



NATIONAL CENTER FOR TRANSPORTATION SYSTEMS PRODUCTIVITY AND MANAGEMENT

# Automated Data Collection for Origin/Destination Studies of Freight Movement

Contract # DTRT12GUTC12 with USDOT Office of the Assistant Secretary for Research and Technology (OST-R)

Final Report

January 2017

Principal Investigator: Amr A. Oloufa, Ph.D.



**National Center for Transportation Systems  
Productivity and Management**  
O. Lamar Allen Sustainable Education Building  
788 Atlantic Drive, Atlanta, GA 30332-0355  
P: 404-894-2236 F: 404-894-2278  
nctspm@ce.gatech.edu nctspm.gatech.edu



*DISCLAIMER*

*The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.*

## **ACKNOWLEDGEMENTS**

The author thanks the Florida and Georgia Departments of Transportation for sponsoring this research project. Special thanks to the Marion and Sumter Maintenance Offices for assisting with this research.

## EXECUTIVE SUMMARY

Origin and destination (O–D) include the start and end points and times of any vehicular trip. These data are valuable to traffic modelers and transportation planners. The collection of O–D data usually comes from surveys, visual counts, classifier counts, or other methods. These methods of collection tend to be expensive and time consuming. The aim of this research project is to develop a novel method of automated real time O–D data collection that is reliable, inexpensive, and portable using a mix of commercial off-the-shelf hardware and custom software. As such, the researchers conducted an automated license plate reading methodology.

The first step was to identify the length of highway in which cameras could be installed such that license plates would be in view and three stations could be set up to get the maximum interpretation of origins and destinations. The second step included selection of the appropriate hardware configuration (i.e., camera and trigger systems) and a solar power design for each location that would be cost effective and safe. The third step was the installation of the hardware and solar power components. The final step was the development of software to process and interpret the collected O–D data.

The initial plan was to use triggering devices to detect trucks and take snapshots of the rear license plate. This proved to be quite difficult due to fluctuations in the speed and lengths of the trucks. Also, truck tires often direct dirt and debris onto license plates, rendering them unreadable. The solution was to turn the cameras around and capture the front license plates. The plates would be processed by optical character recognition (OCR) software and the results of the OCR would be stored in a database. The results were then analyzed using database and pattern-matching techniques to show when a truck entered and left this small transportation network.

The researchers utilized a three-step approach to match license plate reads between locations. The first attempt to match was a simple database query based on exact matches within a selected time frame. The second attempt was to use a fuzzy match using database queries with wildcards and substitutions for common OCR misreads. The third attempt used partial matches via coded algorithms. Any reads that were unmatched either were assumed to have entered or did enter the network between locations.

The findings from the University of Central Florida's (UCF's) Location 3 two-week data collection indicate that historical traffic data collected by FDOT in 2015 is consistent with the UCF system in terms of weekday traffic trends. Traffic counts collected by the UCF system at Location 3 were within 107 percent of FDOT's weekday average, and truck counts were within approximately 109 percent of FDOT's average in year 2015.

# TABLE OF CONTENTS

<b>Disclaimer .....</b>	<b>ii</b>
<b>Acknowledgements .....</b>	<b>iii</b>
<b>Executive Summary .....</b>	<b>v</b>
<b>List of Tables .....</b>	<b>viii</b>
<b>List of Figures.....</b>	<b>ix</b>
<b>Chapter 1 Introduction.....</b>	<b>1</b>
<b>1.1 Background Statement .....</b>	<b>1</b>
<b>1.2 Objectives and Supporting Tasks.....</b>	<b>1</b>
<b>Chapter 2 Literature Review .....</b>	<b>0</b>
<b>2.1 Automatic License Plate Recognition.....</b>	<b>0</b>
2.1.1 Usage.....	0
2.1.2 Restrictions on the Use of ALPR.....	1
<b>2.2 Bluetooth Devices as Traffic Probes.....</b>	<b>1</b>
<b>2.3 Cell Phone Tracking .....</b>	<b>2</b>
<b>2.4 Toll Tag Readers.....</b>	<b>2</b>
<b>Chapter 3 Methodology .....</b>	<b>4</b>
<b>3.1 Location Selection .....</b>	<b>4</b>
<b>3.2 Hardware Selection.....</b>	<b>6</b>
<b>3.3 Solar Power System .....</b>	<b>6</b>
3.3.1 Alternative Foundation Design .....	6
3.3.2 Installation of Solar Power Panels at the Three Sites .....	6
<b>3.4 Software and Analysis Tools.....</b>	<b>9</b>
3.4.1 Software Development.....	9
3.4.2 Hardware and Software Modification.....	10
3.4.3 Management of Origin and Destination Data and Analyses.....	11
3.4.4 Database Design and Programming.....	14
3.4.5 Generating a Simulated Dataset to Test the Software .....	17
<b>Chapter 4 Data Collection .....</b>	<b>20</b>
<b>4.1 Summary.....</b>	<b>20</b>
<b>4.2 Background .....</b>	<b>23</b>
<b>4.3 Findings.....</b>	<b>27</b>
4.3.1 Monday 10/31/2016.....	31
4.3.2 Wednesday 11/02/2016.....	34
4.3.3 Friday 11/04/2016.....	37
4.3.4 Saturday 11/05/2016.....	40
<b>4.4 License Plate Matching (MODDA) Results.....</b>	<b>43</b>
<b>Chapter 5 Conclusions.....</b>	<b>45</b>
<b>References.....</b>	<b>46</b>
<b>Appendix-A Sample Codes.....</b>	<b>48</b>

<b>A.1</b>	<b>Code – Camera Software.....</b>	<b>48</b>
<b>A.2</b>	<b>Code – Trigger Software .....</b>	<b>51</b>
	<b>Appendix-B Additional Photos .....</b>	<b>57</b>

## LIST OF TABLES

Table 3-1: OD_PlateRead .....	14
Table 3-2: OD_Location .....	15
Table 3-3: OD_PlateReadImport .....	15
Table 3-4: OD_Roadway .....	15
Table 3-5: OD_Direction .....	15
Table 3-6: OD_RoadwayLinks .....	16
Table 3-7: OD_Match .....	16
Table 3-8: OD_MatchType .....	17
Table 3-9: Assignment Groups of Matching Types .....	18
Table 4-1: Summary of Traffic Counts .....	20
Table 4-2: Plate Recognition Initial Findings .....	21
Table 4-3: Historical AADT Report (FDOT) .....	28
Table 4-4: Hourly Traffic Counts at NB Lanes on I-75 in September 2015 (FDOT) .....	29
Table 4-5: Matched Plates .....	43

## LIST OF FIGURES

Figure 3-1: Overview of Location 1 .....	4
Figure 3-2: Overview of Location 2 .....	5
Figure 3-3: Overview of Location 3 .....	5
Figure 3-4: Solar Power System – Location 1 .....	7
Figure 3-5: Solar Power System – Location 2 .....	8
Figure 3-6: Solar Power System – Location 3 .....	9
Figure 4-1: Daily Traffic Trends for Days Spanning Two Weeks .....	22
Figure 4-2: Daily Truck Percentages .....	22
Figure 4-3: Monday (10/31) – Truck Percentage Moving Average .....	22
Figure 4-4: Friday (11/4) – Truck Percentage Moving Average .....	23
Figure 4-5: Location 3 on I-75 at Warm Springs Ave. ....	24
Figure 4-6: Capture System Mounted on Overpass .....	25
Figure 4-7: Sensor Aimed at Traffic .....	26
Figure 4-8: Bracket Close-up Showing Sensor, Camera, and Processor .....	26
Figure 4-9: FDOT Telemetered Traffic Monitoring Site (Red Point) .....	27
Figure 4-10: Typical Monday in October 2015 (FDOT) .....	31
Figure 4-11: Vehicle Counts per Time Interval (5-minute) .....	31
Figure 4-12: Number of Cars, Trucks, and Total Vehicles (Cumulative) .....	32
Figure 4-13: Truck Percentages .....	33
Figure 4-14: Hourly Truck Volumes .....	33
Figure 4-15: Typical Wednesday in November 2015 (FDOT) .....	34
Figure 4-16: Vehicle Counts per Time Interval (5-minute) .....	34
Figure 4-17: Number of Cars, Trucks, and Total Vehicles (Cumulative) .....	35
Figure 4-18: Truck Percentages .....	36
Figure 4-19: Hourly Truck Volumes .....	36
Figure 4-20: Typical Friday in November 2015 (FDOT) .....	37
Figure 4-21: Vehicle Counts per Time Interval (5-minute) .....	37
Figure 4-22: Number of Cars, Trucks, and Total Vehicles (Cumulative) .....	38
Figure 4-23: Truck Percentages .....	39
Figure 4-24: Hourly Truck Volumes .....	39
Figure 4-25: Typical Saturday in November 2015 (FDOT) .....	40
Figure 4-26: Vehicle Counts per Time Interval (5-minute) .....	40
Figure 4-27: Number of Cars, Trucks, and Total Vehicles (Cumulative) .....	41
Figure 4-28: Truck Percentages .....	42
Figure 4-29: Hourly Truck Volumes .....	42
Figure 4-30: 5-Minute Intervals – Average Speeds in mph .....	44
Figure 4-31: 15-Minute Intervals – Average Speeds in mph .....	44

# Chapter 1 INTRODUCTION

## 1.1 BACKGROUND STATEMENT

The collection of reliable origin–destination (O–D) data for freight has profound consequences for a large range of applications in both planning and operations. Indeed, in both research and practice, a large number of applications rely heavily on such data. The demand for this information is also expected to increase with the modernization of the Panama Canal and the related implications on freight in United States ports.

The challenge over the years has been the tremendous expense in both time and cost associated with the collection of such data. This is true for both regular and freight traffic where O–D data collection has relied heavily either on manual counts and interviews, or on expensive intelligent transportation system (ITS) technologies that require excessive Maintenance of Traffic MOT and lane stoppages for installation.

Previous ITS technologies for O–D data collection relied on vehicles equipped with sensors such as radio-frequency identification (RFID) or Bluetooth technologies. This limits their applicability and also does not allow for distinguishing regular traffic from freight traffic, which makes it harder to separate freight traffic.

In an exploratory project that was funded (and recently completed) by the Florida Department of Transportation (FDOT), the principal investigator and his research team developed a novel approach for tracking trucks using their license plate numbers. In that project, the research team selected two gantries on Hwy 528 near Orlando, Florida. Both gantries were fitted with cameras aimed at capturing truck license plates. The system was triggered by a height sensor (to capture trucks only), and image data were then read using advanced optical character recognition (OCR) techniques and the resulting license plates were stored. The data were then used to compare results at the next gantry location and thereby measure speed and travel time of each truck. This information can then be used in an origin–destination model.

## 1.2 OBJECTIVES AND SUPPORTING TASKS

That FDOT project, which was completed recently, clearly demonstrated the feasibility of the license plate reader approach; however, more work needs to be done before a system can be adopted for wide application. The remaining challenges are the following:

- Due to the lack of available 110V power, the system could only be tested for a few hours at a time. For the same reason, a very limited amount of data was collected.
- Most freeway gantries do not have available power. The ones that do may have “switched” systems that only operate at night. Most of those do not run on 110V, which would require costly transformers. Also, wiring each gantry for 110V power is relatively expensive.
- Power consumption of cameras was not a design issue as the assumption was that power is readily available and cameras require relatively low wattage.
- The system used an expensive triggering device. Due to cost considerations, a single trigger was used for all lanes. This means that only one lane can be triggered at any given

time. While this worked fine for Phase I, clearly this approach is limited in busy corridors where multiple trucks can be using several lanes simultaneously.

- A relatively long time is needed for system installation, as wiring the system was time-consuming.

In this project, these limitations were addressed and the researchers expanded the field test to three locations covering a total of nine lanes. These locations were selected in consultation with FDOT. Data will be collected for a full year to account for seasonal changes, which will enable the research team to fully evaluate the approach on a large data set.

The objectives of this O–D project are as follows:

- *To read and capture heavy-truck license plate numbers that are located on the front of a truck–trailer combination during both daylight and nighttime hours.* The captured numbers would be sent to a local computer and stored for batch processing later. Imaging software that includes optical character recognition capabilities would be used to obtain license plate number matches as the vehicle moves through the identified corridor and passes another O–D technology location. Software for the read-and-capture system was developed during this project by the research team.
- *To develop a database-based system for O–D data archival and analysis.*
- *To design and deploy a hardware system for rapid installation in a large number of installation scenarios on highway gantries.* To accomplish the license plate number read and data capture, the deployment strategies were based on site infrastructure and the recommendation of the FDOT District 5 Maintenance Office.
- *To develop a solar-based system for continuous power.*

Cameras were mounted on overpasses at Locations 1 and 3; and on a gantry at Location 2 over every two travel lanes. They were used to capture images of the front license plates. The cameras were triggered by a laser sensor mounted underneath the cameras. The radar sensor selected for use is an expensive sensor that can accurately measure the distance from the ground. To avoid capturing photographs of light trucks and passenger vehicles, the sensor would constantly measure the height of each passing vehicle and trigger the system to save the last multiple photos of vehicles of certain heights ( $\geq 100$  inches).

## Chapter 2 LITERATURE REVIEW

### 2.1 AUTOMATIC LICENSE PLATE RECOGNITION

Automatic license plate recognition (ALPR) is the process of acquiring still images or frames of a video feed and detecting the license plate using software processing techniques.

#### 2.1.1 Usage

ALPR technologies are used in enforcement applications and data collection efforts (Rossetti & Baker, 2000). Data collection includes origin–destination studies, traffic-flow studies, high-occupancy-vehicle analysis, and weigh-in-motion systems. ALPR has been used in origin–destination studies (The Traffic Group, Inc., 2011). The Ohio Department of Transportation (ODOT) performed a proof-of-concept study using ALPR technologies for the purpose of demonstrating this as a data collection method for high-speed highways and as an alternate for low-volume (<25,000 AADT [annual average daily traffic]) traffic stops to administer surveys (Giamo, 2011). In this study, one camera was placed on the median to capture plates on vehicles driving on the inside lane and one camera was placed on the shoulder to capture those on the outside lane. Two stations, spaced roughly 21 miles apart, simultaneously recorded license plates in the northbound travel direction.

Issues that ODOT’s study revealed include light level, camera height, vehicle reflectors, platooning, and illegible plates. The data were then cleaned using manual methods to verify the OCR correctly read the plates. In cases where there were no reads or partial reads, visual inspection was used to generate the license plate. During this process, the vehicles were also classified visually. The cleaned data were then passed through a four-step process of matching the license plates between stations to identify the following:

- Exact matches
- Similar matches – using common substitutions such as 2 and Z
- Wildcard match – used only on manually edited records (e.g., ABC1\* matches ABC123 or ABC?23 matches ABC123)
- Partial match – investigator-selected four-character minimum of coinciding sequential characters to form a match

All matches were evaluated based on a realistic amount of time difference between reads. ODOT’s study produced the following conclusions:

- ALPR provides a useful method for determining through-travel percents.
- Careful expansion/weighting is key to obtaining good results.
- Little manual cleanup/editing is necessary (especially if trucks will not be segregated).
- Rear plates provide the best data.
- Further development of the field equipment, such as fixed mountings and dedicated trailers, can improve the reliability to potentially alleviate the need for continuous monitoring.
- The technology also can be used to obtain travel times, weave movements, and dwell times.

### **2.1.2 Restrictions on the Use of ALPR**

Several states have placed restrictions on the use of ALPR. The State of Maine has limited the legal use of ALPR to law enforcement agencies, the Department of Transportation, and the Maine Turnpike (Maine Legislature, 2009; 2003). Maine's legislation not only criminalizes the use of ALPR, but also sets guidelines for the purpose for the data collection, on which agencies can access the collected data, and how long the data can be retained.

New Hampshire has also legislated the use of ALPR. New Hampshire's law is more restrictive, allowing it only for the use of law enforcement and toll collection, and for securing three specific bridges in Portsmouth (New Hampshire Legislature, 2007).

The State of New Jersey has set guidelines (Dow, 2010) for the use of ALPR technology in law enforcement, stating that ALPR can be used to cross reference against a list of targeted license plates, or the read tags can be stored with location information to be cross referenced later. In the case where license plate reads are stored for later processing, the data are restricted to use by law enforcement only and must be purged after a reasonable amount of time (Giamo, 2011).

## **2.2 BLUETOOTH DEVICES AS TRAFFIC PROBES**

Bluetooth devices, such as cell phones, in-vehicle hands-free devices, headsets, mobile GPS systems, and vehicle navigation systems, broadcast a unique hardware ID when enabled. These devices can be detected by roadside monitoring sensors. A distinct advantage that Bluetooth sensing has is that the device IDs are not registered to a user, which maintains anonymity (Houston Transtar, n.d.). Tracking these devices and their hardware signatures over a road network can provide real-time traffic monitoring and data for origin–destination studies (Resource Systems Group, 2011). One such O–D study was performed by the Resource Systems Group for FDOT to analyze travel movements along State Road 23. The Resource Systems Group's website states, "FDOT used the results of this work to verify the current travel patterns observed in their travel demand model and as an after model adjustment tool for future year forecasts."

Although Bluetooth sensing can be a promising and anonymous methodology for collecting probe data, it should be used in conjunction with other reporting methodologies (Rescot, 2011). Because it is dependent on the drivers to have their Bluetooth devices enabled, having an adequate sample size may not always be guaranteed.

Rescot examined the reliability of Bluetooth sensing by changing antenna types, cell phone placement in the vehicle, lateral setback of the antenna, vertical elevation of the antenna, and vehicle speed. The results were reported as follows:

- For antenna selection, a standard 3 dB antenna had the highest positive matches.
- For cell phone placement, there was a significantly lower rate of positive reads when the cell phone was lower in the vehicle as opposed to resting on the dashboard.
- Lateral setback was measured at intervals between 6 and 70 ft. with no significant difference in reliability between them for the aforementioned 3 dB antenna.

- Vertical elevation of the antenna, mounting the antenna at least 3 ft. off the ground, provided significant increase in the reliability.
- Vehicle speed affected the reliability when the vehicle speeds were at 60 mph. This was due to the device not spending enough time in the sensing area of the antenna.

It has also been suggested that Bluetooth sensing not be used in a dense road network in urban areas because of the high occurrence of false reads from devices that are traveling on nearby roads. Also, because the sensors can detect devices in the opposing travel direction, it is not possible to tell the direction of unmatched detections (Zhong, 2012).

### **2.3 CELL PHONE TRACKING**

State departments of transportation have partnered, or have attempted to partner, with cell phone service providers to use cell phones as probes for use in real-time traffic reporting (Associated Press, 2005; Lieb, 2007). One type of system is designed to use triangulation to position the probe on a highway and measure the travel distance over a known time frame to determine its rate of travel. Although this method of estimating travel times worked under normal traffic conditions, congested traffic conditions did not show consistent results (Lieb, 2007; Wunnava, Yen, & Babij, n.d.).

Another type of system that utilizes cell phones as traffic probes uses double handover (DHO) events to estimate travel speeds (Wunnava, Yen, & Babij, n.d.; Wilson, 2004). This system estimates a vehicle's position as it moves from the range of one cell phone tower to another and measures the time until another DHO event occurs. This process gives two points along a known roadway, thus providing distance and a measured time so the speed can be calculated. Shortcomings for this type of system have been identified as the following (Krygsman & de Jong, 2008):

- Due to atmospheric conditions, signal strengths can vary, causing a cell phone to be handed over amongst a small group of nearby base stations and making the vehicle appear to be moving in circles.
- Base stations are often taken down for maintenance during off-peak times. This can result in the interpretation of a vehicle jumping from one location to another.
- In relatively sparsely populated areas, a cell can be quite large. This can cause a sample size to be reduced because a larger percentage of the vehicles that enter the cell may be a local trip, not leaving the cell giving the second location.
- Another difficulty with using cell phones as probes is the privacy concerns. In areas that have deployed them, the data have been used for transportation management. In states that have attempted to use the data for public presentation, privacy protection groups have contested its use.

### **2.4 TOLL TAG READERS**

Toll tags have been identified as a technology to determine travel times along a roadway segment. By matching a toll tag over a known distance, the travel time can be determined for the vehicle carrying the device. The iFlorida Model Deployment made extensive use of toll tag readers on the transportation network, including non-tolling facilities (Haas & al., 2009) . The

inclusion of using the toll tag readers as a data source was based on the high number of vehicles in the Central Florida area that have toll tags. At the time, it was estimated that 20 percent of all vehicles had the devices. It is important to have a high penetration rate to overcome the deficiencies of toll tags. These deficiencies were noted in the iFlorida Deployment as follows:

- Not all vehicles have toll tags, so at best you can have a representative sample.
- Not all toll tags are read at every sensor.
- Toll tags can be misread.
- In a non-free-flow environment, vehicles can sit within the detection zone long enough to be read duplicate times; this could be exacerbated by duplicate reads at both ends of the segment.
- Toll tags moving in the opposite direction can be read.
- Vehicles can divert the network between sensors.
- Vehicles can stop along the segment.
- For privacy, the actual identifiers of the toll tags are not stored but are converted; however, it is possible for more than one toll tag ID to have the same hash, causing hash key mismatch.

Another factor identified in the iFlorida Deployment that can affect the success of using toll tags is the matching efficiency, which is the measure of how often the same toll tag is identified and matched along the sampling area. Factors that affect this measure are vehicles leaving the segment between sensors and roadway coverage. Subfactors for vehicles leaving the segment between sensors include the length of the segment, number of major intersections in the segment, driving patterns on a particular segment, and access control to the segment. As for lane coverage, if not all lanes are monitored by the sensors at both or one end of a segment, then there will be fewer reads and fewer matches.

Toll tags can be used for determining O–D matrices, but they share the same primary issue as when used for determining travel times—the availability of toll tag readers within the network (Kwon & Varaiya, 2005). In performing their research, Kwon and Variya developed a statistical model to overcome the shortfall of data. They also noted that they limited themselves to uniquely traversable paths and that the inclusion of non-uniquely traversable paths will require adjustments to their model.

# Chapter 3 METHODOLOGY

## 3.1 LOCATION SELECTION

Researchers in this project selected three stations on a corridor on Interstate 75. The locations were selected in consultation with FDOT for this project:

- Location 1 is the NW Highway 320 intersection with I-75.
- Location 2 is the gantry just before the SR 44 off-ramp on NB I-75.
- Location 3 is at the Warm Springs Avenue intersection.

These three locations are shown in Figure 3-1, Figure 3-2, and Figure 3-3, respectively.



Figure 3-1: Overview of Location 1

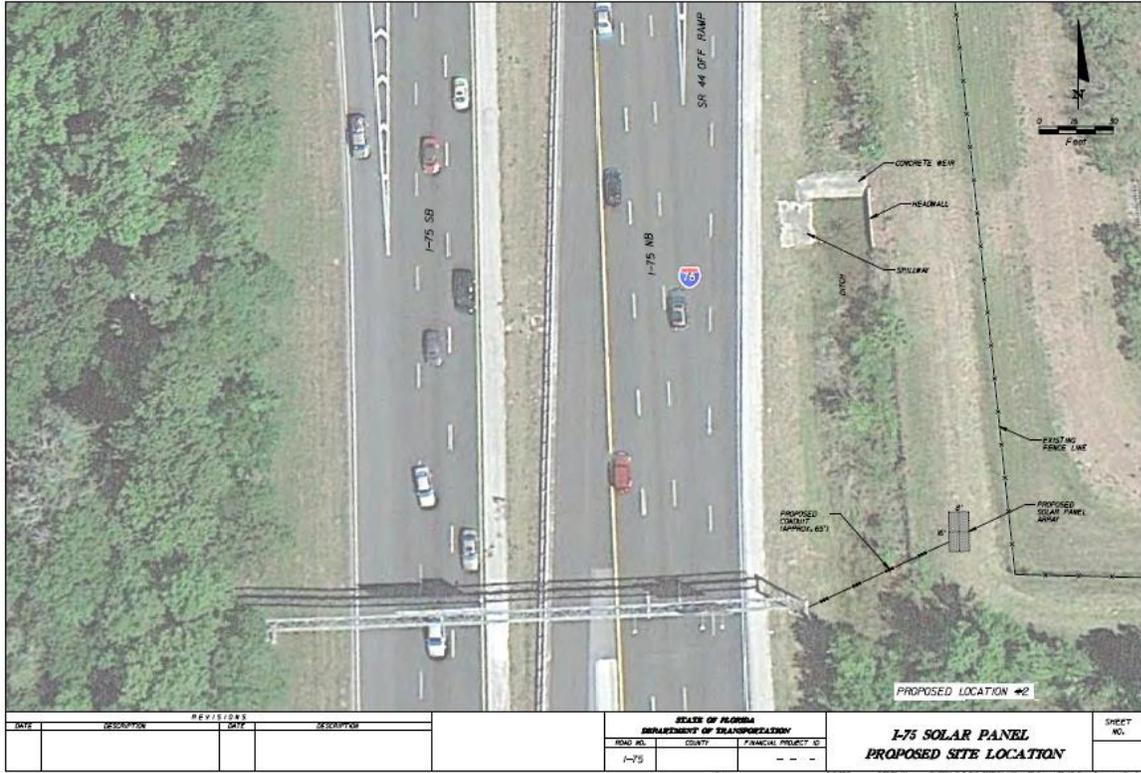


Figure 3-2: Overview of Location 2



Figure 3-3: Overview of Location 3

## **3.2 HARDWARE SELECTION**

Extensive time went into selecting and testing various cameras and lenses for this project. Testing involved camera selection, lab testing, initial field testing, and software development, followed by final field testing.

Cameras need to have good overall sensitivity, adaptation to changing light conditions, ease of installation, and a frame rate exceeding 30 frames per second. Cameras from two vendors were tested: (1) Point Grey, and (2) IDS Imaging.

Two custom-built brackets were designed, built, and installed by the research team. Both designs worked very well. Design considerations included the following:

- The design must be rugged to withstand wind, and should enable height customization.
- The two brackets must require parts readily available from Pelco Products and Lowe's Home Improvement.
- For reliable operations, the system must have a very accurate trigger that is rigidly connected to a vertical member.

## **3.3 SOLAR POWER SYSTEM**

The research team obtained several quotations for the installation of the solar power system from specialized contractors. Quotations averaged from \$30,000 to in excess of \$40,000. These costs did not even include hardware.

### **3.3.1 Alternative Foundation Design**

The high costs of the solar system were due primarily to the heavy foundations required and the location of the structures relative to the gantries, which are factors that could not have been estimated by the research team at the project's onset.

To reduce costs to a more manageable amount, the team worked with a contractor on an alternative foundation design. Universal Solar, a contractor specializing in solar equipment installation, was selected after evaluation of more than six companies. The new design uses Cantsink metal piles instead of reinforced concrete.

FDOT approved the new design for the alternative foundation. Metal piles are much cheaper and faster to set, do not require the use of concrete, and can even be removed and reused. They are by far the best alternative for supporting solar arrays.

### **3.3.2 Installation of Solar Power Panels at the Three Sites**

To generate the power necessary for equipment operation, solar cells and their associated structure were purchased. From competitive bids, the research team selected a contractor who installed the system at all three sites. Sample images of solar power systems in each location are provided in Figure 3-4, Figure 3-5, and Figure 3-6. In addition, a zipped file containing the process-of-installation images was uploaded to the FDOT Research Office's website.



**Figure 3-4: Solar Power System – Location 1**



**Figure 3-5: Solar Power System – Location 2**



**Figure 3-6: Solar Power System – Location 3**

### **3.4 SOFTWARE AND ANALYSIS TOOLS**

To meet the goals of the study, the researchers in this project developed three levels of software to receive and analyze the license plate reads. These programs allowed for the receipt of the laser sensor signals, control of the cameras' image acquisition, and then for the optical character recognition operation and analysis of those results.

#### **3.4.1 Software Development**

##### *Laser Sensor Signals and Image Acquisition*

The first UCF-OCR software package the research team developed was used to receive the laser sensor signals and control the cameras' image acquisition. The primary features of the software enabled the research team to perform the following:

- Identify the station
- Enable/disable each camera individually
- Identify the cameras

- Designate each camera by IP address on the network
- Send parameters/options to the cameras via URL
- Designate the storage location of the images
- Set how many frames to capture (required to account for speed and length differences)
- Select which COM port the sensor is using
- Set the delay between the sensor trigger and the command to be sent to the cameras to begin image acquisition

In that program, the image files were generated and stored locally during data collection intervals. The filename included the station name and lane information.

### *OCR Operation*

A second custom software package then performed the OCR operation on the license plates. The result of the OCR was stored in a database along with the station the image was taken, the lane the camera was mounted over, and the date and time the picture was taken.

### *OCR Report Analysis*

The third software package designed in this study then analyzed the results of the OCR. With these data points stored in a database, query and string-matching techniques were used to create O-D data. The license plates were matched using a three-step approach beginning with a database query that used exact matches within a selected time frame. The exact matches were stored in a table so that those license plate reads would be removed from further analyses.

After that matching process, a database query using fuzzy logic and wildcards was used to take into account common OCR misreads. Comparisons were only made between read locations that occurred within a selected time frame. Matching results using this fuzzy logic pattern were stored in a separate table.

The remaining unmatched licenses plates were then analyzed using two algorithms. The first algorithm compared string sequences that had a minimum of four characters coincide in the same order. The second analysis used a modified Levenshtein algorithm. The Levenshtein algorithm compares the similarity of two strings based on the number of edits that need to be made on one string to make it the same as the second. Researchers modified this algorithm to take into account common OCR errors. In this application, there was no edit required to change a 2 to a Z or the letter O to the number 0.

Any remaining license plates were assumed to have entered or left the network.

## **3.4.2 Hardware and Software Modification**

Improvements in camera technology have made it potentially possible to use a single camera for every two lanes. This drastically reduces *both* power consumption and the costs of installation and maintenance. The new camera was purchased and the software updated. Sample code is provided in Appendix A.

Furthermore, the research team selected a new trigger. This single trigger can operate on two lanes simultaneously, thereby reducing cost and simplifying installation. The research team wrote software for the new trigger. Sample code for the trigger is included in Appendix A.

### 3.4.3 Management of Origin and Destination Data and Analyses

The management of origin and destination data and analyses (MODDA) software described in this documentation was baselined and designed to support the efforts of the technology deployment for an O–D study project. Rather than just performing post-processing of plate reads by matching one location to another, this software was designed for growth and scalability. As the sensing network expands and includes other roadways and sensing locations, these new elements can be easily added. Although there is an interface for adding or disabling plate-matching algorithms, bringing in new algorithms will require additional programming to support them.

The software was written by the research team on the .NET Framework 4.0 and supports any Windows® operating system from Windows XP and above, as long as the .NET Framework 4.0 is installed. The underlying database was designed and tested using Microsoft SQL Express. The SQL scripts that were created during development can be used on any version of SQL Server above SQL Server 2005.

MODDA is intended to be used by the research team while data collection efforts are ongoing. After the completion of the project, it can continue to be used to manage and analyze plate-read data.

#### *Main Interface*

The main interface comprises two components: (1) the tree view menu on the left and (2) the multiple-document interface (MDI) panel on the right. The MDI panel hosts the forms that appear when the user clicks on the menu items on the left. In this documentation, the following nomenclature will be used:

- *Interface*: The main container of the software.
- *Form*: A window that appears when a menu item is clicked. A form is actually a container for controls that are selected using the toolbar within the form.
- *Control*: The content area that becomes exposed when the toolbar within the form is clicked.

#### *Locations*

The Locations form is used to manage the sensing locations. It has three controls: Locations, Roadways, and Links.

**Locations Control.** The Locations control manages the sensing locations by allowing the user to add, modify, or remove sensing locations. When the control first opens, it is in Insert Mode, meaning the user can enter new values for the data elements Roadway, Direction, MilePoint, and Description, and the submit button has the text “Add”. When one of the rows in the grid is

selected, the data elements are populated from the selected row and the control goes into Edit Mode, meaning the user can change the data elements, and the submit button has the text “Update.” Clicking the reset button clears the data elements and sets the control to Insert Mode. The delete button is only available for when a roadway is first entered and there is an error in data entry. Once any other data point in the database becomes associated with a location, it cannot be deleted. Sensing locations are independent of direction as they might not be at the same lateral location.

**Roadways Control.** The Roadways control is used to manage roadways that have sensing locations on them. The database is seeded with current values, but it can be updated as new locations are added to the network. Just like the Locations control, it has Insert and Edit Modes with the same restriction on deleting.

**Links Control.** The Links control allows for the association of an upstream sensing location with another location downstream. The downstream location does not need to be on the same roadway. If an intersecting roadway has a sensing location downstream from another location, they can be linked. These links are selected during the matching process.

To link two locations, the user follows the steps below:

- 1) Select a roadway and a direction for the upstream location. The location ComboBox will populate based on matching values in the database.
- 2) Click on the Search button to list all the downstream locations that have already been linked to the selected upstream location. Clicking on a row in the GridView will populate the detail data below and put the control into Edit Mode.
  - a) To add a new linked location, click reset if needed to re-enter Insert Mode.
  - b) Use the roadway and direction ComboBoxes to populate the downstream location ComboBox. If the desired downstream location does not appear, it must be added using the Locations control.
- 3) Once the downstream location is selected, enter the distance between the two locations and enter the date the location started collecting data in the Date Active element. It is imperative that an accurate value is entered for the distance as this is used in the matching algorithms to filter plate reads over time.

### *Raw Data*

The Raw Data form is set up to manage the importing, managing, and exporting of the raw data.

**View Raw Data.** The View Raw Data control allows for the user to search for and view raw data for a specific day, roadway, direction, and location. It is intended to be used to verify the data being imported and the outcome of the matching algorithms.

**Import Raw Data.** The Import Raw Data control allows the user to upload a comma-separated values (CSV) text file containing the desired data: lane, plate (as output by the OCR software), and time of detection. Although the imported text file may have more columns than these and in a different order, the control allows the user to map the data to the database columns. The data are imported by selecting the source file, selecting the location after filtering on roadway name

and direction, selecting the checkbox if the header row is included in the source file, and then clicking the Load button. This loads the source file into memory and enables the data mapping elements on the control. The elements match the column headers in the grid to the columns in the database. Date is not included in the form as each row in the source file includes the date.

**Export Raw Data.** Currently, the Export Raw Data control allows the user to filter down to a sensing location and export the data into a CSV text file. This will be modified so the data can be filtered by these elements but not restricted by them. This means that in a single export, more than one location or direction can be exported.

### *Matched Plates*

The Matched Plates form contains the controls that are used to perform and manage plate matching.

**Match Reads.** The Match Reads control allows the user to perform plate matches for specified links on certain dates. The user can also select which matching algorithms to use during this process. As the matching algorithms are processed, there is a space for the results to be displayed below.

**View Matches.** The View Matches control allows the user to view the results of the matching algorithms by *filtering* on roadway, direction, location, date, and matching algorithm. A button to export these results will be added.

**Match Types.** The Match Types control allows users to add or modify the descriptive data of the matching algorithms. Once entered, they cannot be deleted via the interface. Disabling an algorithm will suppress its inclusion during the matching process. Changing the runOrder number changes the order that they are implemented during the matching process. The runOrder is also used to sort other data elements within the interface.

The matching algorithms that are currently supported are:

- *Direct SQL Match:* These matches have the highest confidence because the “equals” operator is used.
- *SQL Fuzzy Match:* This matching algorithm provides a high level of confidence because it takes into account typical OCR errors, such as a Z for a 2 or a C for a G.
- *Modified Levenshtein:* The Levenshtein Proximity algorithm states how similar two character strings are based on the number of changes one string has to undergo to become the same as the other one. This algorithm has been modified to take into account the typical OCR errors.
- *Visual Inspection:* This requires looking at the source photos to either confirm or create a match. This has not been implemented in the software yet and cannot be included in the automated process.

### 3.4.4 Database Design and Programming

A database to store data points was designed and implemented in Microsoft SQL Server in this study. A simulated data set for three locations was created.

#### *OD\_PlateRead*

The database schema has a star pattern centering around the table that stores the results of the OCR process. The table that stores these data is OD\_PlateRead (see Table 3-1). The data are imported within the application, which assumes the input file is exclusive to a specific location and contains the following data points:

- *lane* – the lane closest to the median being 1 and increasing by 1 for each lane
- *plateNo* – the plate number as read by the OCR software
- *time* – the date and time that the truck was detected
- *id* – an Identity column
- *id\_Location* – a Foreign Key for a sensing location that encapsulates the direction of traffic flow
- *plateEncrypted* and *salt* – columns having to do with encrypting the value in *plateNo*
- *id\_PlateReadImport* – a Foreign Key for the file from which the data were imported

**Table 3-1: OD\_PlateRead**

COLUMN NAME	DATA TYPE	LENGTH
id (PK)	bigint	
id_Location (FK)	int	
lane	tinyint	
plateNo	varchar	20
time	datetime	
plateEncrypted	bit	
salt	varchar	30
id_PlateReadImport (FK)	bigint	

#### *OD\_Location*

The table OD\_Location stores information about the sensing locations (see Table 3-2). It includes the following:

- *id* – a Primary Key identity column
- *id\_Roadway* – a Foreign Key to the OD\_Roadway table
- *description* – a real world description of the location, such as “Overpass at SR 319”
- *milePoint* – as described in FDOT Intersections data and a Foreign Key to the OD\_Direction table

**Table 3-2: OD\_Location**

COLUMN NAME	DATA TYPE	LENGTH
id (PK)	int	
id_Roadway (FK)	int	
description	varchar	60
milePoint	decimal	
id_Direction (FK)	int	

*OD\_PlateReadImport*

The table OD\_PlateReadImport stores data regarding the data import process (see Table 3-3). It stores the name of the file that was used to import the data (*filename*) and the date that the file was imported (*date*). This table serves two purposes: (1) it helps to ensure that the same file is not imported more than once, creating duplicate data; and (2) it provides a means to undo an import if for some reason the source data are found to be inaccurate.

**Table 3-3: OD\_PlateReadImport**

COLUMN NAME	DATA TYPE	LENGTH
id (PK)	bigint	
filename	varchar	255
date	datetime	

*OD\_Roadway*

The table OD\_Roadway simply holds the name of a roadway (*roadName*) that contains sensing locations (see Table 3-4). Its inclusion in the database is based solely on scalability.

**Table 3-4: OD\_Roadway**

COLUMN NAME	DATA TYPE	LENGTH
id (PK)	int	
roadName	varchar	20

*OD\_Direction*

The table OD\_Direction (see Table 3-5) stores the four directions of travel roadways given: north, south, east, and west. There is no data validation based on actual directions for an included roadway, nor is there any validation based on the odd/even standard. The direction is used to further classify a sensing location based on the direction of travel that occurs at that station.

**Table 3-5: OD\_Direction**

COLUMN NAME	DATA TYPE	LENGTH
id (PK)	int	
travelDirection	varchar	5

### *OD\_RoadwayLinks*

The table OD\_RoadwayLinks defines the relationship between an upstream sensing location and a downstream sensing location (see Table 3-6). This relationship is defined by two Foreign Keys to the OD\_Location table and is characterized by the other data fields: *distance*, *dateActive*, and *dateInactive*. The *distance* data point is used to help determine boundary conditions for matching plate reads, while the *date* fields are used to help determine when to apply this link. The relationship between sensing locations is irrespective of the roadway, meaning that a downstream location can be on a different roadway.

**Table 3-6: OD\_RoadwayLinks**

<b>COLUMN NAME</b>	<b>DATA TYPE</b>	<b>LENGTH</b>
id (PK)	int	
id_LocationUpstream (FK)	int	
id_LocationDownStream (FK)	int	
distance	decimal	
dateActive	datetime	
dateInactive	datetime	

### *OD\_Match*

The OD\_Match table (see Table 3-7) stores the matches between plate reads in the OD\_PlateRead table. The *id* is a Primary Key identity column, while the Foreign Keys *Id\_PlateRead\_1* and *id\_PlateRead\_2* set the relationship between the two reads. The location of each of the plate reads in the relationship must be a defined link in OD\_RoadwayLinks. The Foreign Key *id\_MatchType* describes the algorithm used to match the plate reads.

**Table 3-7: OD\_Match**

<b>COLUMN NAME</b>	<b>DATA TYPE</b>	<b>LENGTH</b>
id (PK)	bigint	
id_PlateRead_1 (FK)	bigint	
id_PlateRead_2 (FK)	bigint	
id_MatchType (FK)	int	

### *OD\_MatchType*

The OD\_MatchType table stores the information about the matching algorithms used to match the plate reads (see Table 3-8). The *id* column is a Primary Key identity column. The data field *displayName* is an abbreviated description of an algorithm and is intended to be displayed within the software. The *description* column stores a lengthier version of the algorithm. The *isActive* column indicates whether an algorithm is to be used. The *runOrder* column indicates in which order the algorithms are applied when matching. It is also used as a sort order in the software.

**Table 3-8: OD\_MatchType**

<b>COLUMN NAME</b>	<b>DATA TYPE</b>	<b>LENGTH</b>
id (PK)	int	
displayName	varchar	30
description	varchar	250
isActive	bit	
runOrder	int	

### 3.4.5 Generating a Simulated Dataset to Test the Software

For this project, researchers created a simulated data set for three locations. The data were designed in a way to create exact and fuzzy matches at the three locations to enable the research team to evaluate the routing algorithm. Generation steps were followed as described for each location.

#### *Generation Steps of Location 1 Data*

- a) Random generation of 7-character license plate reads were made (8578 rows).
- b) Characters were defined as: 0 to 9 and A to Z excluding the letter “O.”
- c) “Date & Time” starting from “9/1/2013 00:16:00” and keeping the plate-read rate consistent at six plates per minute was assigned in another column.
- d) Random generation of lane numbers (1 to 3) for each plate read was used.
- e) Plate reads were performed for a single 24-hour period.

#### *Generation Steps of Location 2 and Location 3 Data*

- a) Location 1 data were copied as Location 2 and Location 3 data.
- b) Time of plate read at Location 2 and Location 3 was changed to be approximately the time it would take to drive 65 mph from Location 1 to Location 2 and Location 2 to Location 3. Thus, “Date & Time” of Location 2 became 31 minutes later than that assigned to Location 1, and “Date & Time” of Location 3 became 4 minutes later than that of Location 2.
- c) Random generation of lane numbers (1 to 3) for each plate read was used for each location.
- d) Assignment of target plate match type was performed as follows:
  - At the outset, the plate reads from one location to the next contained all exact matches, so they needed to be altered manually.
  - In another column, a number from 1 to 50 was randomly generated. So, the whole data set was divided into 50 groups with each group assigned to a certain number.
  - To have differences among the three locations’ plate reads, some of these groups were assigned to a matching type for each location pairing. Each group was also assigned to a certain color. The assignment groups are shown in Table 3-9, with definitions following the table.

**Table 3-9: Assignment Groups of Matching Types**

<b>Group Number</b>	<b>Matching Type</b>	<b>Location Pairing</b>	<b>Number of Plates</b>	<b>Color</b>
5	Fuzzy	Loc. 1 to 2 & Loc. 1 to 3	172	BLACK
6	Fuzzy	Loc. 1 to 2 & Loc. 2 to 3	160	RED
7	Fuzzy	Loc. 1 to 3 & Loc. 2 to 3	163	ORANGE
8	Partial – Drop off a non-fuzzy character	Loc. 1 to 2 & Loc. 1 to 3	183	YELLOW
9	Partial – Drop off a non-fuzzy character	Loc. 1 to 2 & Loc. 2 to 3	204	L. GREEN
10	Partial – Drop off a non-fuzzy character	Loc. 1 to 3 & Loc. 2 to 3	191	L. BLUE
11	Partial – Change a non-fuzzy character	Loc. 1 to 2 & Loc. 1 to 3	155	GREEN
12	Partial – Change a non-fuzzy character	Loc. 1 to 2 & Loc. 2 to 3	196	PURPLE
13	Partial – Change a non-fuzzy character	Loc. 1 to 3 & Loc. 2 to 3	163	BLUE
14	Partial – Drop off a non-fuzzy character and make a fuzzy replacement	Loc. 1 to 2 & Loc. 1 to 3	171	D. BLUE
15	Partial – Drop off a non-fuzzy character and make a fuzzy replacement	Loc. 1 to 2 & Loc. 2 to 3	165	BG YELLOW
16	Partial – Drop off a non-fuzzy character and make a fuzzy replacement	Loc. 1 to 3 & Loc. 2 to 3	167	BG RED
17	Partial – Change a non-fuzzy character and make a fuzzy replacement	Loc. 1 to 2 & Loc. 1 to 3	174	BG GREEN
18	Partial – Change a non-fuzzy character and make a fuzzy replacement	Loc. 1 to 2 & Loc. 2 to 3	172	BG BLUE
19	Partial – Change a non-fuzzy character and make a fuzzy replacement	Loc. 1 to 3 & Loc. 2 to 3	171	BG PURPLE
<b>Location Number</b>				
20	Deleted	Loc.-1	169	
21	Deleted	Loc.-2	169	
22	Deleted	Loc.-3	186	

*Definitions for Table 3-9:*

- Location Pairing
  - By changing the plate numbers in Location 1, the matching between Location 1 to Location 2 and Location 1 to Location 3 would be affected, while Location 2 to Location 3 had exact matches at this point.
  - In the same way, by changing the plate numbers in Location 2, the matching between Location 1 to Location 2 and Location 2 to Location 3 would be affected, while Location 1 to Location 3 had exact matches at this point.
  - Lastly, by changing the plate numbers in Location 3, the matching between Location 1 to Location 3 and Location 2 to Location 3 would be affected, while Location 1 to Location 2 had exact matches at this point.

- Fuzzy Matches
  - In this matching type, certain characters or numbers were considered as the same. So, in these groups the following replacements were made:
    - “B” was replaced with “8”
    - “D” was replaced with “O”
    - “Q” was replaced with “0”
    - “Z” was replaced with “2”
    - “S” was replaced with “5”
    - “I” was replaced with “L”
    - “1” was replaced with “7”
  - To create fuzzy matches, Excel string functions were used to determine if there were already fuzzy-eligible characters in the upstream Location. If there were, an eligible replacement was used in the downstream Location. If there were not, an eligible character was put in the upstream Location and a valid replacement was put in the downstream Location.
- Partial Matches
  - Four types of partial matching were created:
    - Drop off one non-fuzzy character: One of the characters or numbers that was not a fuzzy-eligible character was found and deleted.
    - Change one non-fuzzy character: One of the characters or numbers that was not a fuzzy-eligible character was found and replaced with another non-fuzzy-eligible character.
    - Drop off one non-fuzzy character and make a fuzzy replacement: One of the characters or numbers that was not a fuzzy-eligible character was found and deleted. In addition, a fuzzy-eligible character was replaced with another fuzzy-eligible character.
    - Change one non-fuzzy character and make a fuzzy replacement: One of the characters or numbers that was not a fuzzy-eligible character was found and replaced with another non-fuzzy-eligible character. One of the characters or numbers that was not a fuzzy-eligible character was found and replaced with another non-fuzzy-eligible character.
- Deleted Cases
  - To create non-matches between location pairs, plate numbers were deleted in three different groups for the corresponding locations.

## Chapter 4 DATA COLLECTION

### 4.1 SUMMARY

A comparison of traffic counts collected by the University of Central Florida (UCF) system with historical data collected by the Florida Department of Transportation (FDOT) is presented in this report. A summary of findings for traffic counts for days spanning a two-week period from Monday 10/24/2016 until Saturday 11/05/2016 from both the UCF system and the FDOT system are presented in Table 4-1. Daily average truck percentages are recorded in the UCF system; however, the FDOT system truck percentage records are provided in a yearly average as “T Factor = 19.8” for the year 2015. Furthermore, two days of data collection results—on Thanksgiving and the day before—were added to the summary table.

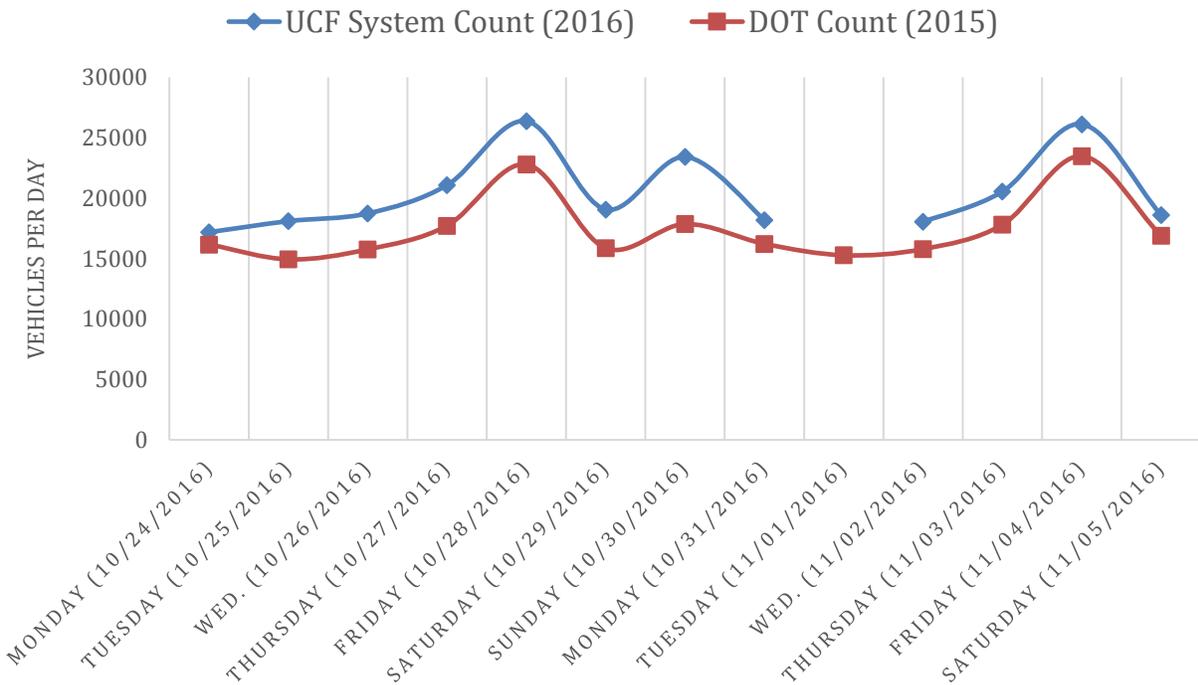
**Table 4-1: Summary of Traffic Counts**

Date	Cars	Trucks	UCF System Count (2016)	Truck (%)	DOT Count (2015)	DOT Truck (%)	
Monday (10/24/2016)	11,742	5440	17,182	32	16,168	19.8	
Tuesday (10/25/2016)	12,584	5512	18,096	30	14,938	19.8	
Wednesday (10/26/2016)	13,095	5638	18,733	30	15,761	19.8	
Thursday (10/27/2016)	15,555	5521	21,076	26	17,718	19.8	
Friday (10/28/2016)	21,278	5092	26,370	19	22,790	19.8	
Saturday (10/29/2016)	16,564	2499	19,063	13	15,864	19.8	
Sunday (10/30/2016)	18,237	5179	23,416	22	17,856	19.8	
Monday (10/31/2016)	13,129	5065	18,194	28	16,219	19.8	
Wednesday (11/02/2016)	12,442	5618	18,060	31	15,786	19.8	
Thursday (11/03/2016)	15,051	5490	20,541	27	17,803	19.8	
Friday (11/04/2016)	20,908	5211	26,119	20	23,479	19.8	
Saturday (11/05/2016)	16,015	2596	18,611	14	16,889	19.8	
Thanksgiving	Wednesday (11/23/2016)	28,280	3717	31,997	12	N/A	N/A
	Thursday (11/24/2016)	19,760	1246	21,006	6	N/A	N/A

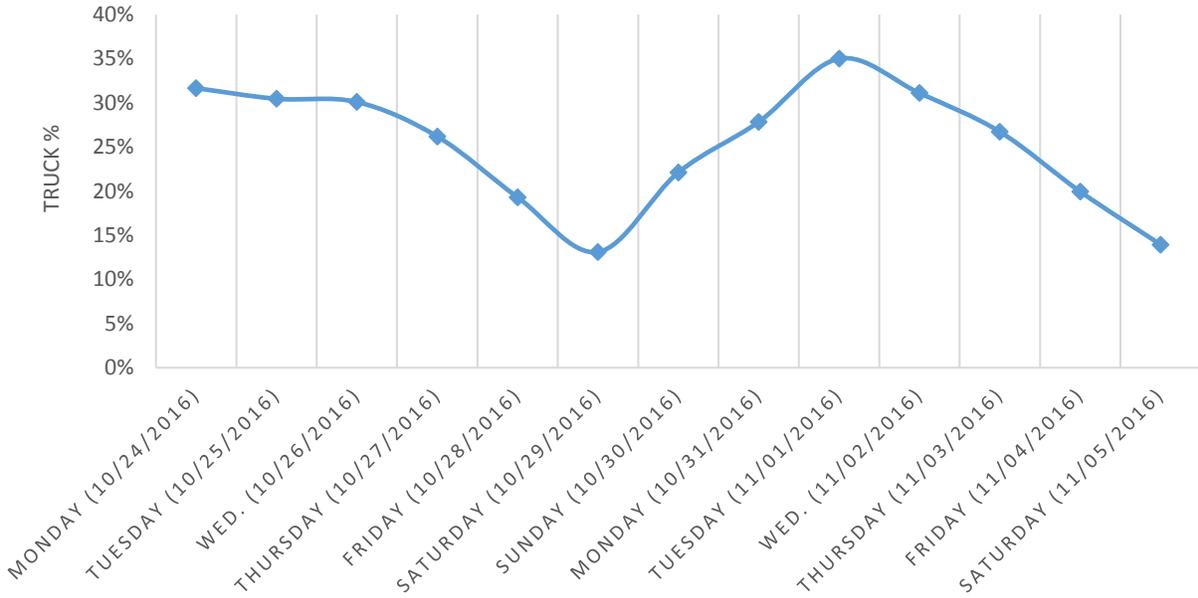
In addition, two days (10/25/2016 and 11/04/2016) of plate recognition tests were performed on Location 1 and Location 3. A summary table of total captured photos, plate readings, and read percentages for both nighttime and daytime are provided in Table 4-2. Daily traffic captured by the UCF system and the FDOT system trend lines are presented in Figure 4-1. Daily truck percentages and moving averages are presented in Figures 4-2, 4-3, and 4-4.

**Table 4-2: Plate Recognition Initial Findings**

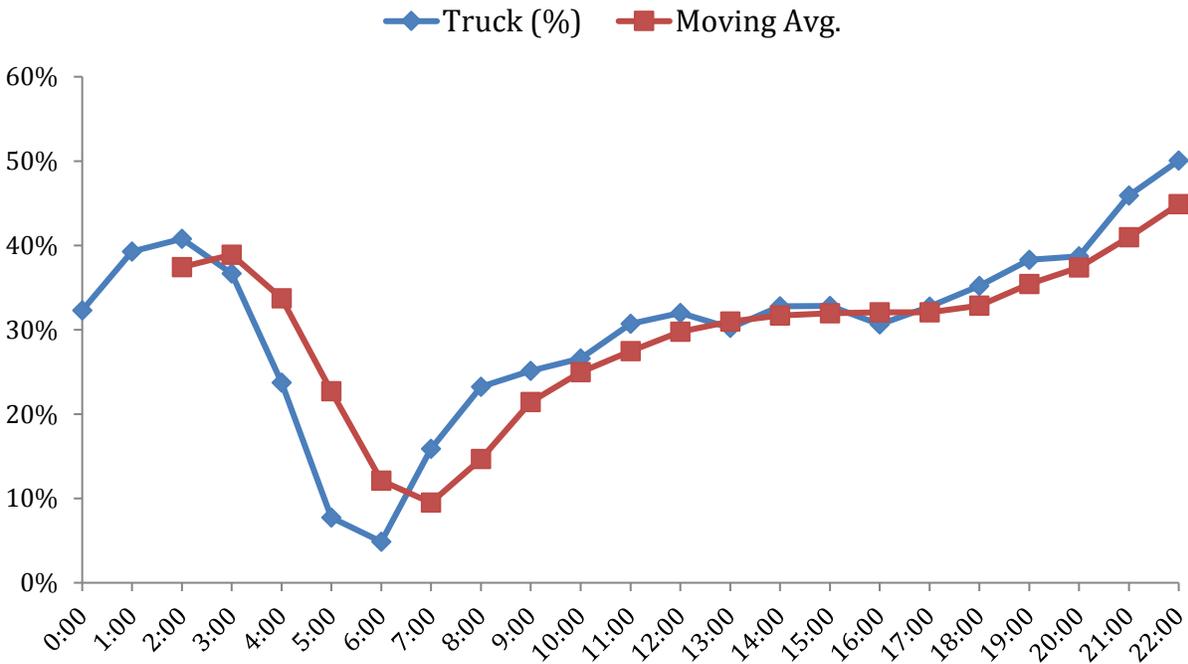
		Location 1 (10/25/16)			Location 3 (10/25/16)			Location 1 (11/4/16)		
		Captured	Read	Read (%)	Captured	Read	Read (%)	Captured	Read	Read (%)
Daytime	Left Ln	3274	1701	52	2850	1524	53.5	2686	1820	67.8
	Right Ln	514	153	29.8	533	337	63.2	556	331	59.5
Nighttime	Left Ln	2370	1396	58.9	1701	227	13.3	1446	616	42.6
	Right Ln	403	65	16.1	280	53	18.9	180	81	45
Total	Left Ln	5644	3097	54.9	4551	1751	38.5	4132	2436	58.9
	Right Ln	917	218	23.8	813	390	48	736	412	56



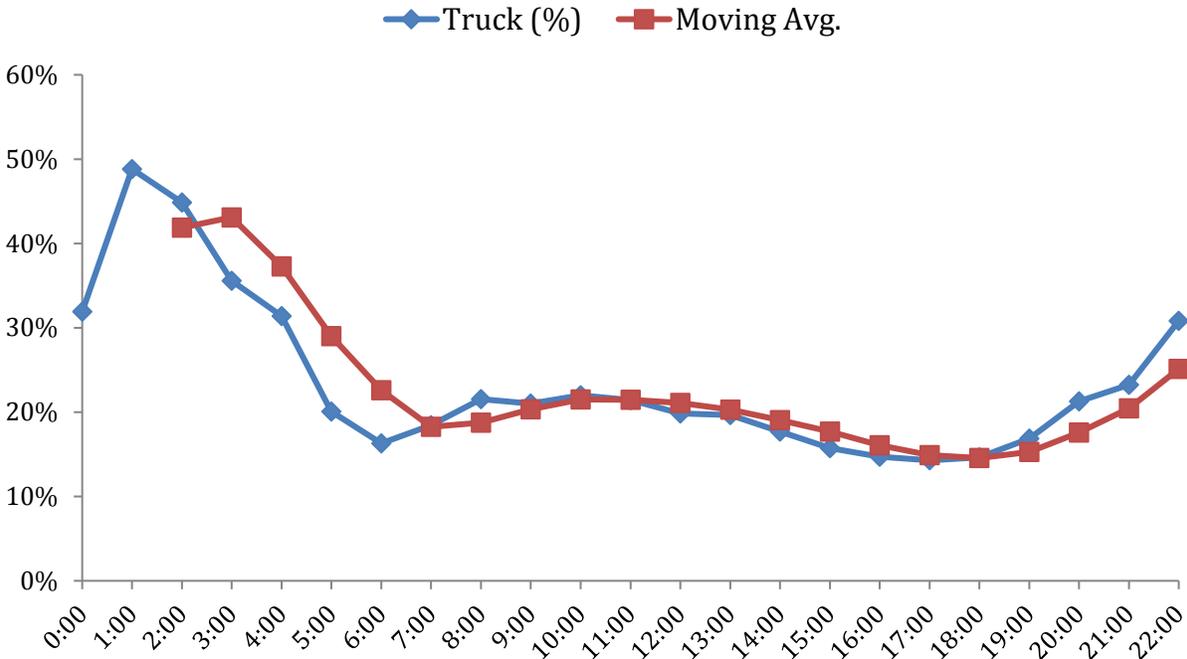
**Figure 4-1: Daily Traffic Trends for Days Spanning Two Weeks**



**Figure 4-2: Daily Truck Percentages**



**Figure 4-3: Monday (10/31) – Truck Percentage Moving Average**



**Figure 4-4: Friday (11/4) – Truck Percentage Moving Average**

## 4.2 BACKGROUND

The researchers evaluated traffic count data obtained by the UCF data collection system during two weeks from October into November 2015. The system consists of three northbound locations on Interstate 75. Location 3 is a data collection point in Sumter County, Florida, on I-75 at the Warm Springs Avenue intersection. This intersection is 3.5 miles south of Florida’s Turnpike ramp. The site location images are presented in Figure 4-5 to 4-8. Additional photos can be found in Appendix B. In the UCF data collection system, the camera is mounted on the overpass at Location 3 for two travel lanes to capture images of the front license plates. The camera is triggered by a laser sensor. The radar sensor used is an inexpensive sensor that can accurately measure the distance from the ground. To avoid capturing light trucks and passenger vehicle photos, the sensor would constantly measure the height of each passing vehicle and trigger the system to save the last multiple photos of vehicles that are at a specified height. The data collection system captures the number of trucks as well as other vehicles based on their heights. For simplicity, other vehicle types are named “Car” throughout the report.



**Figure 4-5: Location 3 on I-75 at Warm Springs Ave.**



**Figure 4-6: Capture System Mounted on Overpass**



**Figure 4-7: Sensor Aimed at Traffic**



**Figure 4-8: Bracket Close-up Showing Sensor, Camera, and Processor**



Table 4-3 presents the historical AADT records from year 2000 to 2015 for this point. In year 2015, AADT was calculated as 44,301 for the combined directions at this site. Northbound AADT is 22,463. When T factor is applied, truck counts will be approximately 4448. In the UCF system data collection, average daily traffic for 2 weeks is 20,455 and average daily trucks is 4905. In Table 4-4, continuous counts in September 2015 of the northbound lanes of this location are listed by hour of the day. The last complete month of hourly volume counts for 2015 made by FDOT ends in September; the remainder of the year has more than a few missing days. According to Table 4-4, average weekday traffic is computed as 19,609 in September at this location. Consequently, both daily traffic and truck percentages from the UCF data collection system are found to be very close to 2015 averages provided by FDOT.

**Table 4-3: Historical AADT Report (FDOT)**

FLORIDA DEPARTMENT OF TRANSPORTATION  
 TRANSPORTATION STATISTICS OFFICE  
 2015 HISTORICAL AADT REPORT

COUNTY: 18 - SUMTER

SITE: 9920 - SR-93/I-75, 3.5 MI S OF FL TPK, SUMTER CO.

YEAR	AADT	DIRECTION 1	DIRECTION 2	*K FACTOR	D FACTOR	T FACTOR
2015	44301 C	N 22463	S 21838	10.50	54.70	19.80
2014	42323 C	N 21205	S 21118	10.50	55.10	19.40
2013	40900 C	N 20537	S 20363	10.50	56.00	19.80
2012	39544 C	N 19899	S 19645	10.50	56.80	20.30
2011	41424 C	N 20983	S 20441	10.50	56.80	19.90
2010	41116 C	N 20725	S 20391	11.28	56.76	22.80
2009	41311 C	N 20793	S 20518	11.34	55.03	19.70
2008	40398 C	N 20435	S 19963	11.33	56.56	24.40
2007	43616 C	N 22036	S 21580	10.71	54.89	23.30
2006	44532 C	N 22488	S 22044	10.67	55.73	21.50
2005	44205 C	N 22292	S 21913	10.90	54.80	22.80
2004	44109 C	N 22274	S 21835	10.70	57.40	21.70
2003	41722 C	N 20472	S 21250	10.10	59.80	21.70
2002	39000 F	N	S	11.00	55.10	18.00
2001	37456 C	N 19462	S 17994	10.10	59.80	26.30
2000	36200 C	N 19223	S 16977	11.20	56.30	23.40

Source: <http://flt.dot.state.fl.us/website/FloridaTrafficOnline/viewer.html>

**Table 4-4: Hourly Traffic Counts at NB Lanes on I-75 in September 2015 (FDOT)**

DATE 05/20/16

FLORIDA DEPARTMENT OF TRANSPORTATION  
 TRAFFIC COUNTS  
 HOURLY CONTINUOUS COUNTS FINAL REPORT  
 SEPTEMBER 2015

COUNTY NAME: SUMTER STATION: 9920 DIRECTION: N LANE: 0  
 DESCRIPTION: SR-93/I-75,3.5 MI S OF FL TPK,SUMTER CO.  
 LOCATION: COUNTY 18 SECTION 130 SUBSECTION 000 MILEPOST 17.589 ROUTES: I-75 SR-93

DY	D	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	DAILY
1	T	227	160	156	128	211	380	634	862	900	1000	1080	1082	1063	997	964	1032	999	839	686	614	495	407	340	279	15535N
2	W	202	146	128	165	231	432	719	990	967	1086	1172	1165	1105	1080	987	1036	1131	1017	733	626	534	470	392	302	16816N
3	R	220	139	136	152	254	424	816	1108	1137	1273	1333	1328	1279	1262	1325	1292	1362	1211	1048	964	776	640	525	410	20414N
4	F	246	200	184	189	313	543	889	1260	1252	1370	1614	1682	1771	1780	1955	2027	2046	2138	2252	1666	1350	1164	731	516	29138A
5	A	419	267	240	205	316	452	704	1096	1369	1601	1905	2045	1845	1790	1650	1523	1280	1118	740	635	597	455	438	346	23036S
6	S	243	161	112	105	147	213	345	536	776	1029	1283	1342	1323	1210	1086	1072	1027	1069	859	855	645	547	411	304	16700N
7	M	202	150	116	127	161	263	419	670	887	1217	1616	2000	2208	2055	2003	1789	1642	1400	1149	994	864	637	402	287	23258H
8	T	201	117	112	148	275	533	882	1134	1145	1385	1491	1488	1379	1250	1163	1150	1071	952	774	636	516	461	401	294	18958A
9	W	193	122	133	156	220	436	758	1059	955	1208	1315	1223	1064	1021	1096	1028	1077	992	759	655	542	427	374	282	17095N
10	R	199	161	134	157	237	418	727	1025	1005	1144	1101	1379	1425	1210	1188	1265	1139	1101	937	721	633	494	378	315	18493N
11	F	311	199	170	194	254	430	772	1098	1115	1225	1424	1640	1634	1684	1695	1898	1814	1759	1497	1340	948	995	549	407	25052N
12	A	288	250	148	156	235	464	792	1006	1177	1220	1549	1549	1546	1529	1089	1243	1315	1422	748	620	497	453	380	338	20014S
13	S	263	143	122	96	143	221	354	564	762	913	1247	1411	1410	1486	1217	1181	1176	1123	1034	875	917	700	439	272	18069N
14	M	253	155	124	124	239	447	783	1076	1004	1166	1462	1423	1297	1205	1153	1119	1103	1017	709	609	527	422	359	286	18062N
15	T	218	140	144	140	206	397	709	899	947	1053	1220	1184	1096	1037	1038	1067	1038	948	732	616	467	401	351	295	16343N
16	W	204	155	149	143	239	409	731	949	1016	1051	1146	1207	1157	1022	1088	1126	1087	990	679	700	548	412	338	300	16846N
17	R	231	150	140	131	229	412	720	1063	986	1153	1346	1364	1266	1242	1183	1245	1316	1141	933	791	603	462	406	336	18849N
18	F	234	175	155	163	269	464	794	982	1064	1304	1476	1542	1441	1614	1557	1647	1735	1636	1431	1227	958	723	503	421	23515N
19	A	328	208	168	170	223	386	658	930	1188	1279	1562	1546	1501	1109	1118	1036	899	897	860	686	524	505	411	316	18508N
20	S	206	120	120	100	140	231	366	617	775	1043	1378	1410	1426	1457	1402	1397	1375	1351	1252	974	835	684	410	257	19326N
21	M	166	153	102	149	225	486	736	1013	1017	1141	1395	1357	1302	1176	1142	1106	1070	897	747	620	528	431	353	279	17591N
22	T	206	164	133	145	199	406	699	923	912	1093	1227	1189	1008	1012	1029	1046	1036	948	712	648	512	432	358	283	16320N
23	W	195	145	112	148	225	411	742	984	940	1125	1168	1224	1262	1020	1118	1164	1114	993	787	696	529	477	381	271	17231N
24	R	219	192	136	172	269	468	823	1086	1074	1225	1445	1414	1363	1405	1225	1262	1297	1082	956	820	658	595	455	344	19985N
25	F	217	184	193	183	240	498	811	1222	1187	1290	1570	1657	1595	1736	1651	1811	1782	1551	1536	1394	1081	812	614	449	25264N
26	A	199	313	159	180	234	351	750	1073	1416	1921	2399	2389	1979	1508	1161	964	971	866	787	673	546	428	308	308	21883S
27	S	180	155	148	114	148	245	369	620	788	1098	1267	1553	1553	1438	1371	1501	1390	1189	1193	838	785	537	367	284	19131A
28	M	207	147	111	132	239	459	799	993	954	1134	1328	1418	1312	1208	1158	1177	1057	1004	751	581	514	393	332	271	17679N
29	T	174	146	132	132	194	346	631	872	918	1115	1151	1114	966	1098	1120	1070	1042	912	742	584	537	413	322	285	16016N
30	W	208	136	152	151	220	390	746	944	1001	1122	1188	1214	1182	1190	1127	1125	1132	1009	799	687	590	450	354	326	17443N

WEEKDAY AVERAGE = 19609 SATURDAY AVERAGE = 20860 SUNDAY AVERAGE = 18307 NUMBER OF GOOD DAYS 30 TOTAL MONTHLY COUNT = 582570  
 MONTHLY AVERAGE = 19602

COMMENTS:  
 9/5: NEW MEXICO STATE @ UF (7:30PM)  
 9/7: LABOR DAY  
 9/12: E. CAROLINA @ UF (7PM)  
 9/26: TENNESSEE @ UF (3:30PM)

"B"====> BAD DAY  
 "N"====> NORMAL DAY  
 "A"====> ATYPICAL DAY  
 "H"====> ATYPICAL DAY (HOLIDAY)  
 "S"====> ATYPICAL DAY (SPECIAL EVENT)

In the remainder of this section, the UCF system Location 3 traffic counts for Monday 10/31 (Figure 4-10 to Figure 4-14), Wednesday 11/02 (Figure 4-15 to Figure 4-19), Friday 11/04 (Figure 4-20 to Figure 4-24), and Saturday 11/05 (Figure 4-24 to Figure 4-30) are presented separately in the following formats:

- The FDOT website (<http://www3.dot.state.fl.us/trafficinformation/>) for real time traffic information provides average hourly volumes of each weekday of each month. The average hourly traffic counts are plotted for the corresponding days.
- UCF system traffic counts are presented at 5-minute time intervals for evaluation. Total counts of both left and right lanes were plotted by time intervals separately for each vehicle type.
- The cumulative car, truck, and total counts per 5-minute time interval are plotted for comparison.
- Truck counts are converted into percentile and cumulative percentile charts in 5-minute time intervals.
- The hourly truck volumes collected by the UCF system show the truck traffic trend within the corresponding day.

### 4.3.1 Monday 10/31/2016

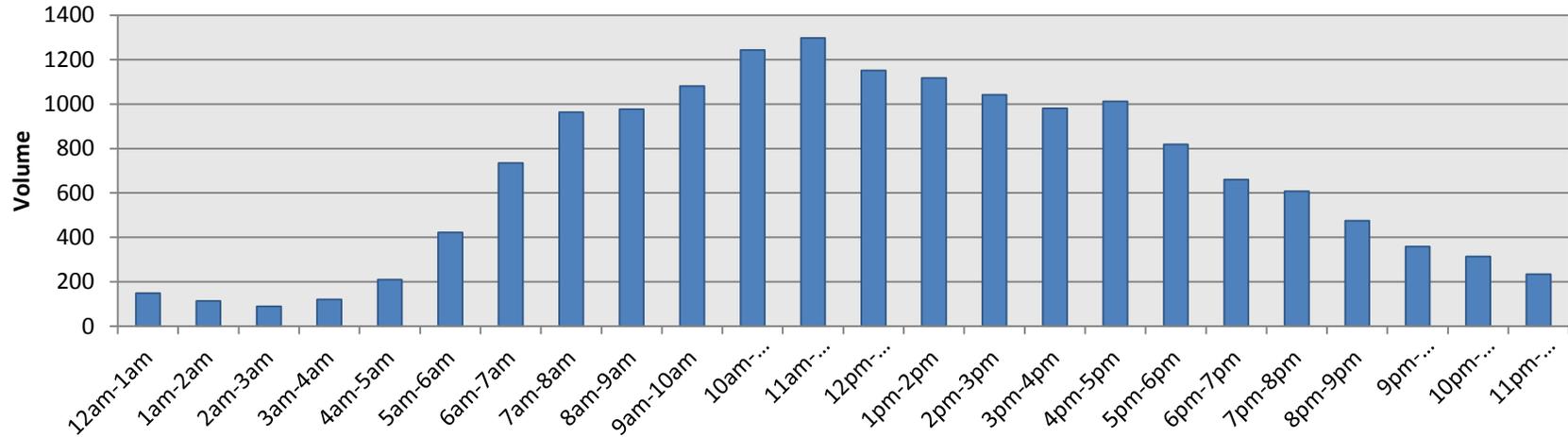


Figure 4-10: Typical Monday in October 2015 (FDOT)

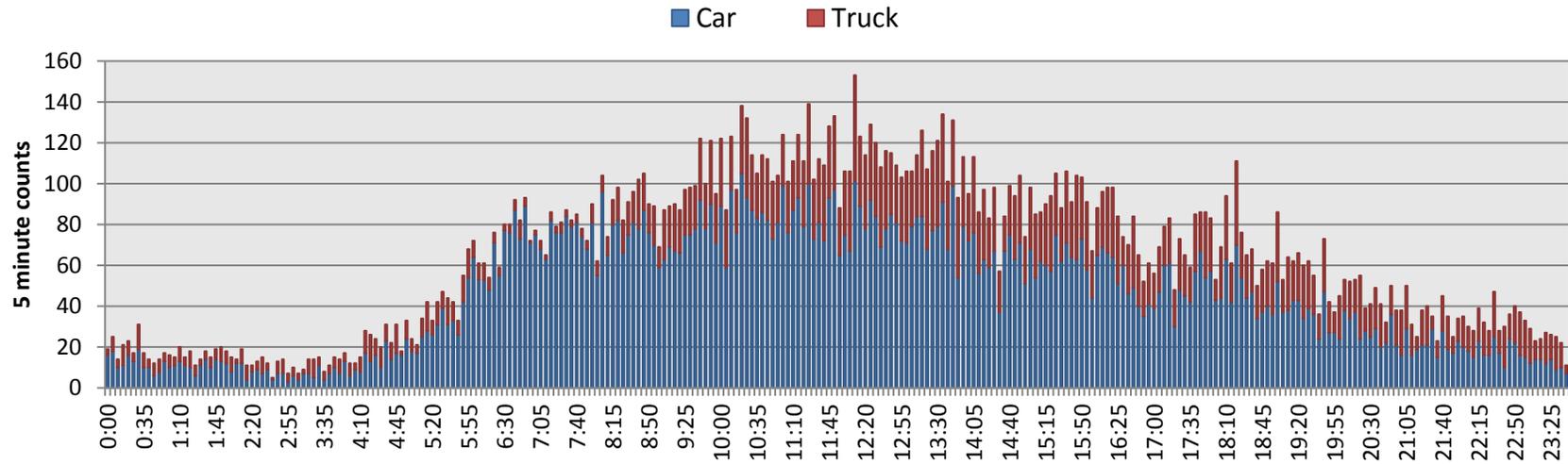


Figure 4-11: Vehicle Counts per Time Interval (5-minute)

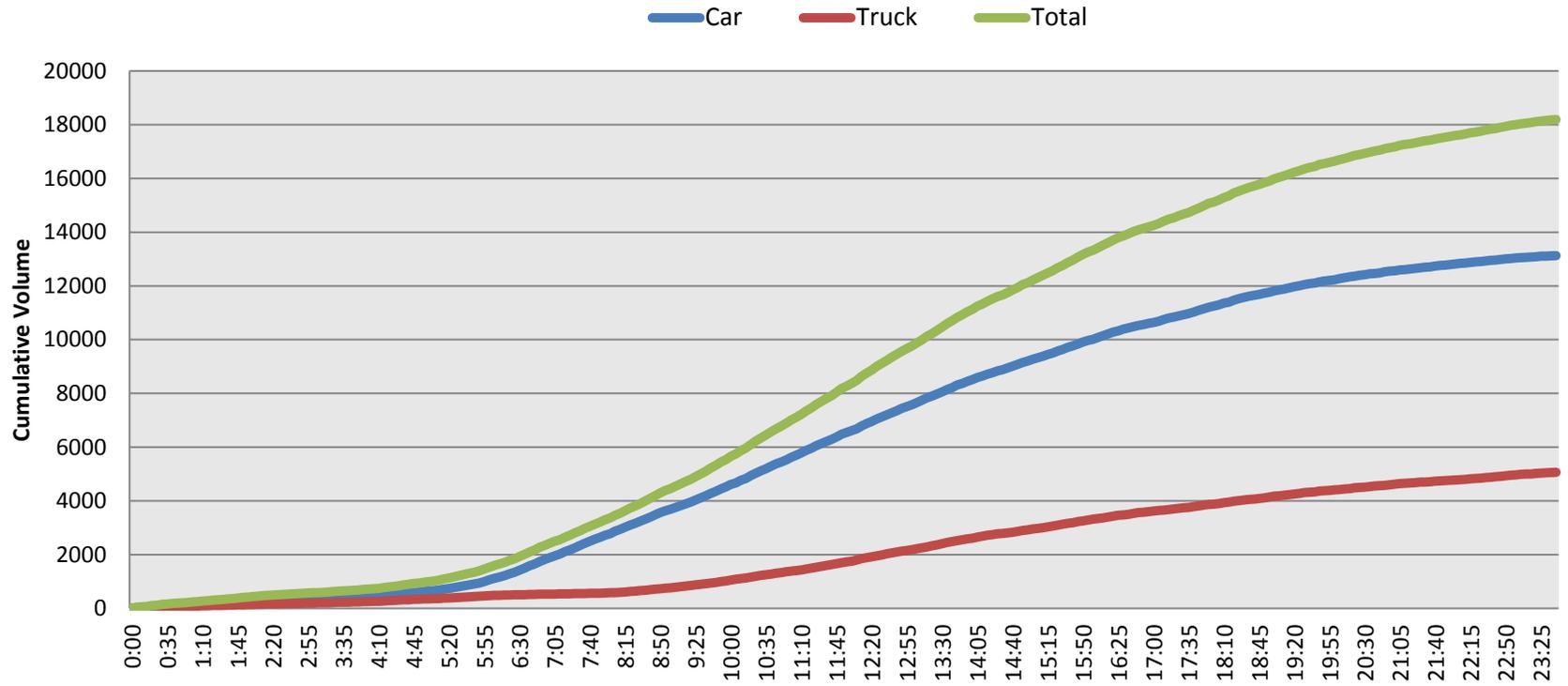
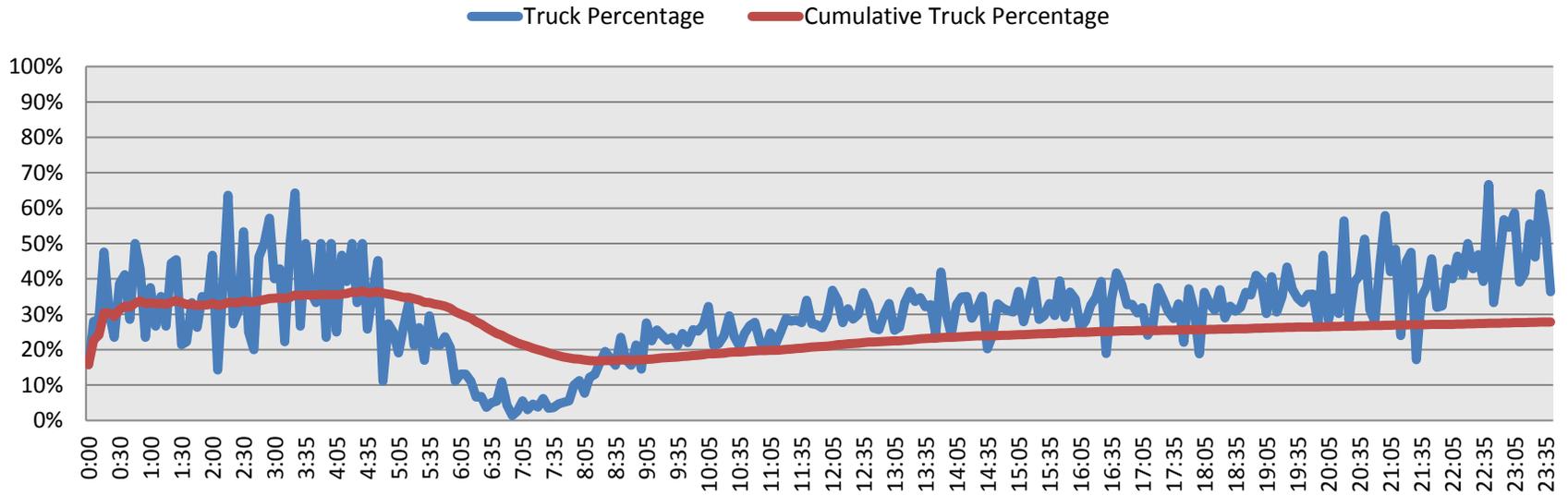
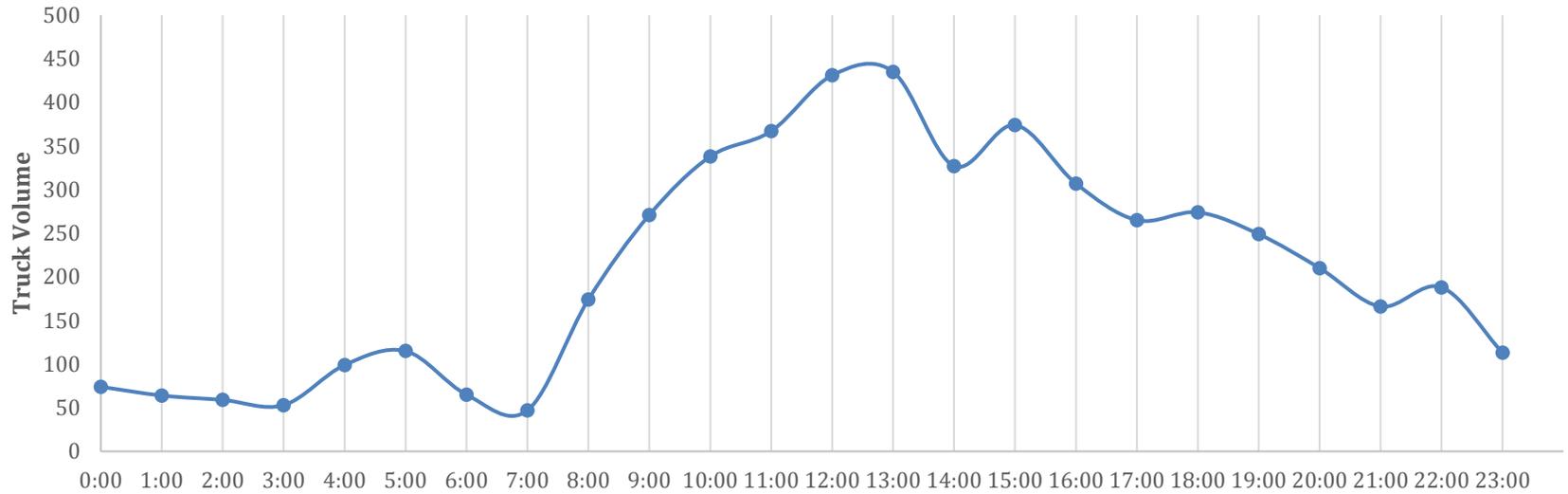


Figure 4-12: Number of Cars, Trucks, and Total Vehicles (Cumulative)



**Figure 4-13: Truck Percentages**



**Figure 4-14: Hourly Truck Volumes**

### 4.3.2 Wednesday 11/02/2016

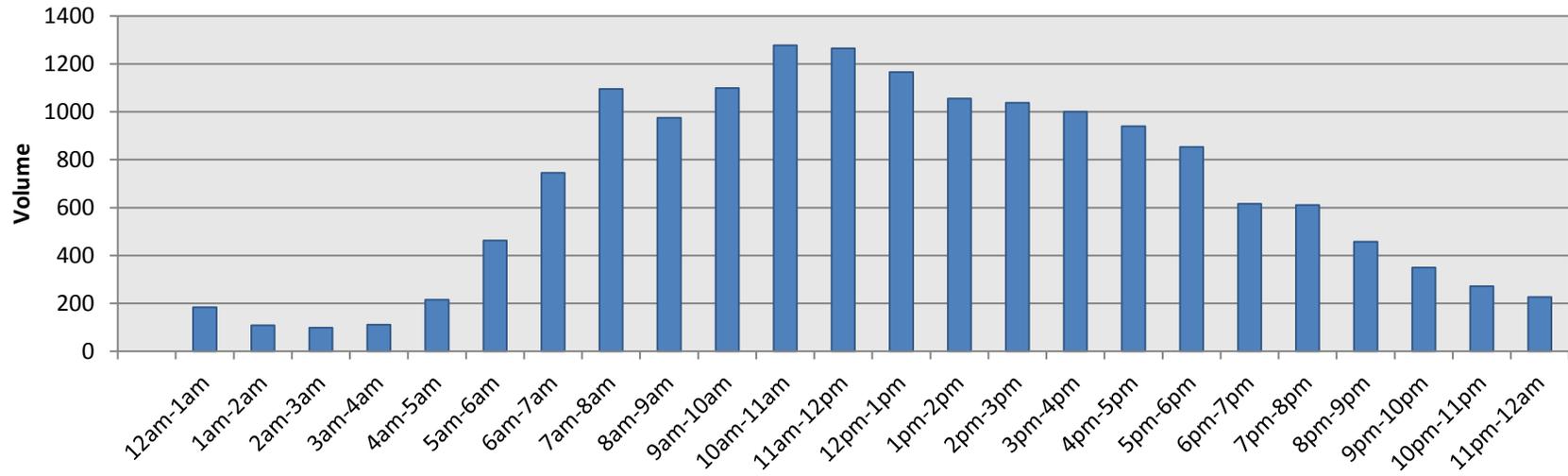


Figure 4-15: Typical Wednesday in November 2015 (FDOT)

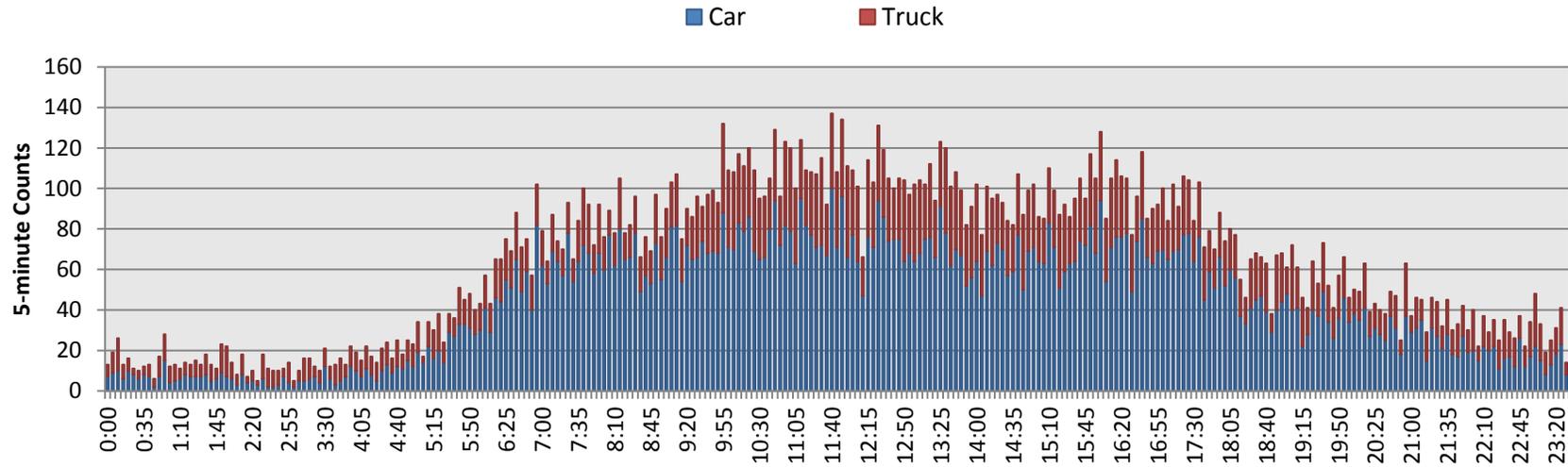
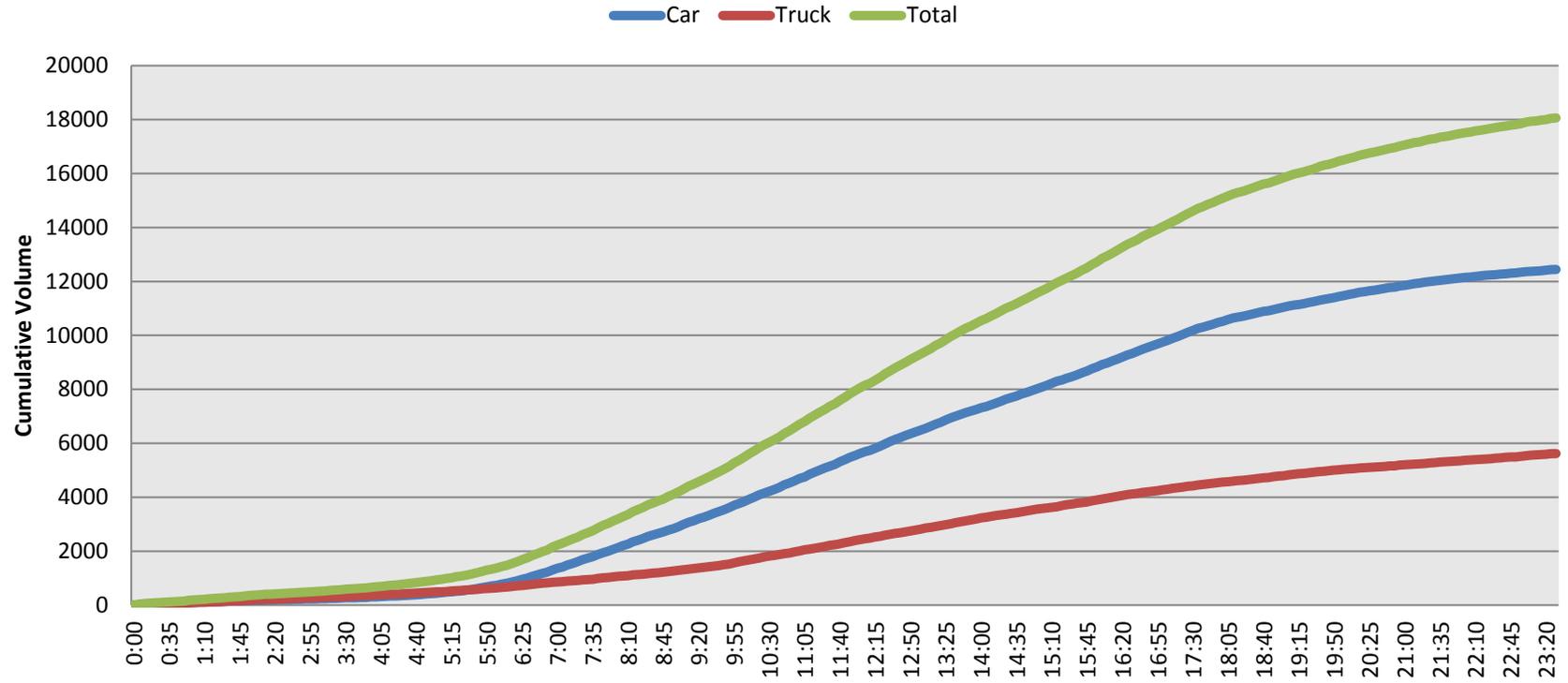
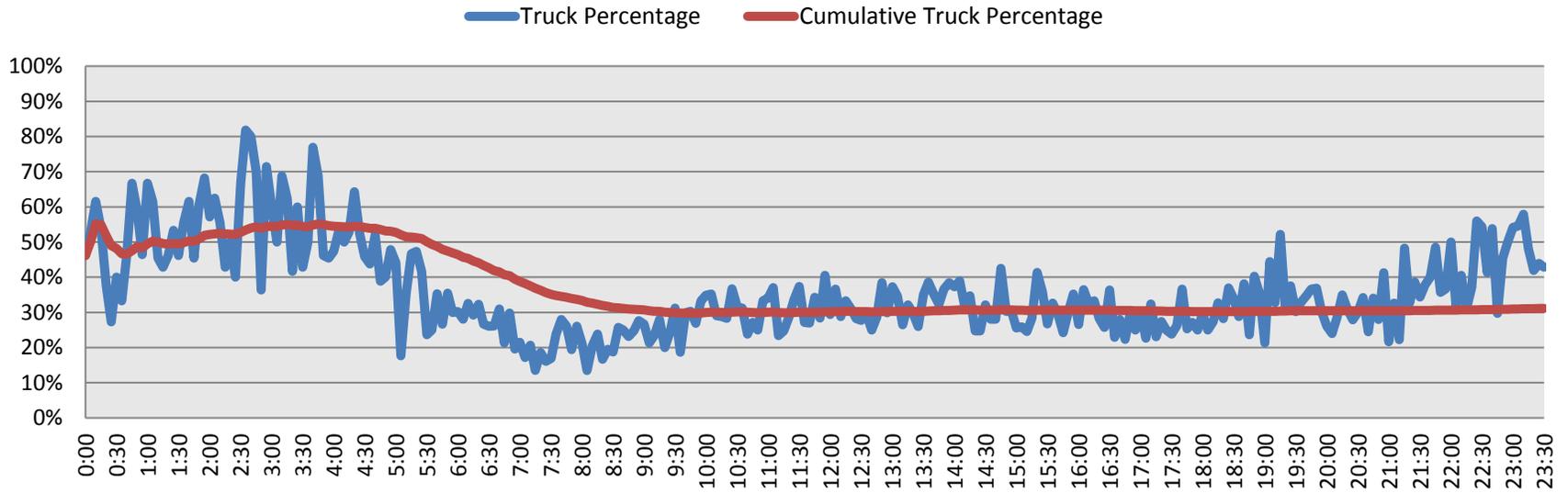


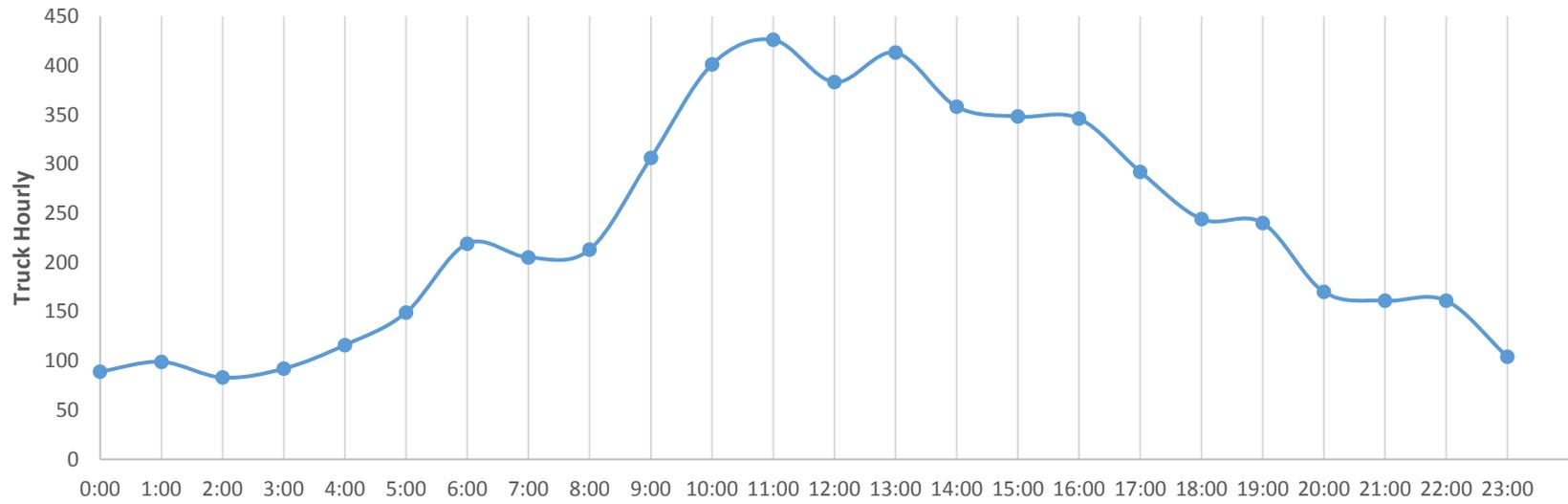
Figure 4-16: Vehicle Counts per Time Interval (5-minute)



**Figure 4-17: Number of Cars, Trucks, and Total Vehicles (Cumulative)**



**Figure 4-18: Truck Percentages**



**Figure 4-19: Hourly Truck Volumes**

### 4.3.3 Friday 11/04/2016

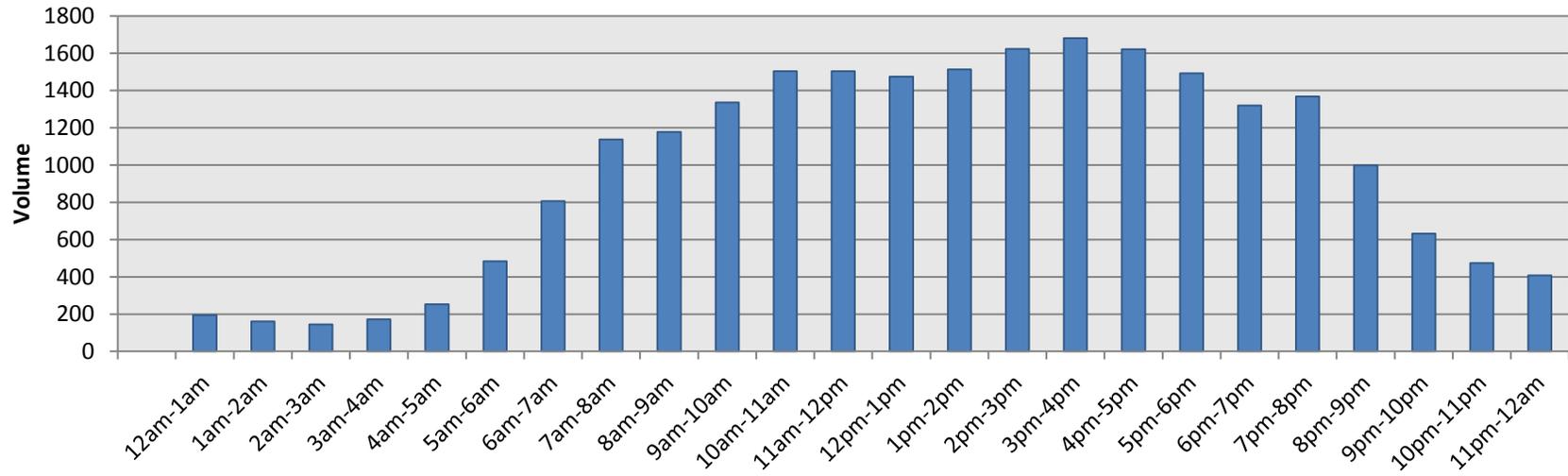


Figure 4-20: Typical Friday in November 2015 (FDOT)

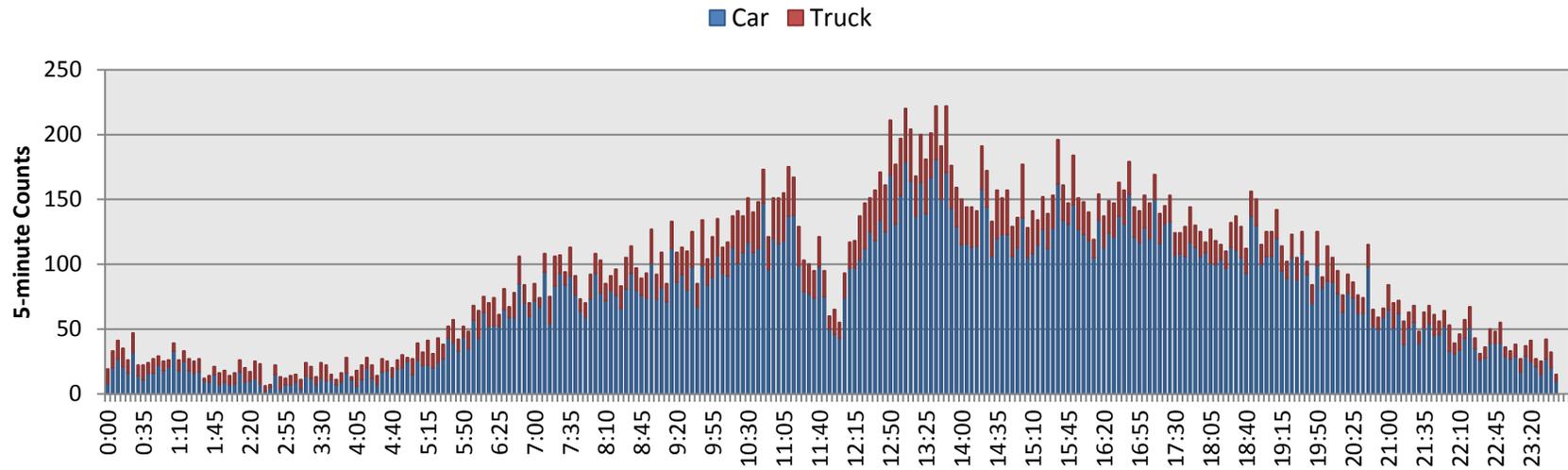


Figure 4-21: Vehicle Counts per Time Interval (5-minute)

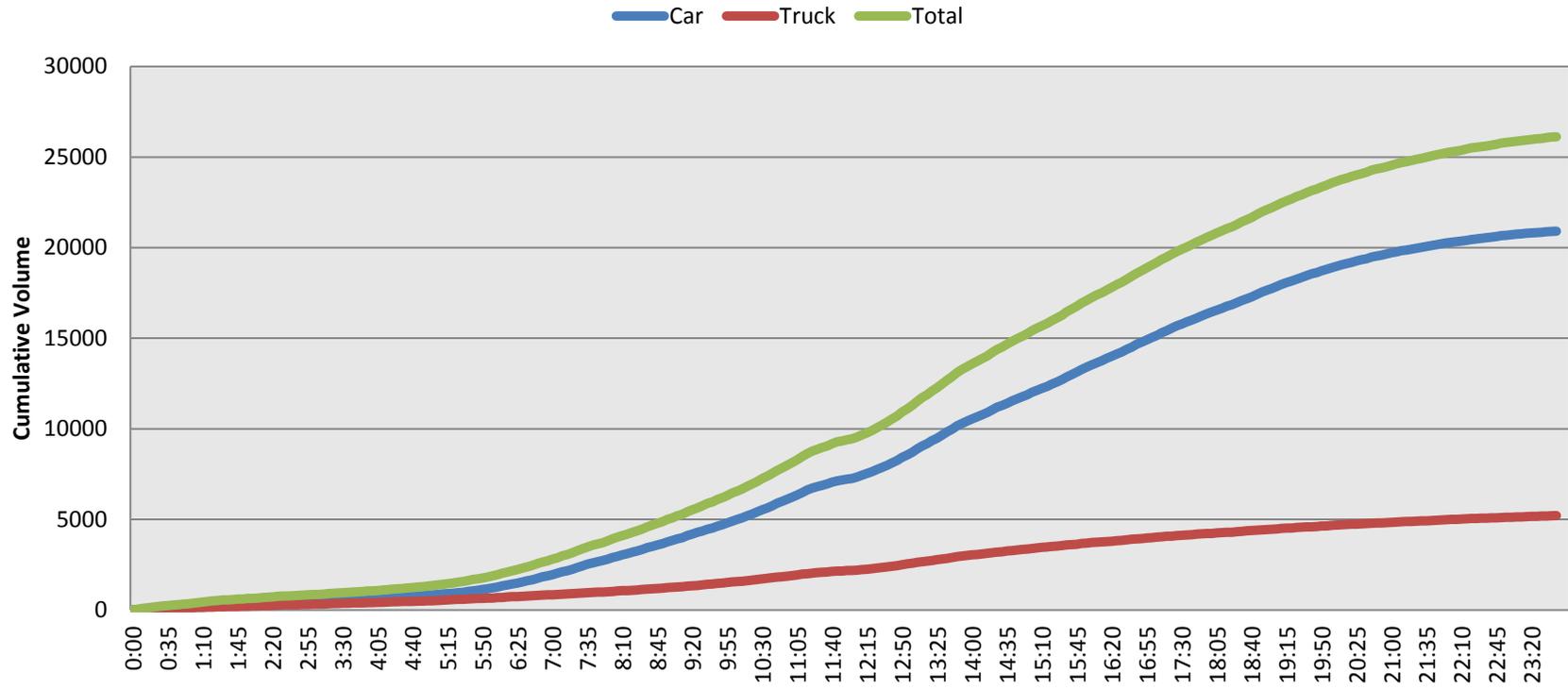


Figure 4-22: Number of Cars, Trucks, and Total Vehicles (Cumulative)

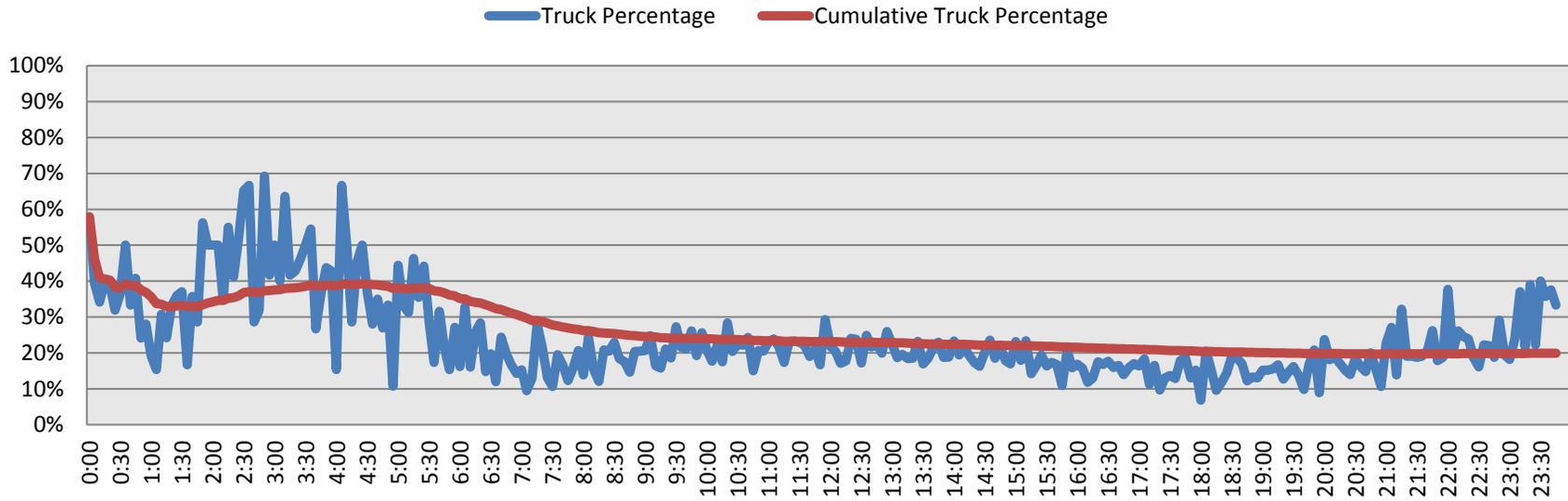


Figure 4-23: Truck Percentages

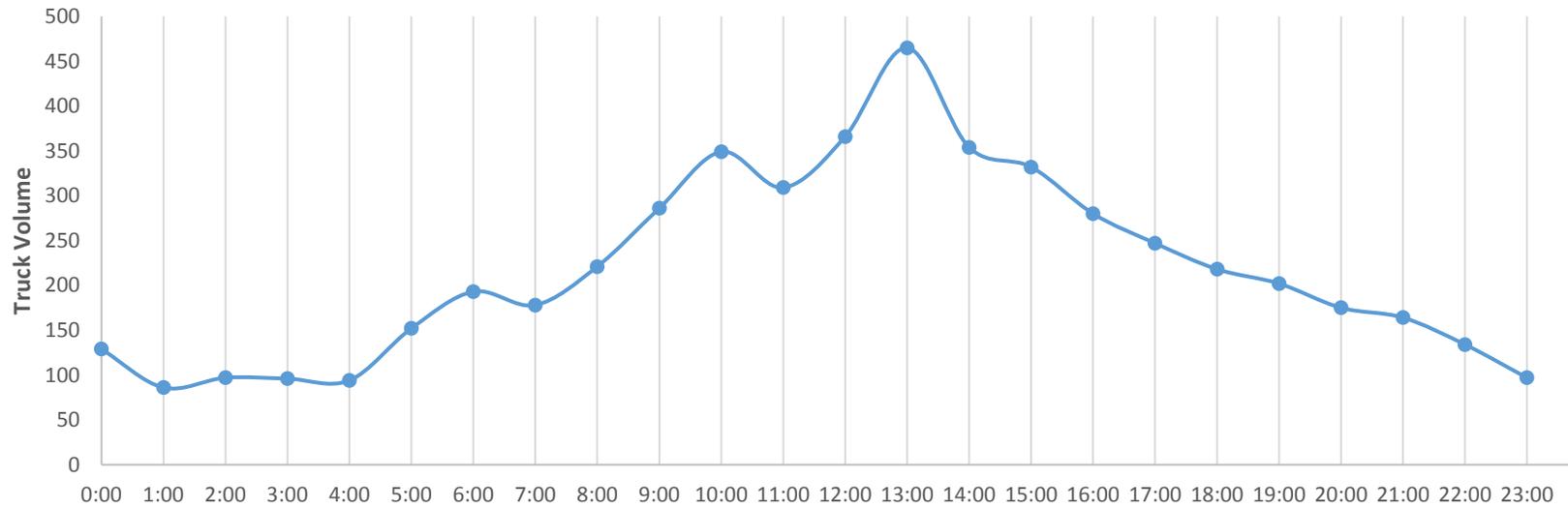


Figure 4-24: Hourly Truck Volumes

### 4.3.4 Saturday 11/05/2016

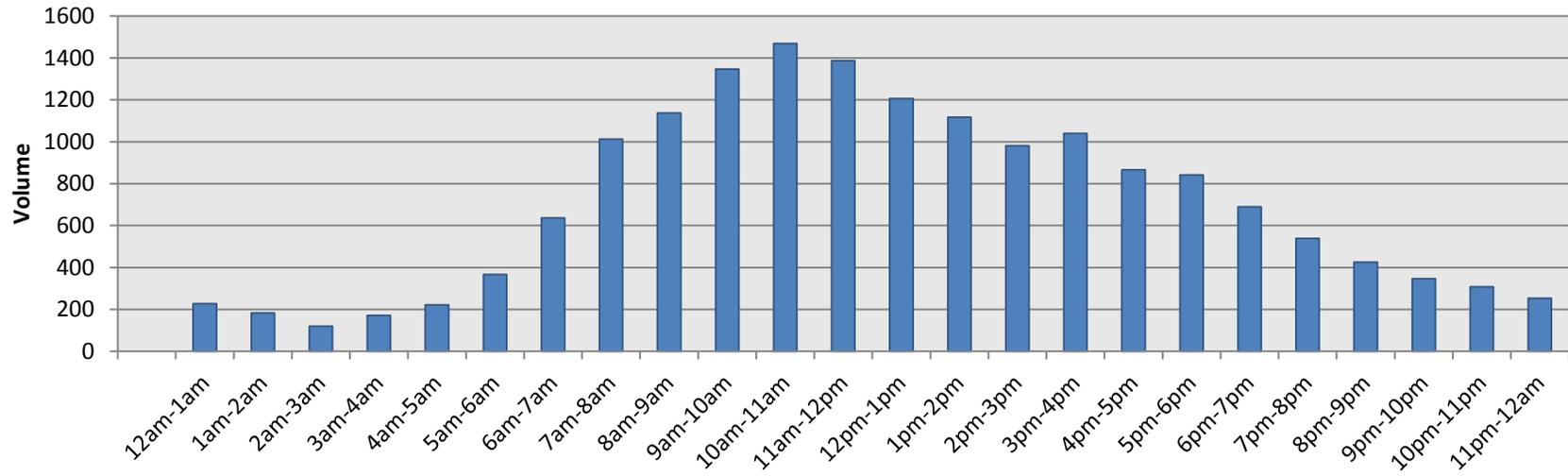


Figure 4-25: Typical Saturday in November 2015 (FDOT)

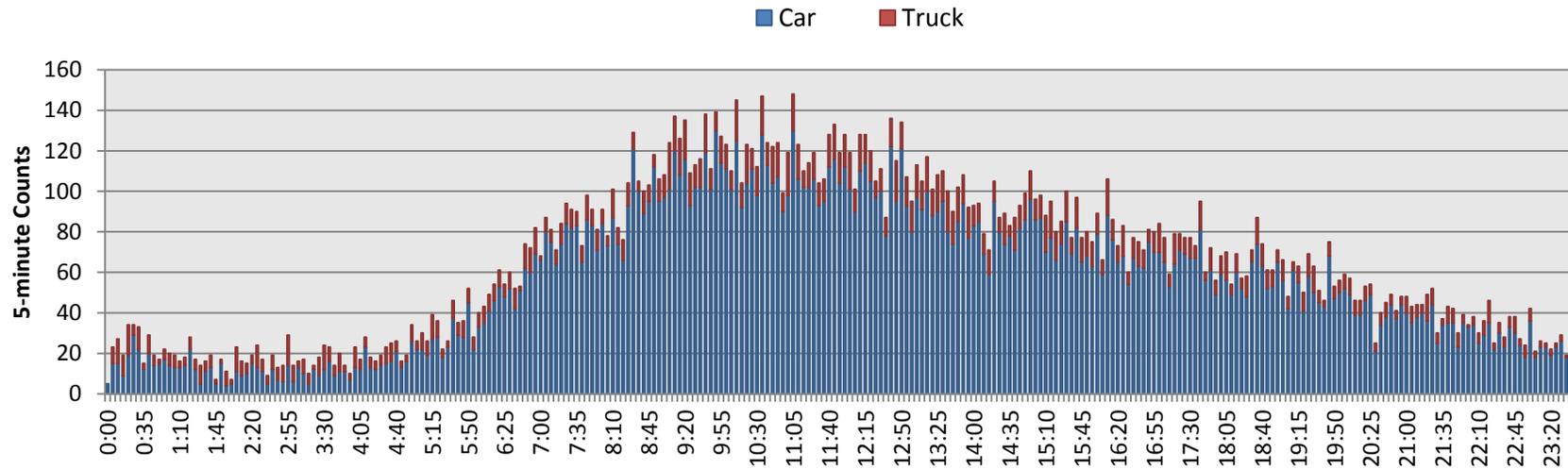


Figure 4-26: Vehicle Counts per Time Interval (5-minute)

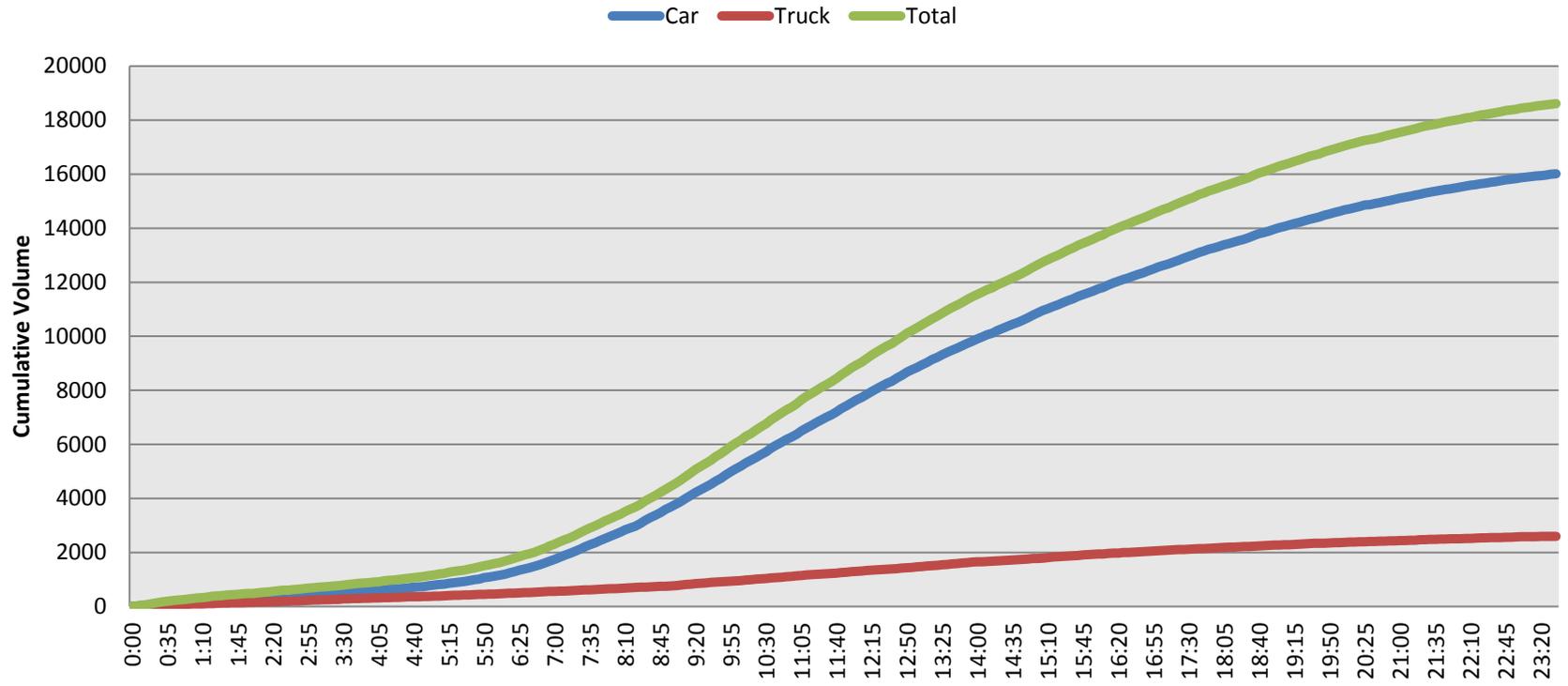


Figure 4-27: Number of Cars, Trucks, and Total Vehicles (Cumulative)

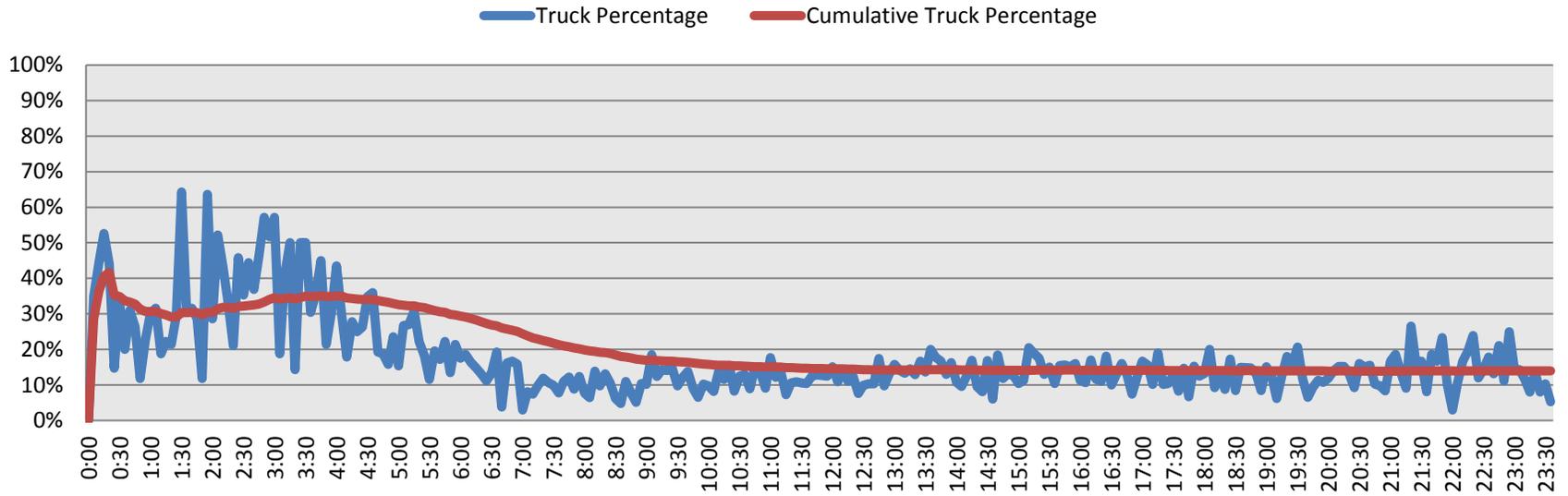


Figure 4-28: Truck Percentages

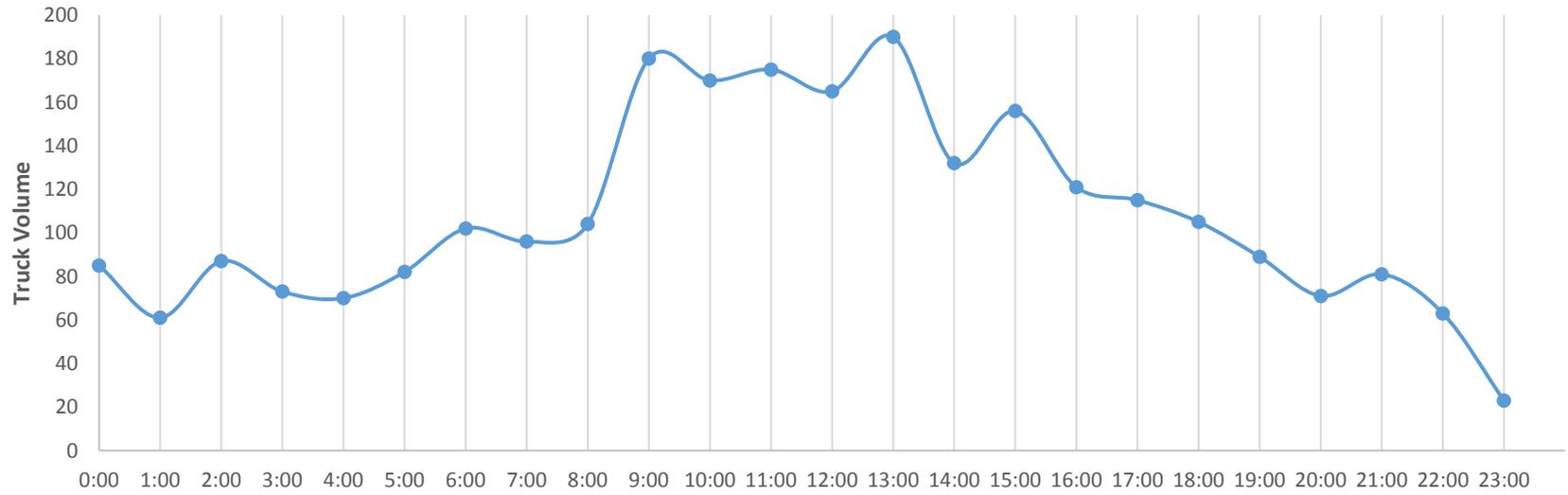


Figure 4-29: Hourly Truck Volumes

#### 4.4 LICENSE PLATE MATCHING (MODDA) RESULTS

Truck license plate information collected at the northbound overpass for CR 514/Warm Springs Ave (i.e., Location 3) were matched with truck license plates collected at the overpass for NW 63rd Street in Ocala by using MODDA. The results for Tuesday, October 25, 2016, are presented in this section.

The matching algorithm consists of three matching types: exact matches, fuzzy matches, and partial matches.

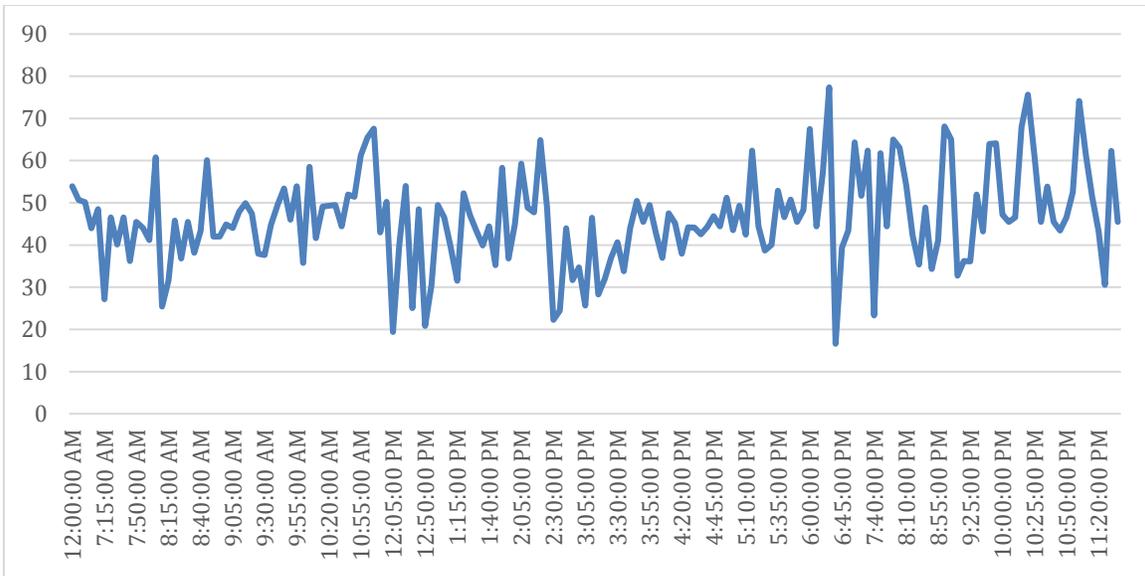
- *Exact* (Direct SQL (Structured Query Language) Match): These matches have the highest confidence because the “equals” operator is used.
- *Fuzzy* (SQL Fuzzy Match): This matching algorithm provides a high level of confidence because it takes into account typical OCR errors such as a Z for a 2 or a C for a G.
- *Partial* (Modified Levenshtein): The Levenshtein Proximity algorithm states how similar two-character strings are based on the number of changes one string has to undergo to become the same as the other string. This algorithm has been modified to take into account the typical OCR errors.

The number of license plate readings at Location 3 on the 10/25/2016 data collection day was 2141, and the number at Location 1 was 3315. Total matched plates with the above-mentioned three algorithms are listed in Table 4-5.

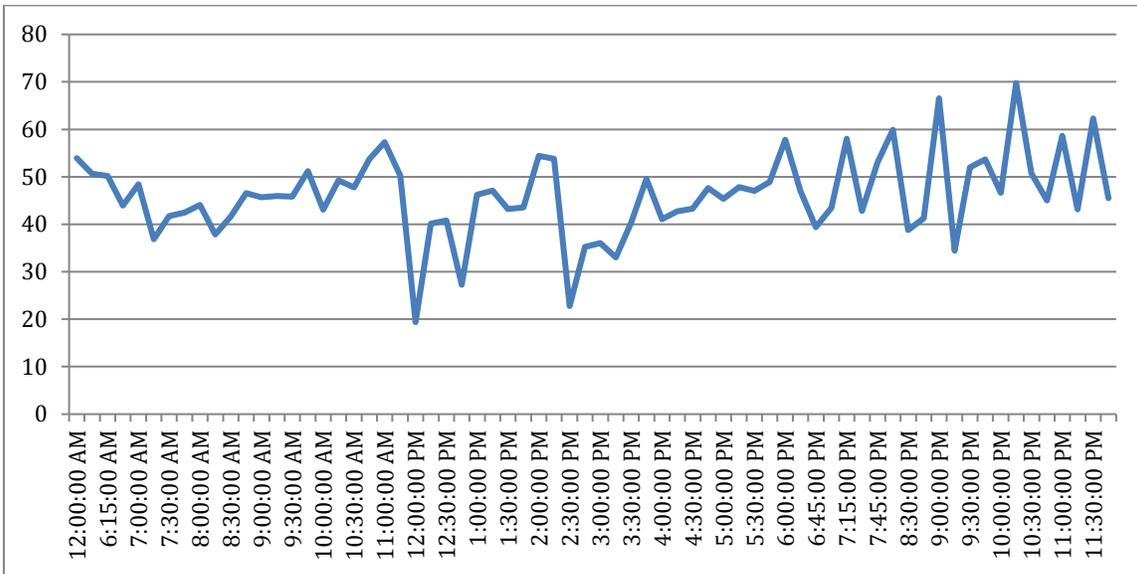
**Table 4-5: Matched Plates**

<b>Match Type</b>	<b>Frequency</b>
SQL Exact Match	409
Modified Levenshtein	163
SQL Fuzzy Match	10
Total	582

The northbound I-75 corridor starting from Location 3 and ending at Location 1 is 46.8 miles. Average speeds are computed by using the timestamps of matched vehicles. Average speeds at 5-minute and 15-minute intervals are presented in Figure 4-30 and Figure 4-31, respectively.



**Figure 4-30: 5-Minute Intervals – Average Speeds in mph**



**Figure 4-31: 15-Minute Intervals – Average Speeds in mph**

## Chapter 5 CONCLUSIONS

This study confirmed that the data collection system developed by UCF in this project can be used to acquire license plate images of trucks traveling at highway speeds. These images can then be read and compared to locations downstream of the travel direction. By matching these readouts, transportation planners can determine travel routes, times, directions, and speed of trucks.

These origin/destination data are extremely accurate, can be collected 24/7 (i.e., day and night) with relatively little cost. The installation itself does not require any invasive roadway work and, other than MOT (which can be done at night), causes little disruption to traffic.

The technology does not require drivers to have any specialized hardware installed in their vehicles, so there is no need for Radio Frequency Identification (RFID) tags, Bluetooth, or other devices to be present. Also, the technology only selects large vehicles (higher than 100 inches), which reduces privacy concerns associated with using brute force approaches such as tags or cell phones.

The findings from the Location 3 two-week data collection indicate a similar trend to historical traffic data collected by FDOT in 2015. Traffic counts collected by the UCF system at Location 3 were within 107 percent of FDOT's weekday average, and truck counts were within approximately 109 percent of FDOT's average in year 2015.

## REFERENCES

Associated Press. (2005, October 15). Tracking Cell Phones for Real-time Traffic Data. Retrieved November 27, 2012, from MSNBC:  
[http://www.msnbc.msn.com/id/9698139/ns/technology\\_and\\_science-wireless/t/tracking-cell-phones-real-time-traffic-data/#.UHgqbWl27yc](http://www.msnbc.msn.com/id/9698139/ns/technology_and_science-wireless/t/tracking-cell-phones-real-time-traffic-data/#.UHgqbWl27yc)

Dow, P. T. (2010, December 10). Directive No. 2010-5. Retrieved November 26, 2012, from The Official Web Site for the State of New Jersey:  
<http://www.state.nj.us/oag/dcj/agguide/directives/Dir-2010-5-LicensePlateReadersl-120310.pdf>

Giamo, G. (2011, January). ALPR Camera Surveys in Ohio—Experience and Lessons Learned. Retrieved November 25, 2012, from Travel Survey Methods:

Haas, R., Carter, M., Perry, E., Trombley, J., Bedsole, E., & Margiotta, R. (2009, January). iFlorida Model Deployment Final Evaluation Report. Retrieved November 29, 2012, from the Research and Innovative Technology Administration National Transportation Library: <http://ntl.bts.gov/lib/31000/31000/31051/14480.htm>

Houston Transtar. (n.d.). Houston TranStar and Bluetooth Traffic Monitoring. Retrieved November 26, 2012 from Houston Transtar:

Krygsman, S., & de Jong, T. (2008May 25). Deriving Transport Data with Cellphones: Methodological Lessons from South Africa. Retrieved November 27, 2012, from the International Steering Committee for Travel Survey Conferences:  
<http://isctsc.let.fr/papiers/workshop%20final%20version/63%20B4%20Krygsman%20and%20DeJong.pdf>

Kwon, J., & Varaiya, P. (2005, March 5). Real-Time Estimation of Origin–Destination Matrices with Partial Trajectories from Electronic Toll Collection Tag Data. Retrieved November 28, 2012, from UC Berkeley—Electrical Engineering and

Lieb, D. A. (2007, September 23). MoDOT Tracking Cell Phone Signals to Monitor Traffic Speed, Congestion. Retrieved November 27, 2012, from Southeast Missourian:  
<http://www.semissourian.com/story/1279072.html>

Maine Legislature. (2003). Title 23, §1980: Penalties. Retrieved November 25, 2016, from Maine Revised Statutes:  
<http://www.mainelegislature.org/legis/statutes/23/title23sec1980.html>

Maine Legislature. (2009). Title 29-A, §2117-A: Use of Automated License Plate Recognition Systems. Retrieved November 25, 2012, from Maine Revised Statutes:  
<http://www.mainelegislature.org/legis/statutes/29-a/title29-Asec2117-A.html>

New Hampshire Legislature. (2007, July 16). SB 0041. Retrieved November 25, 2012, from The New Hampshire General Court:  
<http://www.gencourt.state.nh.us/legislation/2007/sb0041.html>

Rescot, R. A. (2011, August 3). Feasibility of Bluetooth Data as a Surrogate Analysis Measure of Traffic. Retrieved November 27, 2012, from The University of Kansas:  
[http://kuscholarworks.ku.edu/dspace/bitstream/1808/8381/1/Rescot\\_ku\\_0099D\\_11736\\_DATA\\_1.pdf](http://kuscholarworks.ku.edu/dspace/bitstream/1808/8381/1/Rescot_ku_0099D_11736_DATA_1.pdf)

Resource Systems Group. (2011). S.R. 23 Origin Destination Study Utilizing Bluetooth Technology. Retrieved November 27, 2012, from Resource Systems Group:  
<http://www.rsginc.com/transportation/transportation-market-research/s-r-23-origin-destination-study-utilizing-bluetooth-technology/>

Rossetti, M. D., & Baker, J. (2000). Applications and Evaluations of Automated License Plate Reading Systems. Retrieved November 25, 2012, from the University of Arkansas:  
[http://www.uark.edu/~rossetti/conference\\_papers/its\\_2000\\_alpr.pdf](http://www.uark.edu/~rossetti/conference_papers/its_2000_alpr.pdf)

The Traffic Group, Inc. (2011). Projects. Retrieved November 25, 2012, from The Traffic Group: <http://www.trafficgroup.com/projects/ALPR%20Beaufort%20County.html>

Wilson, E. (2004, April 4). Real-time Traffic Monitoring Using Mobile Phone Data. Retrieved November 27, 2012, from Smith Institute for Industrial Mathematics and System Engineering: <http://www.smithinst.ac.uk/Projects/ESGI49/ESGI49-Vodafone/Report/TrafficMonitoring.pdf>

Wunnava, S. V., Yen, K., & Babij, T. (n.d.). Travel Time Estimation Using Cell Phones (TTECP) for Highways and Roadways. Available from the Florida Department of Transportation: [http://www.dot.state.fl.us/research-center/Completed\\_Proj/Summary\\_TE/FDOT\\_BD015\\_12\\_rpt.pdf](http://www.dot.state.fl.us/research-center/Completed_Proj/Summary_TE/FDOT_BD015_12_rpt.pdf)

Zhong, D. (2012). Origin and Destination Study with Bluetooth Detectors. Retrieved November 27, 2012, from the Transportation Research Board:  
<http://onlinepubs.trb.org/onlinepubs/conferences/2012/NATMEC/Zhong.pdf>

## APPENDIX-A SAMPLE CODES

### A.1 CODE – CAMERA SOFTWARE

```
Public Function OpenTheCamera() As Integer
    Try
        '-----
        Const k_fmt7Mode As Mode = Mode.Mode0
        '-----
        PrintBuildInfo()
        '-----
        ReDim m_RawImages_Cap(intNumFrames - 1)
        ReDim m_QueuedImages_Cap(intNumFrames - 1)
        '-----
        Dim busMgr As ManagedBusManager = New ManagedBusManager()
        Dim numCameras As UInteger = busMgr.GetNumOfCameras()
        Debug.Print("Number of cameras detected: {0}", numCameras)
        If numCameras < 1 Then
            Debug.Print("No cameras detected. Press enter to exit...")
            Debug.Print("-----")
            ' Return 0
        End If

        Debug.Print("Camera Serial Number = " & CInt(strCameraSerial))
        Dim guid As ManagedPGRGuid =
busMgr.GetCameraFromSerialNumber(CInt(strCameraSerial))
        '-----
        m_camera = New ManagedCamera()
        m_camera.Connect(guid)
        m_camCtlDlg.Connect(m_camera)
        '-----
        ' Get the camera information
        Dim camInfo As CameraInfo = m_camera.GetCameraInfo()
        PrintCameraInfo(camInfo)
        '-----
        ' Query for available Format 7 modes
        Dim supported As Boolean = False
        Dim fmt7Info As Format7Info = m_camera.GetFormat7Info(k_fmt7Mode,
supported)
        PrintFormat7Capabilities(fmt7Info)

        fmt7ImageSettings = New Format7ImageSettings()
        fmt7ImageSettings.mode = k_fmt7Mode
        fmt7ImageSettings.offsetX = 0
    End Try
End Function
```

```

fmt7ImageSettings.offsetY = 0
fmt7ImageSettings.width = fmt7Info.maxWidth
fmt7ImageSettings.height = fmt7Info.maxHeight
'-----
Debug.Print("Color = " & strMonoColor)
If strMonoColor = "C" Then
    fmt7ImageSettings.pixelFormat =
FlyCapture2Managed.PixelFormat.PixelFormatRaw8 'Get best FPS because processing
is done on computer
Else
    fmt7ImageSettings.pixelFormat =
FlyCapture2Managed.PixelFormat.PixelFormatMono8
End If
Dim settingsValid As Boolean = False
Dim fmt7PacketInfo As Format7PacketInfo =
m_camera.ValidateFormat7Settings(fmt7ImageSettings, settingsValid)
If (settingsValid <> True) Then
    ' Settings are not valid
    MsgBox("Format 7 Settings are not valid")
End If
m_camera.SetFormat7Configuration(fmt7ImageSettings,
fmt7PacketInfo.recommendedBytesPerPacket)
'-----
' Get embedded image info from camera
Dim embeddedInfo As EmbeddedImageInfo =
m_camera.GetEmbeddedImageInfo()

' Enable timestamp collection
If (embeddedInfo.timestamp.available) Then
    embeddedInfo.timestamp.onOff = True
End If

' Set embedded image info to camera
m_camera.SetEmbeddedImageInfo(embeddedInfo)
'-----
' Power on the camera
Const k_cameraPower As UInteger = &H610
Const k_powerVal As UInt32 = &H80000000UI
m_camera.WriteRegister(k_cameraPower, k_powerVal)

' Wait for camera to complete power-up
Const k_millisecondsToSleep = 100
Dim regVal As UInteger = 0

Do While ((regVal And k_powerVal) = 0)
    System.Threading.Thread.Sleep(k_millisecondsToSleep)

```

```

    regVal = m_camera.ReadRegister(k_cameraPower)
Loop
'-----
' Set embedded timestamp to on
// Dim embeddedInfo As EmbeddedImageInfo =
m_camera.GetEmbeddedImageInfo()
embeddedInfo.timestamp.onOff = True
m_camera.SetEmbeddedImageInfo(embeddedInfo)
'-----
intCurrentStrobeVal = 0
SetMyStrobe() 'Turn Off
'-----
' Get the camera configuration
Dim config As FC2Config = m_camera.GetConfiguration()
'-----
config.grabMode = GrabMode.BufferFrames
'set numBuffers (default = 10)
config.numBuffers = Buffer_Size
'-----
' Set the grab timeout to 50 seconds
config.grabTimeout = 50000

' Set the camera configuration
m_camera.SetConfiguration(config)
'-----
' Camera is ready, start capturing images
m_camera.StartCapture()
m_grabImages = True
StartGrabLoop()
'-----
If (useSoftwareTrigger) Then
    Debug.Print("Trigger the camera by sending a trigger pulse to GPIO." & "," &
vbNewLine & "," & MyTriggerMode.source)
End If
'-----
'Start Timers
If intWithTimers = 1 Then
    TriggerTimer.Enabled = True
    ResetTimer.Enabled = True
End If
'-----
ExecuteCameraSettings()
'-----
Return 1
Catch ex As Exception

```

```

End Try
End Function

```

## A.2 CODE – TRIGGER SOFTWARE

```

Private Sub DeviceNewData()
    Try
        Do While (True)
            +-----+
            +-----+
            If intSensorType = 4 Then
                If MainFormRef.cbFreeze0.Checked = False And
MainFormRef.cbShowSensor0.Checked = True Then
                    MainFormRef.fLVDetections0.Invoke(DirectCast(Function()
MainFormRef.fLVDetections0.Items.Clear()
                                                                    End Function,
Action))

                    End If

                    Dim uTimestamp As UInteger
                    Dim readings As CReadings
                    '-----
                    dtDeviceNewData = Now
                    '-----

                    If GetDetections(MainFormRef.nuNumDet0.Value, uTimestamp,
readings) And bAcquire = True Then

                        *****
                            If bStuckTruck = True Then
                                Try
                                    '-----
                                    --
                                    Dim iCount0 As Integer = 0
                                    Dim iCount1 As Integer = 0
                                    For mm As UInteger = intSectorNumLow To
intSectorNumHigh
                                        If readings.MyAmplitude(mm) = 0 Then      'i.e.
reading very weak or wrong
                                            iCount0 = iCount0 + 1
                                        End If
                                        If (readings.MyDistance(mm)) * sConversion >
dTrigDist Then
                                            iCount1 = iCount1 + 1
                                        End If

                                        Dim iDiff As Integer
                                        iDiff = intSectorNumHigh - intSectorNumLow + 1
                                        If iDiff = 1 Then
                                            iDiff = 2
                                        End If
                                        If iCount0 >= intTrigCounter Or iCount1 >=
intTrigCounter Then
                                            'Reset Sensor

```



```

        If readings.MyAmplitude(kk) >= sMinAmplitude
And (readings.MyDistance(kk)) * sConversion <= dVehTrigDist Then
            intLaneNum = 2
            If bLaneBusy(intLaneNum) = False Then
                bLaneBusy(intLaneNum) = True
                intVehNum(intLaneNum) =
intVehNum(intLaneNum) + 1
            If intVehNum(intLaneNum) >
intVehArrSize Then
                intVehNum(intLaneNum) = 1
                WriteLaneVehicles(intLaneNum)
            End If
            strVehTime((intLaneNum),
intVehNum(intLaneNum)) = Now.ToString("HH:mm:ss.fff")
            End If

            If bLaneBusy(intLaneNum) = True Then
                bLaneBusy(intLaneNum) = False
                lLane(intLaneNum) = lLane(intLaneNum)
+ 1
            End If
        End If
    End If
Catch ex As Exception
    Debug.Print("Vehicle Stats E R R O R " &
ex.Message)
End Try

'=====  

=====  

        If MainFormRef.cbCamLock.Checked = False Then
            '-----  

                If (bStuckTruck = False) And ((bDOTTrigger =
False) And (kk >= intSectorNumLow) And (kk < intSectorNumDeadLow)) Or ((kk >
intSectorNumDeadHigh) And (kk <= intSectorNumHigh)) Then
                    If readings.MyAmplitude(kk) >= sMinAmplitude
And (readings.MyDistance(kk)) * sConversion <= dTrigMinDist And
(readings.MyDistance(kk)) * sConversion <= dTrigDist And (readings.MyDistance(kk))
* sConversion <= dTrigMaxDist Then
                        Dim elapsedSpan As TimeSpan = New
TimeSpan(Now.Ticks - dtDOTTriggerTime.Ticks)
                    '=====  

                    =====  

                        If elapsedSpan.TotalMilliseconds >
CInt(MainFormRef.tbClearInterval.Text) Then
                            '-----  

                                Try
                                    If MainFormRef.cqPics.Count <=
(intNumThreads * (MainFormRef.nuMaxQ.Value)) Then
                                        '-----  

                                        bStuckTruck = True
                                        bAcquire = False
                                        dtDOTTriggerTime = Now
                                        '-----

```

CommObject

= kk 'Sector number triggered (may be used to detect lane)

-----  
(kk < intSectorNumDeadLow) Then

lLaneTrucks(intLaneNum) + 1

And (kk <= intSectorNumHigh) Then

lLaneTrucks(intLaneNum) + 1

-----  
btInput

MainFormRef.lTruckID + 1

dtDOTTriggerTime 'Now

-----  
MainFormRef.lTrucksIn + 1

MainFormRef.cbSafeCam.Enqueue(MyDetectedObject)

..... >>>> Distance: " & readings.MyDistance(kk) \* sConversion & " Segment  
= " & kk)

(readings.MyDistance(kk) \* sConversion)

CInt(sLastTriggerVal)

on Sector # " & kk)

-----  
MainFormRef.cbShowResults.Checked = True Then

Dim MyDetectedObject As New

Dim btInput(7) As Byte

btInput(0) = 10

btInput(1) = 0

btInput(2) = 0

btInput(3) = 0

btInput(4) = 0

btInput(5) = 0

btInput(6) = 0

btInput(7) = 0

MyDetectedObject.intSectorNum

-----

If (kk >= intSectorNumLow) And

intLaneNum = 1

lLaneTrucks(intLaneNum) =

End If

If (kk > intSectorNumDeadHigh)

intLaneNum = 2

lLaneTrucks(intLaneNum) =

End If

-----  
MyDetectedObject.btInput =

MyDetectedObject.lMyTruckID =

MyDetectedObject.dtMyTime =

-----  
MainFormRef.lTrucksIn =

Debug.Print(">>> Car coming

sLastTriggerVal =

sActTriggerDist =

Debug.Print("T R I G G E R

-----  
If

```

MainFormRef.SetLabel(82,
"Dist: " & (readings.MyDistance(kk) * sConversion).ToString("F1") & ", " & kk)
End If
'-----
End If
bAcquire = True
Catch ex As Exception
End Try
End If
'-----
If MainFormRef.cbShowResults.Checked =
True Then
MainFormRef.SetLabel(70, CStr(kk) &
", " & ((readings.MyDistance(kk)) * sConversion).ToString("F1") & ", A:" &
readings.MyAmplitude(kk).ToString("F1"))
MainFormRef.SetLabel(95, "Total - Lane 1 = " & lLane(1))
MainFormRef.SetLabel(96, "Trucks - Lane 1 = " & lLaneTrucks(1))
MainFormRef.SetLabel(97, "Total - Lane 2 = " & lLane(2))
MainFormRef.SetLabel(97, "Trucks - Lane 2 = " & lLaneTrucks(2))
End If
dDistSens1 = readings.MyDistance(kk) * sConversion
dAmpSens1 = readings.MyAmplitude(kk)
intSecSens1 = kk
'=====
====
End If
End If
End If
'-----
Try
If MainFormRef.cbShowSensor0.Checked = True And
MainFormRef.cbFreeze0.Checked = False Then
Dim lItem As New ListViewItem(kk.ToString())
lItem.SubItems.Add(((readings.MyDistance(kk))
* sConversion).ToString("F2"))
lItem.SubItems.Add(readings.MyAmplitude(kk).ToString("F1"))
MainFormRef.fLVDetections0.Invoke(DirectCast(Function()
MainFormRef.fLVDetections0.Items.Add(lItem)
End Function, Action))
Else
If MainFormRef.cbFreeze0.Checked = True And
bFrozen = False Then
bFrozen = True
MainFormRef.fLVDetections0.Invoke(DirectCast(Function()
MainFormRef.fLVDetections0.Items.Clear()
End Function, Action))

```

```

                                'Show last results
                                For z = intSectorNumLow To
intSectorNumHigh
                                    Dim lItem As New
ListViewItem(z.ToString())
lItem.SubItems.Add(((readings.MyDistance(z)) * sConversion).ToString("F2"))
lItem.SubItems.Add(readings.MyAmplitude(z).ToString("F1"))
MainFormRef.fLVDetections0.Invoke(DirectCast(Function()
MainFormRef.fLVDetections0.Items.Add(lItem)
End Function, Action))
                                    Next
                                    End If
                                End If

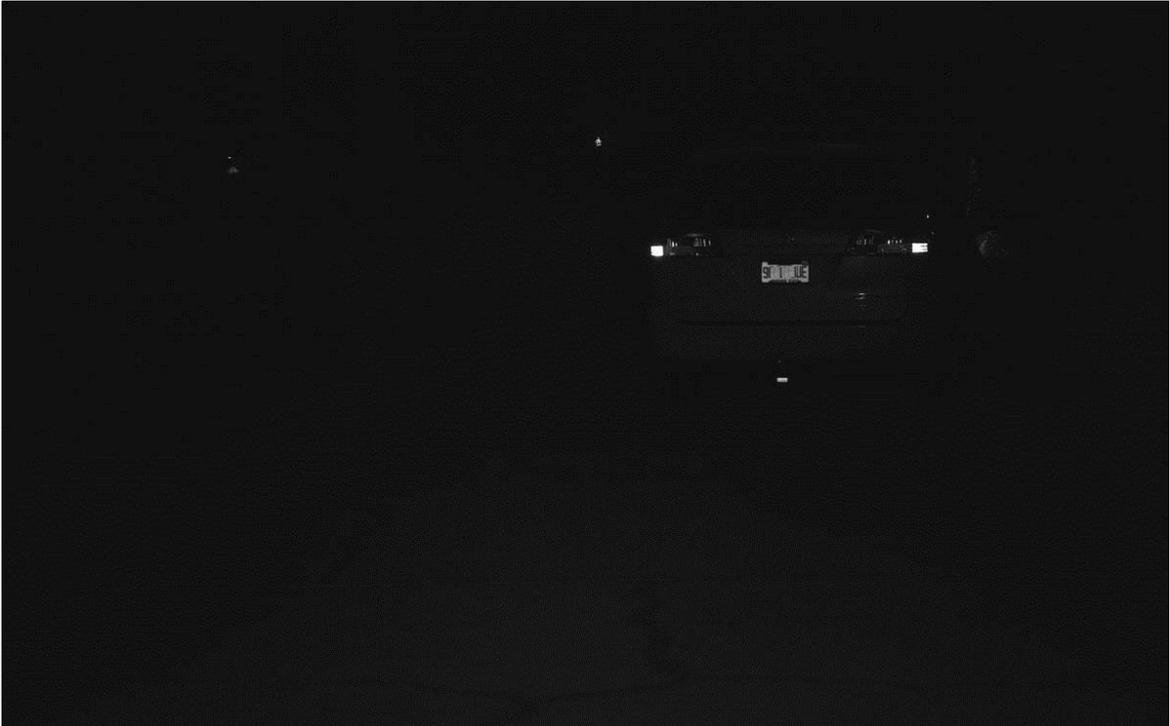
                                Catch ex As Exception
                                    MsgBox("Problem on Sensor # " & intSensorNum)
                                End Try
'-----
-----
                                Next kk
                                End If
                                End If
'
+++++
+++++
                                Loop
                                Catch ex As Exception
                                End Try
                                End Sub

```

## APPENDIX-B ADDITIONAL PHOTOS



**Figure Error! No text of specified style in document.-1: Lens Testing – Day**



**Figure Error! No text of specified style in document.-2: Lens Testing – Night**



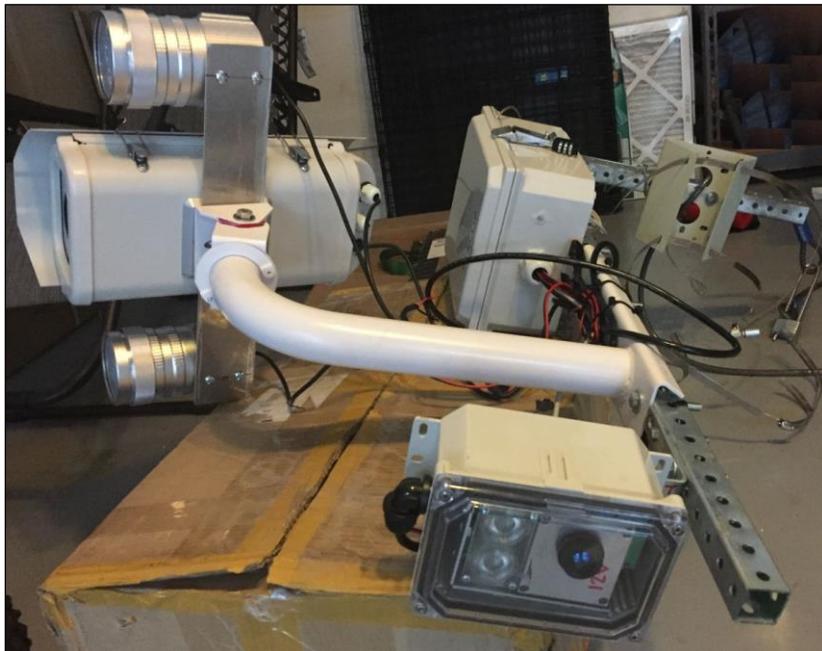
**Figure Error! No text of specified style in document.-3: Bracket on Overpass at Location 3**



**Figure Error! No text of specified style in document.-4: Camera Bracket**



**Figure Error! No text of specified style in document.-5: Bracket Close-up**



**Figure Error! No text of specified style in document.-6: Camera Bracket and Sensor Pre-installation**



**Figure Error! No text of specified style in document.-7: Bracket Connection to Overpass**



**Figure Error! No text of specified style in document.-8: Bracket Strapped to Overpass**





**Figure Error! No text of specified style in document.-9: Camera and Truck Traffic (from Overpass)**



**Figure Error! No text of specified style in document.-10: Camera and Truck Traffic (from Ground)**



**Figure Error! No text of specified style in document.-11: Solar Power Unit – Location**  
3



**Figure Error! No text of specified style in document.-12: Solar Power Unit – Location**  
3



**Figure Error! No text of specified style in document.-13: Sample Day Image – Lane 1**



**Figure Error! No text of specified style in document.-14: Sample Day Image – Lane 2**



**Figure Error! No text of specified style in document.-15: Truck Changing Lanes**



**Figure Error! No text of specified style in document.-16: Sample Day Image – Close-up**



**Figure Error! No text of specified style in document.-17: Sample Night Image – Lane 1**



**Figure Error! No text of specified style in document.-18: Sample Night Image – Close-up**



**Figure Error! No text of specified style in document.-19: Location 3 Camera Bracket**



**Figure Error! No text of specified style in document.-20: Location 3 with Sensor**



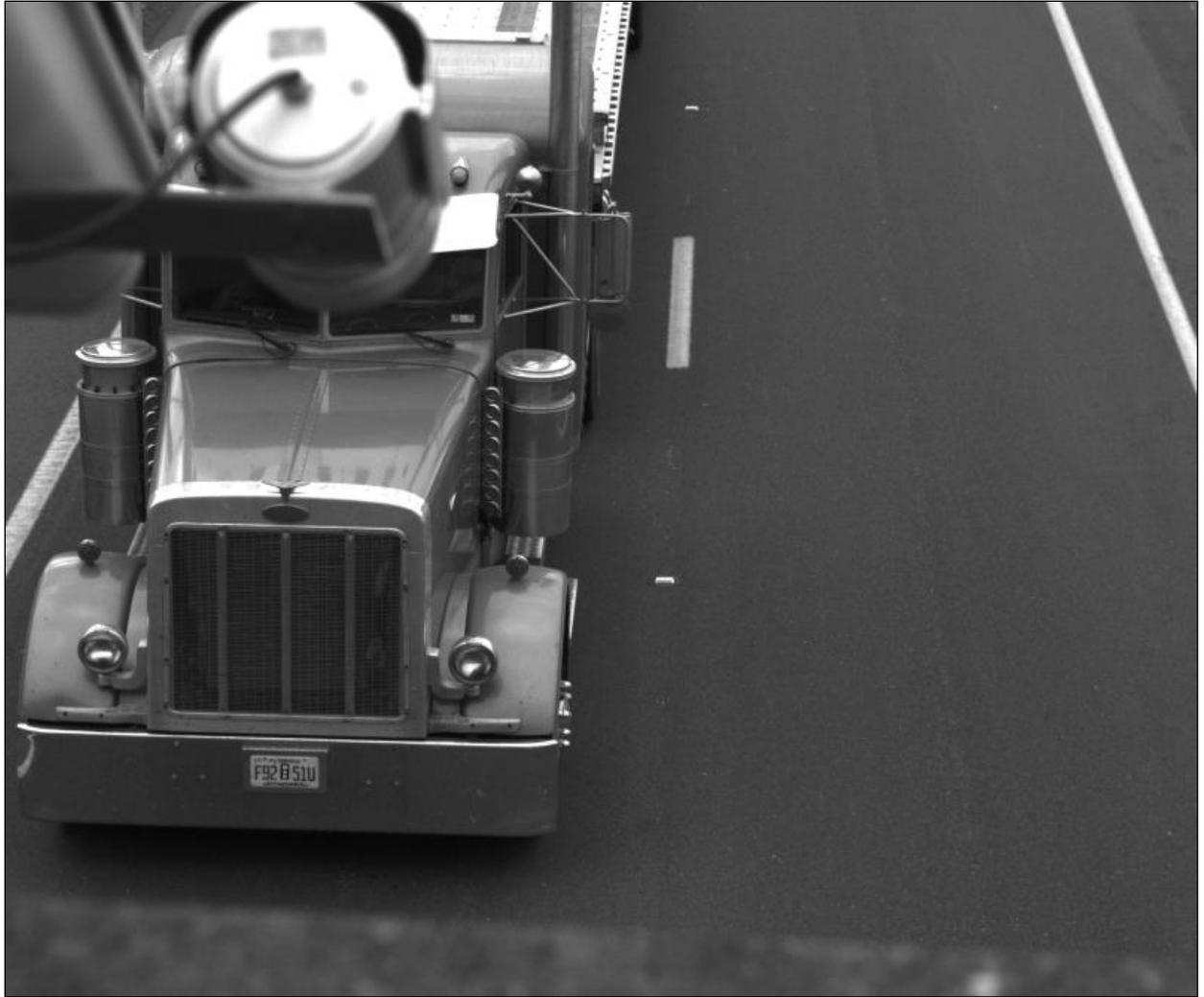
**Figure Error! No text of specified style in document.-21: Location 3 with Sensor – Approaching Traffic**



**Figure Error! No text of specified style in document.-22: Location 1 Installation1**



**Figure Error! No text of specified style in document.-23: Location 1 Installation-2**



**Figure Error! No text of specified style in document.-24: Location 2 – New Camera –  
Truck Left Lane**



**Figure Error! No text of specified style in document.-25: Location 2 – New Camera –  
Truck Right Lane**