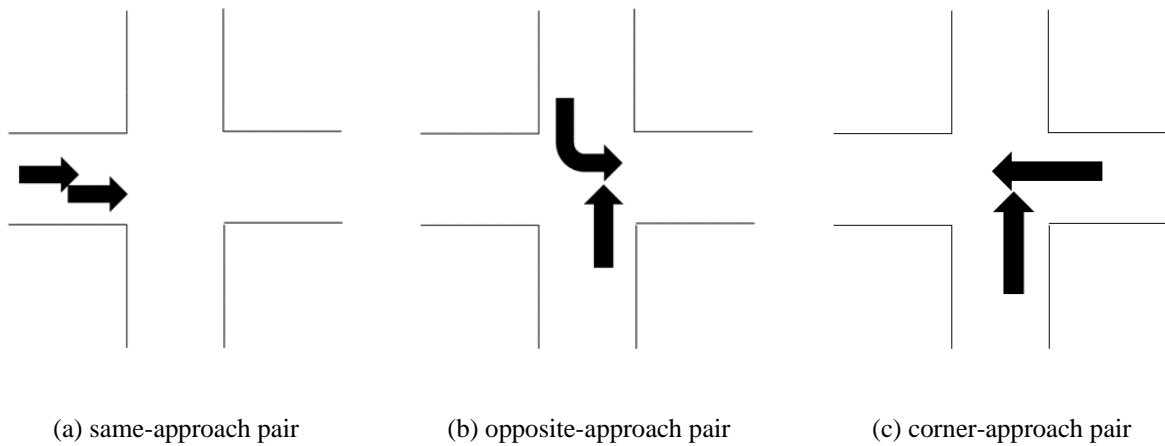


Figure D.1 TScan Examples of a different approach pairs.

Corner-approach pairs refer to the pair of approaches that are not the same and belong to different roads. This corner-approach pair concept is quite similar to the right-angle pair, but just as in the concept of opposite-approach pairs, it does not require a strict 90-degree geometric formation. Any two approaches belonging to different roads at one intersection are called pairs of corner approaches. An example of a typical corner-approach pair is presented in Figure D.1c

One key clarification is that all the definitions of the pairs of approaches above are about the abstract relative positions of approaches and have nothing to do with the movements of vehicles, which is quite efficient in standardizing the data structure.

With the categorization of the three types of intersection crashes above, statistical models were developed separately for these three types to estimate the crash probability.

#### *D.2.2 Hourly Resolution*

To get the desired expansion factors between short and long periods, the crash probability would be analyzed at a very disaggregated level. According to current safety practice and the resolution of potential data at their sources, an hour is a proper unit for estimating crash probability and calculating the expansion factors afterward. Considering the limited traffic conflict data collection time by TScan every day, an hourly level analysis could allow more flexibility in the final expansion factors calculation. In addition, given the presence of emerging data, it is possible to obtain safety-related data at higher resolutions, whereby hourly resolution could strike a good balance between research efforts and data quality.

#### *D.2.3 Logistic Regression*

Given the rarity of crashes, the final dataset for modeling contains a limited number of hourly observations that include one crash, as well as observations that include no crashes. Only crashes involving two vehicles are considered in this study. As mentioned, every observation was related to

a certain pair of approaches within each intersection. In the case of multiple crashes within the same observation, a respective observation is created for each crash to reflect this phenomenon.

The probability that a crash happens during an observation time interval (hour) is expected to be modeled with various explanatory variables regarding traffic, speed, road characteristics, and weather conditions. The total expected crash frequency is to be the sum of the predicted crash frequency (one multiplied by the estimated crash probability) for each hourly observation categorized by specific criteria.

The dataset created above belongs to binary data, which usually occurs due to a nonlinear relationship between crash probability and safety-related factors. Therefore, the most important and typical model used in this case would be the multiple binomial logistic regression model, which has great benefits in explaining how each significant variable impacts the odds ratios. In this study, to model crash probability,  $P$  is specified in the logistic model setting as the probability that a crash happens during one specific observation.

$$\log \left( \frac{P}{1-P} \right) = \eta = X\beta \quad (\text{Eq. D.1})$$

$$P = \frac{e^\eta}{1 + e^\eta} \quad (\text{Eq. D.2})$$

The statistical settings of the proposed models are shown in Equations (D.1) and (D.2), where  $P$  is the probability of a crash happening,  $\eta$  represents the logit link function in this generalized linear model setting,  $X$  represents the safety-related factors, such as traffic, temperature, speed, and geometry, and  $\beta$  is the coefficient to be estimated for the models. SAS was used to estimate the logistic models in this study.

Since historical data is used to estimate the logistic models, this study should be classified as a retrospective study in the frame of statistical analysis. However, to estimate the crash probability, the coefficients of prospective models should be estimated first. Thus, certain derivations are performed here to relate retrospective models to corresponding prospective models.

$$\pi(X) \equiv P(X) = \frac{e^{X\beta}}{1 + e^{X\beta}} \quad (\text{Eq. D.3})$$

$$\theta_0 = P(Z_i = 1 | Y_i = 0) \quad (\text{Eq. D.4})$$

$$\theta_1 = P(Z_i = 1 | Y_i = 1) \quad (\text{Eq. D.5})$$

Assume that there is a prospective model as shown in Equation D.3. However, the input data is sampled retrospectively with Equations D.4 and D.5, where  $Z_i$  is the observation sample including indicators,  $Y_i$  is the binomial response,  $\theta_0$  is the sampling rate for no-event-happen observations, and  $\theta_1$  is the sampling rate for event-happen observations. Given the conditions above, following Bayes' theorem, Equation (3.6) on retrospective model estimation can be derived.

$$\text{logit}\{P(Z_i = 1, X_i)\} = \log \left( \frac{\theta_1}{\theta_0} \right) + X_i\beta \quad (\text{Eq. D.6})$$

$$X_i\beta = \text{logit}\{P(Z_i = 1, X_i)\} - \log \left( \frac{\theta_1}{\theta_0} \right) \quad (\text{Eq. D.7})$$

Through a simple transformation on Equation (D.6), Equation (D.7) is obtained, which shows that an offset value  $\log\left(\frac{\theta_1}{\theta_0}\right)$  should be subtracted from the estimated intercept of retrospective models to obtain the corresponding prospective model's estimation results.

As for this study, according to past research, a 1:100 ratio between crash-happen observations and no-crash-happen observations is adopted to obtain better model estimation results.

#### *D.2.4 Calculation of Expansion Factors*

After the logistic regression models are developed for each type  $k$  of approach movement interactions (same approach, opposite approaches, corner (right-angle) approaches) and their offset values are adjusted for the missing zero-observations, they can be used to calculate the corresponding expansion factors from a short period  $H$  with  $h$ -indexed hourly intervals to year  $Y$  with  $y$ -indexed hourly intervals.

$$EF_{k|T,Y} = \frac{\sum_{y \in Y} P_{ky}}{\sum_{t \in T} P_{kt}} \quad (\text{Eq. D.8})$$

where:

$EF_{k|H,Y}$  = expansion factor from a period  $H$  to year  $Y$  for crashes between approach pair  $k$ ,

$P_{kh}$  = probability of crash between approach pair  $k$  in hour  $h$ ,

$P_{ky}$  = probability of crash between approach pair  $k$  in hour  $y$ .

### **D.3 Data Preparation**

This section explains the scope of this study and each type of data used for developing the logistic regression models

#### *D.3.1 Data Scope*

All selected data sources were used to extract corresponding data for the selected 194 intersections from 2017 to 2019. The 194 intersections were randomly chosen from state-administered intersections in Indiana with their intersection IDs (Figure D.2). Both four-leg and three-leg intersections were included in the sample, as shown in Table D.3

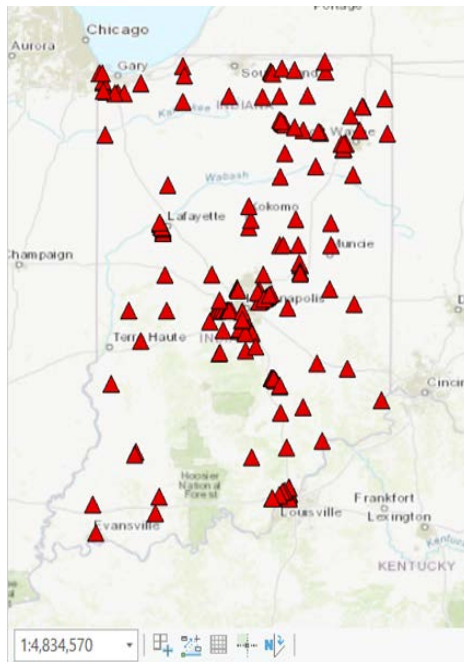


Figure D.2 Selected intersections within Indiana.

Table D.3 Intersections summary by number of legs

Number of Legs	Number of Intersections Selected
4	149
3	45

Based on the previously defined pairs of approaches for intersections, it can be concluded that four-leg intersections have ten different pairs of approaches. In comparison, three-leg intersections have six different pairs of approaches. Therefore, a total number of 1,760 pairs of approaches should be considered in this analysis, out of which 731 are same-approach pairs, 343 opposite-approach pairs, and 686 corner-approach pairs (Table D.4). Additionally, to better represent each intersection, a reasonable bearing value is given to every approach linked to one selected intersection.

Table D.4 Summary of pairs of approaches

Type of Approach Pair	Number in the Sample
Same Approach	731
Opposite Approach	343
Corner Approach	686
Total	1760

In addition, since the selected intersections are state-administered, they are either signalized or two-way stop-controlled (Table D.5), which could be an important factor in crash probability estimation.

Table D.5 Intersections summary by control type

Control Type	Number of Intersections Selected
Signalization	135
Two-Way Stop	59

The three-year temporal scope was adopted to ensure the sample covers adequate temporal variation from year to year. To avoid any unexpected noise from the COVID-19 pandemic in 2020, the years 2017, 2018, and 2019 were selected for this study.

The other adopted data sources are introduced in the following sections.

### D.3.2 Crash Data

The Automated Reporting Information Exchange System (ARIES) was selected as the data source for historical crash data. ARIES is the State of Indiana’s crash repository. Crash data is generated through first-responder crash reports and collected within ARIES. Data is available from 2007 to the present day. Examples of this data include crash details, such as vehicle information, road conditions, crash severity, weather conditions, location, date, and time. The Indiana Traffic Records Coordinating Committee (TRCC), which is made up of numerous agencies, offers continuous input into formulating and updating ARIES. Member agencies of TRCC include the State Police, local and county police departments, INDOT, National Highway Traffic Safety Administration (NHTSA), DNR, BMV, Indiana Criminal Justice Institute (ICJI), Fatal Accident Reporting System (FARS), and the ISP Commercial Vehicle Enforcement Division.

The crash data used in this study was extracted for the selected 194 intersections from 2017 to 2019. Only the crashes involving two vehicles are considered in this dataset. The definition for the range of an intersection is the area within 250 ft of the centroid of the intersection.

Out of the numerous variables in the raw crash table, only the nine most relevant variables are included in this study (Table D.6).

Table D.6 Useful variables in raw crash table

Variable Name	Explanation
MstrRecNbrTxt	Crash ID
State_ID	Intersection ID
TravDirCde1	Pre-collision traveling direction of vehicle 1
TravDirCde2	Pre-collision traveling direction of vehicle 2
PreCollActCde1	Pre-collision action of vehicle 1
PreCollActCde2	Pre-collision action of vehicle 2
collkte	Collision date
CollTimeMilitaryTxt	Collision time
Narratives	Description of the crash

From the table above, the pre-collision traveling directions, pre-collision actions, and narratives are used to identify the approach of each involved vehicle. This process must be completed manually due to the natural language involved and the varying geometries of intersections. Each approach is represented by its bearing value, which should be consistent with the approach-bearing settings in Section D.3.1

Another key point to note is that the collision time in the crash table is listed in Indiana’s current local time with daylight saving correction and thus should be linked properly to the sample. The crash table (after preprocessing) contains the listed variables in Table D.7

Table D.7 Variables in preprocessed crash table

Variable Name	Explanation
MstrRecNbrTxt	Crash ID
State_ID	Intersection ID
Approach1	Approach bearing of vehicle 1
Approach2	Approach bearing of vehicle 2
collkte	Collision date
CollTimeMilitaryTxt	Collision time

### D.3.3 Speed Data

INRIX data will serve as the data source for speed data on the approaches at the temporal resolution of one minute. INRIX serves speed data through JTRP with INDOT to evaluate Indiana’s roadway system. This data source is currently maintained by INRIX but available through JTRP. According to introductions on INRIX websites, the GPS probe vehicle-based speed data is collected every 60 seconds, and the average length of the data collecting segments is 0.5 miles. Because the INRIX speed data is collected from GPS probe vehicles or cell phones, certain time epochs exist without observation, which could be incorporated in this study by creating a missing-value level. The INRIX speed data table includes location (INRIX segment ID), date, time, length, speeds, and a data quality index (a three-level categorical variable indicating the number of probe vehicles used to estimate the speed). However, the speed is not classified by vehicle type. Though INRIX provides speed data with relatively high temporal resolution (consistent one-minute data collection interval), spatial resolution (consistent average 0.5 miles length of the segments), and higher coverage (most of the state roads and some of the local roads), one major challenge of this data source is the relatively large data size and computing demands.

In this study, the corresponding shapefiles for INRIX data from 2017 to 2019 were provided, processed, and linked to intersection approaches using ArcGIS.

The first step of processing INRIX data is extracting the required INRIX segments from the shapefiles provided by INRIX (Figure D.3). These INRIX segments are defined by INRIX, and each segment is assigned a unique ID. However, the shapefiles could be updated at an interval of several months to one year, which may cause extra efforts to be contributed because the INRIX ID of the practically same segment could be changed. This shapefile complexity is especially true for INRIX data before 2021.



Figure D.3 INRIX shapefile and selected intersections.

The INRIX shapefile segments would be related to each approach of selected intersections by their relative locations, and the output table would include the listed variables in Table 4.1.

Table D.8 Variables in INRIX shapefile output table

Variable Name	Explanation
Intersection	Intersection ID
Approach	Approach bearing
XDSegID1	INRIX ID for period (09/20/2016-05/14/2017)
XDSegID2	INRIX ID for period (05/15/2017-10/23/2017)
XDSegID3	INRIX ID for period (10/24/2017-03/29/2018)
XDSegID4	INRIX ID for period (03/30/2018-12/03/2018)
XDSegID5	INRIX ID for period (12/04/2018-04/15/2019)
XDSegID6	INRIX ID for period (04/16/2019-09/08/2019)
XDSegID7	INRIX ID for period (09/09/2019-)

After the INRIX segment IDs were attached to all the approaches, the raw INRIX speed files could be preprocessed to obtain the mean and standard deviation values of desired resolution, which is an hour in this study. The useful variables before preprocessing are listed in Table D.9.

Table D.9 Useful variables in raw INRIX files

Variable Name	Explanation
tstamp	Date and time of this speed record
xdid	INRIX segment ID
speed	Speed value in mi/h

The date and hour values are extracted from the tstamp variable, and the speed values are processed to obtain the mean and standard deviation of speed by INRIX segment ID, date, and hour. The time



zone for INRIX data is GMT (Greenwich Mean Time) and should be considered when attaching INRIX speed data to the sample. The INRIX speed table should contain the variables listed in Table D.10 after preprocessing.

Table D.10 INRIX speed table after preprocessing

Variable Name	Explanation
xdid	INRIX segment ID
Date	Date of the record
Hour	Hour of the record
avgSpd	Average speed
stdSpd	Standard deviation of speed

#### D.3.4 Traffic Volume Data

Three-year average AADT data from the INDOT road inventory for roads within Indiana were obtained for this analysis. The shapefile for the traffic data was also provided and linked to each approach at intersections using ArcGIS. The output table structure for shapefile linking is shown in Table D.11.

Table D.11 Variables in traffic shapefile output table

Variable Name	Explanation
Intersection	Intersection ID
Approach	Approach bearing
CRS_Seg	Segment ID in this road shapefile
Seg_Directions	Number of traffic directions on the segment

Here, the numbers of traffic directions on the segment were extracted from Google Maps because the road segments could contain either two-way traffic or one-way traffic, while the AADT values in the inventory represent the traffic volume for the entire segment. Therefore, the original AADT values are divided by the number of traffic directions to obtain the AADT values for specific approaches. The traffic volume data after being preprocessed include the variables shown in Table D.12.

Table D.12 Traffic volume table after preprocessing

Variable Name	Explanation
State_ID	Intersection ID
Approach	Approach bearing
AADTd	AADT value of the approach

#### D.3.5 Roadway Feature Data

Speed limit data from the INDOT road inventory for all roads within Indiana is used in this study. The shapefile for the speed limit data was provided and related to all the approaches.

Road class data from the INDOT road inventory for state and local roads within Indiana are adopted in this study. The coding rules for road class are Code 1 – Interstates, Code 2 – Principal Arterial, Code 3 – Other Principal Arterials, Code 4 – Minor Arterials, Code 5 – Major Collectors, Code 6 –

Minor Collectors, and Code 7 – Locals. The shapefile for the road class data was also provided and linked to each approach. In addition, for those unsignalized two-way stop-controlled intersections, the corresponding control types were identified by road class and AADT values of each approach. The opposite approaches with higher road class hierarchy and lower AADT are typically identified as stop-controlled approaches. All signalized intersection approaches have signalized control types by nature.

Geometry data at both the intersection and segment levels were collected directly from Google Maps and naturally related to each approach in the sample. Table D.13 shows the table structure of the preprocessed roadway feature data.

Table D.13 Variables in preprocessed roadway feature table

Variable Name	Explanation
State_ID	Intersection ID
Approach	Approach bearing
SPEED_LIMI	Speed limit values in mi/h of the approach
FUNCTIONAL	Functional class of the approach
signaliz	Indicator of signalization
control	Control type of the approach
Landuse	Land-use type of the intersection (U/R)
Close_Int	Indicator for close intersections
LefLane	Indicator for exclusive left-turn lanes
RightLane	Indicator for exclusive right-turn lanes
Tri_Isld	Indicator for right-turn triangular island
Visible	Indicator for marking being easily visible
Cnt	Indicator for close connectors
Num_Lane	Number of continuous lanes
Median	Median type
Shoulder	Shoulder type

### D.3.6 Weather Data

For precipitation data, MPE (Multi-sensor Precipitation Estimates) data from the National Weather Service under the US Department of Commerce is used to obtain the average precipitation value for every hour at each intersection. MPE data is gridded and organized by grid ID, date, and hour. Each intersection in the sample is assigned the closest grid and precipitation data for linking purposes. It was found that only one day’s data was missing for the three years of precipitation data; thus, the values for the specific day were determined by averaging the precipitation values of one day before and one day after. Table D.14 contains the variables in the precipitation table after preprocessing.

Table D.14 Variables in preprocessed precipitation table

Variable Name	Explanation
Pstate_ID	Intersection ID
Pdate	Date of the record
Phour	Hour of the record
Prec	Precipitation value in inch

One issue that must be noted is that the original time zone for MPE data is GMT, and all the date and time values were converted to EST (Eastern Standard Time) for sample assembly.

As for temperature data, NOAA data was obtained to get the hourly temperature for each intersection. NOAA data is station-based data with one or more records within each hour. The data were assigned to each intersection based on the distance between weather stations and intersections. The temperature values were averaged by station ID, date, and hour to match the sampling resolution. In case of missing values, daily or monthly average hourly temperatures were used to fill in the gaps. Similarly, all variables in the temperature table after preprocessing are presented in Table D.15.

Table D.15 Variables in preprocessed temperature table

Variable Name	Explanation
State_ID	Intersection ID
Tdate	Date of the record
Thour	Hour of the record
htemp	Temperature in °F

All temperature data was collected in Indiana’s local standard time without daylight saving correction and can thus be directly used for sample assembly, discussed in Section D.4

#### D.4 Sample Assembly

Starting with a total of 194 intersections and the three-year period from 2017 to 2019, a sample consisting of 412,787 observations can be created for further statistical analysis according to the previously explained methodology.

##### D.4.1 Full Dataset Initialization

The first step for obtaining the desired sample is to initialize an empty table within the full spatial and temporal scope of the study. One empty record is generated for each of the 1760 pairs of approaches at every hour out of the 26280 hours from 2017 to 2019, comprising an empty full dataset of 46,252,800 records.

Approach pair type (SA, OA, or CA), date and hour in EST and EDT time zones, day of week value, and record IDs were added to the table at this stage. The structure of the full table is presented in Table D.16.

Table D.16 Variables in initialized full empty table

Variable Name	Explanation
Intersection	Intersection ID
Approach1	Approach bearing 1
Approach2	Approach bearing 2
Type	Approach pair type
Date	Date in EST
Hour	Hour in EST
Saving	Indicator for Indiana daylight saving period
Sdate	Date in EDT if within daylight saving period
Shour	Hour in EDT if within daylight saving period
Dayofweek	Day of week (Sun to Sat)
ID	Original record ID

#### D.4.2 Crash Observation Identification

Subsequent to the full dataset initialization, crash data was related to the initial empty table first. From the selected 194 intersections within the 2017 to 2019 date range, 4,087 crashes involving two vehicles within 250 ft from the centroid of each intersection were extracted from the ARIES crash system and preprocessed. All crash records were attached to the full table with linking variables: “Intersection,” “Approach 1,” “Approach 2,” “Sdate,” and “Shour.” In this study, four repeated crash records are added to the full dataset as additional records. The newly added variables in the dataset are listed in Table D.17.

Table D.17 Variables added in crash data attachment

Variable Name	Explanation
MstrRecNbrTxt	Crash ID
CrashF	Indicator of a crash

#### D.4.3 No Crash Observation Sampling

For a retrospective binomial data-based study with much more no-crash records than crash records, a sampling process should be undertaken on the no-crash records to ensure the estimated coefficients are within reasonable ranges to improve estimation accuracy.

The rules applied in this study include keeping the final ratio between crash and no-crash records fixed at 1:100 for all three types of pairs of approaches. According to the rules and the actual number of crashes by each type, 291,600 same-approach no-crash records, 39,100 opposite no-crash records, and 78,000 corner-approach records were randomly and separately sampled from the full dataset. Table D.18 summarizes the sampling statistics.

Table D.18 No-crash records sampling statistics

Type of Pair	# of Crash Records	# of No-crash Records Sampled	# of No-crash Records Total
SA	2,916	291,600	19,207,767
OA	391	39,100	9,013,649
CA	780	78,000	18,027,301

After finishing the sampling process, empty samples for same-approach pair, opposite-approach pair, and corner-approach pair types were created.

#### D.4.4 Complete Data Attachment

After three samples consisting of fixed numbers of empty observations were generated, the entirety of the work was attached to the preprocessed data within the empty samples. During this stage, speed, traffic volume, roadway feature, and weather data were attached to each observation. Table D.19 details the newly added variables in the final samples.

Table D.19 Variables newly added in the final samples

Variable Name	Explanation
Prec	Hourly precipitation in inch
htemp	Hourly temperature in °F
AvgSpd 1, 2	Average speed 1, 2 in mi/h
StdSpd 1, 2	Standard deviation of speed 1, 2
Landuse	Land-use type of the intersection
signaliz	Indicator of signalization
FC 1, 2	Road class 1, 2
AADT 1, 2	AADT values 1, 2
Spdlimit 1, 2	Speed limit 1, 2 in mi/h
Control 1, 2	Control type 1, 2
Close_Int 1, 2	Indicator for close intersections 1, 2
LefLane 1, 2	Indicator for exclusive left-turn lanes 1, 2
RightLane 1, 2	Indicator for exclusive right-turn lanes 1, 2
Tri_Isld 1, 2	Indicator for right-turn triangular island 1, 2
Visible 1, 2	Indicator for marking being easily visible 1, 2
Cnt 1, 2	Indicator for close connectors 1, 2
Num_Lane 1, 2	Number of continuous lanes 1, 2
Median 1, 2	Median type 1, 2
Shoulder 1, 2	Shoulder type 1, 2
Year	Year value
Month	Month value in 1, 2...12

## D.5 Model Estimation

As explained in Chapter 3, multiple binomial logistic regression is applied to the three samples and used for estimating the hourly crash probability by approach pair. This chapter elaborates upon the model estimation procedures and includes certain evaluations of the models developed.

### D.5.1 Missing Data Treatment

Prior to beginning the actual modeling steps, it is paramount to check the quality of the data stored within each variable and identify reasonable treatments for data of low quality. This leads to necessary transformations of potentially significant variables during the feature engineering section.

Within this study, missing variable values were first checked and treated. For missing or unreasonable values in the speed limit variable (Table D.20), convenience values were given to the approaches by land-use type. If the intersection is in an urban area, a 30 mi/h value was assigned; otherwise, intersections within rural areas were assigned a value of 55 mi/h.

Table D.20 Statistics of speed limit variable by approach

Value Range	Percentage
15mi/h ~ 60mi/h	80.3%
0mi/h, 99mi/h, or missing values	19.7%

For missing or zero values in the AADT variable (Table D.21), zero values were consistently assigned to the approaches because missing AADT could indicate very little traffic on the approach.

Table D.21 Statistics of AADT variable by approach

Value Range	Percentage
226veh/d ~ 55476veh/d	89.3%
0veh/d or missing values	10.7%

After the speed limit and AADT variables were treated, the remaining variables with missing values in the samples are speed-related (Table D.22). Similarly, zero values were used to fill missing cells, and several corresponding categorical variables with missing-value levels were created to cover this feature.

Table D.22 Statistics of speed-related variables

Variable Name	Value Range	Percentage
avgSpd1&stdSpd1	Non-missing values	53.5%
	Missing values	46.5%
avgSpd2&stdSpd2	Non-missing values	53.6%
	Missing values	46.4%

In addition to missing-value treatment, certain transformations were also conducted to deliver a better performance in model estimation. INDOT seasonal traffic adjustment factors and a convenient list of hourly traffic adjustment factors were jointly adopted to convert AADT values to hourly volumes. Both adjustment factors are included in Appendix A, and the statistics of corresponding volumes are shown in Table D.23.

Table D.23 Statistics of volume-related variables

Variable Name	Mean	Min	Max	N Miss
Vol 1	315.9	0	2,737.7	0
Vol 2	313.4	0	2,737.7	0

Besides the volume conversion, two new variables representing the average approach delays on the approaches were created. Equation 8.9 was used to calculate delay values. Zero values are given if the calculated delays are less than zero. In the instance of missing speed values, zero values are assigned. Additionally, two corresponding categorical variables with missing-value levels were created in parallel. Table D.24 summarizes the statistics of approach delay variables.

$$Approach\ Delay\ j = 3600 \times \left( \frac{0.5\ mi}{avgSpd\ j} - \frac{0.5\ mi}{Spdlimit\ j} \right) (s) \quad (Eq. 8.9)$$

Table D.24 Statistics of delay-related variables

Variable Name	Mean	Min	Max	N Miss
Delay 1	17.9	0	537.5	191914
Delay 2	17.7	0	537.5	191716

D.5.2 Exploratory Analysis

Due to the extreme complexity and large number of observations of this practical problem, it is nearly impossible to obtain any meaningful plots directly from the samples. Therefore, the correlations between converted numerical variables were checked in this section.

Table D.25 Pearson correlation coefficients for numerical variables in SA sample

Pearson Correlation Coefficients, N = 294516 Prob >  r  under H0: Rho=0		
	AdjVol	Delayc
AdjVol	1.00000	0.17244 <.0001
Delayc	0.17244 <.0001	1.00000

According to Table D.25, there is no strong correlation between adjusted volume and continuous delay, which means these two variables can exist concurrently and independently in the final model. Note that the adjusted volume is obtained by dividing the original volume by 1,000.

Both Table D.26 and Table D.27 present a strong correlated relationship between adjusted volume for approach 1 and adjusted volume for approach 2, which suggests including the interaction between these two variables in the models. Here, approach 1 is the approach with a higher AADT value, while approach 2 demonstrates a lower AADT value.

Table D.26 Pearson correlation coefficients for numerical variables in OA sample

Pearson Correlation Coefficients, N = 39491 Prob >  r  under H0: Rho=0				
	AdjVol1	AdjVol2	Delayc1	Delayc2
AdjVol1	1.00000	0.91678 <.0001	0.17446 <.0001	0.19133 <.0001
AdjVol2	0.91678 <.0001	1.00000	0.16596 <.0001	0.19111 <.0001
Delayc1	0.17446 <.0001	0.16596 <.0001	1.00000	0.45932 <.0001
Delayc2	0.19133 <.0001	0.19111 <.0001	0.45932 <.0001	1.00000

Table D.27 Pearson correlation coefficients for numerical variables in CA sample

Pearson Correlation Coefficients, N = 78780 Prob >  r  under H0: Rho=0				
	AdjVol1	AdjVol2	Delayc1	Delayc2
AdjVol1	1.00000	0.51019 <.0001	0.17896 <.0001	0.08596 <.0001
AdjVol2	0.51019 <.0001	1.00000	0.14937 <.0001	0.23995 <.0001
Delayc1	0.17896 <.0001	0.14937 <.0001	1.00000	0.25964 <.0001
Delayc2	0.08596 <.0001	0.23995 <.0001	0.25964 <.0001	1.00000

### *D.5.3 Parameter Estimates*

By using a stepwise method in logistic regression and manually tuning the models, three overall significant logistic models were developed for three types of approach pairs at intersections. The “hplogistic” procedure in SAS was used to fit the models with the “Crash” response.



Table D.28 Parameter estimates for SA model

Parameter	Coefficient Estimate	P-Value
Intercept (with offset)	-10.3268	<.0001
AdjVol	1.1734	<.0001
Speed limit≤30mi/h (ref: 30<spdlimit≤50)	-0.3462	<.0001
Speed limit>50mi/h (ref: 30<spdlimit≤50)	-0.1696	0.0004
27°F≤Temperature<37°F (ref: others)	0.1238	0.0190
Time in [0:00am, 4:00am) on weekends (ref: others)	-1.8377	<.0001
Time in [8:00am, 8:00pm) on weekends (ref: others)	-0.3272	<.0001
Time in [0:00am, 4:00am) on Fridays (ref: others)	-2.1476	<.0001
Time in [12:00pm, 8:00pm) on Fridays (ref: others)	0.2096	0.0014
Time in [8:00pm, 12:00am) on Fridays (ref: others)	-0.9295	<.0001
Time in [0:00am, 4:00am) on Mondays to Thursdays (ref: others)	-2.3397	<.0001
Time in [8:00pm, 12:00am) on Mondays to Thursdays (ref: others)	-1.1850	<.0001
Time in [8:00am, 12:00pm) on Mondays to Thursdays (ref: others)	-0.09308	0.1105
Urban land-use (ref: rural)	-0.09923	0.0222
Delayc	0.003820	<.0001
Delay>10s (ref: delay≤10s)	0.5254	<.0001
Delay value missing (ref: delay≤10s)	0.3443	<.0001
Signalized control (ref: no control)	1.2244	<.0001
Stop control (ref: no control)	0.3589	0.0202
8<Month≤11 (ref: others)	0.08704	0.0407
Rural land-use & no control (ref: others)	0.3527	0.0161

Table D.29 Statistical performance of SA model

Likelihood Ratio Test	AIC	H-L Goodness-of-Fit Test
P-value<.0001	29947 (32,720 for constant model)	P-value=0.0529

Table D.30 Parameter estimates for OA model

Parameter	Coefficient Estimate	P-Value
Intercept (with offset)	-11.1271	<.0001
AdjVol1	1.3560	<.0001
AdjVol2	0.3747	0.3551
AdjVol1*AdjVol2	-0.8802	0.0028
Temperature<=32°F (ref: others)	0.1995	0.1424
Time in [4:00am, 8:00am) on all days (ref: others)	-0.3674	0.0301
Time in [0:00am, 4:00am) on weekends (ref: others)	-0.9645	0.0144
Time in [8:00am, 8:00pm) on weekends (ref: others)	-0.3545	0.0190
Time in [0:00am, 4:00am) on Fridays (ref: others)	-2.2194	0.0275
Time in [12:00pm, 8:00pm) on Fridays (ref: others)	0.2416	0.1810
Time in [8:00am, 12:00pm) on Fridays (ref: others)	-0.5677	0.1196
Time in [0:00am, 4:00am) on Mondays to Thursdays (ref: others)	-1.9469	<.0001
Time in [8:00pm, 12:00am) on Mondays to Thursdays (ref: others)	-0.7169	0.0033
Time in [8:00am, 12:00pm) on Mondays to Thursdays (ref: others)	-0.3832	0.0321
Delayc1	0.003485	0.2605
Delayc2	0.004468	0.1080
Delay1>10s (ref: delay 1<=10s)	0.3890	0.0262
Delay1 value missing (ref: delay 1<=10s)	0.4949	0.0007
Signalized control1 (ref: no control1)	0.3210	0.0483
Stop control1 (ref: no control1)	0.5578	0.0337
Presence of exclusive left-turn lanes1 (ref: absence)	0.1335	0.2740
Urban land-use & signalized intersection (ref: others)	0.2304	0.0518

Table D.31 Statistical performance of OA model

Likelihood Ratio Test	AIC	H-L Goodness-of-Fit Test
P-value<.0001	4237.64 (4389.14 for constant model)	P-value=0.1398

Table D.28 shows the parameter coefficient estimates with mostly significant variables. It can be emphasized that hourly volume positively affects crash probability, while off-peak periods usually negatively affect crash probability. According to Table D.29, the SA model is overall significant and has a good fit for the binomial sample.

Table D.30 details the OA model's coefficient estimates with variables related to both opposite approaches. It can be summarized that both volumes of the approaches have positive effects on crash probability. However, their interaction has a negative effect on crash probability. Additionally, the "Period" variable still has similar effects as in the SA model, and the presence of exclusive left-turn lanes positively contributes to crash probability.

Considering the statistical performance of the OA model (Table D.31), the AIC value is reduced by 151.5 compared to the constant model, which is a considerable reduction. Additionally, the H-L Goodness-of-Fit test's null hypothesis can be confidently accepted due to the large p-value=0.1398. Thus, the OA model's fit can be claimed to be good.

Table D.32 presents the estimation of parameter coefficients for the CA model, which is similar to those for the OA model. Two new parameters worth noting are interactions "AdjVol1\*Delayc1" and "AdjVol2\*Delayc1," which show positive and negative effects on crash probability, respectively. In

the CA model, both the categorical speed limit variables of the two approaches are significant but have opposite effects on crash probability.

Table D.33 reveals the excellent statistical performance of the CA model, whose p-value for the H-L Goodness-of-fit test is as high as 0.2974. In addition, the AIC value was reduced by 456.28, which is an impressive amount considering the complexity of the problem.

Most of the estimated coefficient parameters in the three models presented above may be considered intuitive except few. Given the predictive purpose of these models, counterintuitive parameters are kept in the model to improve the predictive performance of the models.

With all three models developed, it is now possible for users to calculate the actual expansion factors of a specific intersection approach pair for given periods within one year.

Table D.32 Parameter estimates for CA model

Parameter	Coeff. Estimate	P-Value
Intercept (with offset)	-10.9751	<.0001
AdjVol1	0.7226	<.0001
AdjVol2	1.3517	0.0008
AdjVol1*AdjVol2	-0.7253	0.0179
Speed limit1<=30mi/h (ref: spdlimit1>50)	0.2634	0.0773
30<Speed limit1<=50mi/h (ref: spdlimit1>50)	0.1664	0.0863
Speed limit2<=30mi/h (ref: spdlimit2>50)	-0.3416	0.0010
30<Speed limit2<=50mi/h (ref: spdlimit2>50)	-0.3635	<.0001
Temperature<=32°F (ref: others)	0.08808	0.3775
Time in [4:00am, 8:00am) on all days (ref: others)	-0.1366	0.2731
Time in [0:00am, 4:00am) on weekends (ref: others)	-1.0702	0.0012
Time in [8:00am, 8:00pm) on weekends (ref: others)	-0.2768	0.0114
Time in [0:00am, 4:00am) on Fridays (ref: others)	-1.9772	0.0056
Time in [12:00pm, 8:00pm) on Fridays (ref: others)	0.2029	0.1272
Time in [8:00pm, 12:00am) on Fridays (ref: others)	-0.4801	0.1250
Time in [8:00am, 12:00pm) on Fridays (ref: others)	0.3779	0.0280
Time in [0:00am, 4:00am) on Mondays to Thursdays (ref: others)	-1.8817	<.0001
Time in [8:00pm, 12:00am) on Mondays to Thursdays (ref: others)	-0.7861	<.0001
Time in [8:00am, 12:00pm) on Mondays to Thursdays (ref: others)	-0.1546	0.1976
Delayc1	0.007090	0.0198
AdjVol1*Delayc1	0.005732	0.0635
AdjVol2*Delayc1	-0.02325	0.0001
Delayc2	0.001461	0.3673
Delay1>10s (ref: delay 1<=10s)	0.5222	<.0001
Delay1 value missing (ref: delay 1<=10s)	0.3190	0.0020
Urban land-use (ref: rural)	-0.5491	0.0196
Signalized intersection (ref: unsignalized)	0.3798	0.0008
2<Month<=5 (ref: others)	-0.1292	0.1405
Urban land-use & signalized intersection (ref: others)	0.7283	0.0033

Table D.33 Statistical performance of CA model

Likelihood Ratio Test	AIC	H-L Goodness-of -Fit Test
P-value<.0001	456.28 (8753.84 for constant model)	P-value=0.2974

*D.5.4 Sensitivity Analysis*

This section conducts sensitivity analysis for the three types of models developed. Marginal effects are calculated for each parameter and compared. It is worth mentioning that this analysis was conducted before the intercept was adjusted for implementation purposes.

Table D.34 shows the estimated mean, minimum, and maximum marginal effects of all parameters in the SA model. Dummy variables were created for the categorical variables and interaction terms for the convenience of calculation. The calculated marginal effects represent the extent of estimated crash probability change when the corresponding parameter value is increased by one unit. It can be concluded that an increment of 1,000 veh/h in volume and a change from other control types to signalized control would greatly increase the estimated crash probability. On the contrary, a change from any other periods to [0:00am,4:00am) on Mondays to Thursdays leads to the most significant decrease in crash probability estimation.

According to the marginal effects for the OA model in Table D.35, an increment of 1,000 veh/h in the volume of the approach with a higher AADT would have the largest positive effect on estimated crash probability among all parameters. Additionally, a time change to [0:00am, 4:00am) on all weekdays would reduce the estimated crash probability to the greatest extent.

Table D.36 includes the marginal effects of the CA model. Unlike previous models, the adjusted volume of the approach with a lower AADT increases the estimated crash probability the most through a one-unit increment. Still, an hourly interval change to time in [0:00am, 4:00am) on weekdays reduces the estimated crash probability by the greatest amount.

Table D.34 Marginal effects for SA model

Parameter	Mean	Min	Max
AdjVol	0.0113342	0.000158736	0.2362055
Speed limit<=30mi/h (ref: others)	-0.0033434	-0.0696777	-0.000046825
Speed limit>50mi/h (ref: others)	-0.0016378	-0.0341326	-0.000022938
27°F<=Temperature<37°F (ref: others)	0.0011960	0.000016749	0.0249237
Time in [4:00am, 8:00am) on all days (ref: others)	-0.0028527	0.0594508	-0.000039952
Time in [0:00am, 4:00am) on weekends (ref: others)	-0.0177502	-0.3699168	-0.000248593
Time in [8:00am, 8:00pm) on weekends (ref: others)	-0.0031605	-0.0658652	-0.000044263
Time in [0:00am, 4:00am) on Fridays (ref: others)	-0.0207439	-0.4323042	-0.000290519
Time in [12:00pm, 8:00pm) on Fridays (ref: others)	0.0020250	0.000028360	0.0422006
Time in [8:00pm, 12:00am) on Fridays (ref: others)	-0.0089776	-0.1870932	-0.000125731
Time in [0:00am, 4:00am) on Mondays to Thursdays (ref: others)	-0.0225996	-0.4709781	-0.000316509
Time in [8:00pm, 12:00am) on Mondays to Thursdays (ref: others)	-0.0114461	-0.2385388	-0.000160304
Time in [8:00am, 12:00pm) on Mondays to Thursdays (ref: others)	-0.000899040	-0.0187361	-0.000012591
Urban land-use (ref: others)	-0.000958422	-0.0199736	-0.000013423
Delayc	0.000036901	5.1679503E-7	0.000769011
Delay>10s (ref: others)	0.0050745	0.000071069	0.1057528
Delay value missing (ref: others)	0.0033254	0.000046573	0.0693019
Signalized control (ref: others)	0.0118268	0.000165636	0.2464721
Stop control (ref: others)	0.0034663	0.000048545	0.0722372
8<Month<=11 (ref: others)	0.000840757	0.000011775	0.0175214
Rural land-use & no control (ref: others)	0.0034072	0.000047718	0.0710055

Table D.35 Marginal effects for OA model

<b>Parameter</b>	<b>Mean</b>	<b>Min</b>	<b>Max</b>
AdjVol1	0.0132140	0.000552174	0.1796781
AdjVol2	0.0036520	0.000152607	0.0496584
AdjVol1*AdjVol2	-0.0085782	-0.1166421	-0.000358457
Temperature<=32°F (ref: others)	0.0019442	0.000081243	0.0264364
Time in [4:00am, 8:00am) on all days (ref: others)	-0.0035805	-0.0486860	-0.000149619
Time in [0:00am, 4:00am) on weekends (ref: others)	-0.0094029	-0.1278566	-0.000392920
Time in [8:00am, 8:00pm) on weekends (ref: others)	-0.0034548	-0.0469769	-0.000144366
Time in [0:00am, 4:00am) on Fridays (ref: others)	-0.0216278	-0.2940859	-0.000903765
Time in [12:00pm, 8:00pm) on Fridays (ref: others)	0.0023543	0.000098381	0.0320134
Time in [8:00am, 12:00pm) on Fridays (ref: others)	-0.0055302	-0.0751976	-0.000231092
Time in [0:00am, 4:00am) on Mondays to Thursdays (ref: others)	-0.0189711	-0.2579611	-0.000792749
Time in [8:00pm, 12:00am) on Mondays to Thursdays (ref: others)	-0.0069862	-0.0949954	-0.000291933
Time in [8:00am, 12:00pm) on Mondays to Thursdays (ref: others)	-0.0037344	-0.0507783	-0.000156048
Delayc1	0.000033958	1.4189991E-6	0.000461744
Delayc2	0.000043539	1.8193589E-6	0.000592021
Delay1>10s (ref: others)	0.0037908	0.000158409	0.0515463
Delay1 value missing (ref: others)	0.0048232	0.000201549	0.0655842
Signalized controll1 (ref: others)	0.0031282	0.000130719	0.0425361
Stop controll1 (ref: others)	0.0054355	0.000227134	0.0739095
Presence of exclusive left-turn lanes1 (ref: others)	0.0013014	0.000054382	0.0176959
Urban land-use & signalized intersection (ref: others)	0.0022451	0.000093818	0.0305284

Table D.36 Marginal effects for CA model

<b>Parameter</b>	<b>Mean</b>	<b>Min</b>	<b>Max</b>
AdjVol1	0.0070231	0.000181193	0.1702325
AdjVol2	0.0131279	0.000338691	0.3182045
AdjVol1*AdjVol2	-0.0070468	-0.1708059	-0.000181803
Speed limit1<=30mi/h (ref: others)	0.0025596	0.000066037	0.0620423
30<Speed limit1<=50mi/h (ref: others)	0.0016163	0.000041700	0.0391776
Speed limit2<=30mi/h (ref: others)	-0.0033178	-0.0804184	-0.000085596
30<Speed limit2<=50mi/h (ref: others)	-0.0035303	-0.0855703	-0.000091080
Temperature<=32°F (ref: others)	0.000855427	0.000022069	0.0207345
Time in [4:00am, 8:00am) on all days (ref: others)	-0.0013260	-0.0321404	-0.000034210
Time in [0:00am, 4:00am) on weekends (ref: others)	-0.0104019	-0.2521310	-0.000268364
Time in [8:00am, 8:00pm) on weekends (ref: others)	-0.0026882	-0.0651577	-0.000069353
Time in [0:00am, 4:00am) on Fridays (ref: others)	-0.0192053	-0.4655149	-0.000495486
Time in [12:00pm, 8:00pm) on Fridays (ref: others)	0.0019706	0.000050841	0.0477660
Time in [8:00pm, 12:00am) on Fridays (ref: others)	-0.0046524	-0.1127693	-0.000120030
Time in [8:00am, 12:00pm) on Fridays (ref: others)	0.0036730	0.000094762	0.0890302
Time in [0:00am, 4:00am) on Mondays to Thursdays (ref: others)	-0.0182643	-0.4427047	-0.000471207
Time in [8:00pm, 12:00am) on Mondays to Thursdays (ref: others)	-0.0076337	-0.1850313	-0.000196944
Time in [8:00am, 12:00pm) on Mondays to Thursdays (ref: others)	-0.0015019	-0.0364031	-0.000038747
Delayc1	0.000068845	1.7761545E-6	0.0016687
AdjVol1*Delayc1	0.000055688	1.4367041E-6	0.0013498
AdjVol2*Delayc1	-0.000225745	-0.0054718	-5.824094E-6
Delayc2	0.000014205	3.6648402E-7	0.000344316
Delay1>10s (ref: others)	0.0050745	0.000130919	0.1230003
Delay1 value missing (ref: others)	0.0031012	0.000080008	0.0751689
Urban land-use (ref: others)	-0.0053427	-0.1295005	-0.000137838
Signalized intersection (ref: others)	0.0036891	0.000095176	0.0894186
2<Month<=5 (ref: others)	-0.0012554	-0.0304294	-0.000032389
Urban land-use & signalized intersection (ref: others)	0.0070855	0.000182801	0.1717432

## About the Joint Transportation Research Program (JTRP)

On March 11, 1937, the Indiana Legislature passed an act which authorized the Indiana State Highway Commission to cooperate with and assist Purdue University in developing the best methods of improving and maintaining the highways of the state and the respective counties thereof. That collaborative effort was called the Joint Highway Research Project (JHRP). In 1997 the collaborative venture was renamed as the Joint Transportation Research Program (JTRP) to reflect the state and national efforts to integrate the management and operation of various transportation modes.

The first studies of JHRP were concerned with Test Road No. 1 — evaluation of the weathering characteristics of stabilized materials. After World War II, the JHRP program grew substantially and was regularly producing technical reports. Over 1,600 technical reports are now available, published as part of the JHRP and subsequently JTRP collaborative venture between Purdue University and what is now the Indiana Department of Transportation.

Free online access to all reports is provided through a unique collaboration between JTRP and Purdue Libraries. These are available at <http://docs.lib.purdue.edu/jtrp>.

Further information about JTRP and its current research program is available at <http://www.purdue.edu/jtrp>.

## About This Report

An open access version of this publication is available online. See the URL in the citation below.

Tarko, A. P., Romero, M. A., Bandaru, V. K., & Shi, X. (2023). *Guidelines for evaluating safety using traffic encounters: Proactive crash estimation on roadways with conventional and autonomous vehicle scenarios* (Joint Transportation Research Program Publication No. FHWA/IN/JTRP-2023/02). West Lafayette, IN: Purdue University. <https://doi.org/10.5703/1288284317587>