PRODUCT 0-6838-P1

TxDOT Project Number 0-6838

Test Data, Demonstration Videos, And Transceivers

Christian Claudel
Paul Avery
Kara Kockelman

CENTER FOR TRANSPORTATION RESEARCH
BUREAU OF ENGINEERING RESEARCH
THE UNIVERSITY OF TEXAS AT AUSTIN

August 2016

http://library.ctr.utexas.edu/ctr-publications/0-6838-P1.pdf

THE UNIVERSITY OF TEXAS AT AUSTIN
CENTER FOR TRANSPORTATION RESEARCH

0-6838-P1

TEST DATA, DEMONSTRATION VIDEOS, AND TRANSCEIVERS

Christian Claudel
Paul Avery
Kara Kockelman

TxDOT Project 0-6838: Bringing Smart Transport to Texans: Ensuring the Benefit of a Connected and Autonomous Transport System in Texas

AUGUST 2016; PUBLISHED NOVEMBER 2016

Performing Organization:	**Sponsoring Organization:**
Center for Transportation Research The University of Texas at Austin 1616 Guadalupe, Suite 4.202 Austin, Texas 78701	Texas Department of Transportation Research and Technology Implementation Office P.O. Box 5080 Austin, Texas 78763-5080
Performed in cooperation with the Texas Department of Transportation and the Federal Highway Administration.	

Table of Contents

List of Figures

Section 1. Introduction

Smart-driving technologies are evolving quickly and cover a wide range of capabilities. This report describes various test data, demonstrations, and transceivers created and used during the demonstration phase of Project 0-6838. Researchers at Southwest Research Institute (SwRI) and UT Austin's Center for Transportation Research (CTR) demonstrated the connected-vehicle (CV) applications detailed in this publication utilizing state-of-the-art dedicated short-range communication (DSRC) radios, which were loaded with SwRI-built custom application software in compliance with the USDOT's Connected Vehicle Program and applicable standards. They also demonstrated roadside equipment (RSE) performance along Loop 410. This equipment along Loop 410 had been installed previously by the TxDOT San Antonio District and SwRI to develop and demonstrate advanced signage systems, and was used here to help highlight the capability of the system to support multiple parallel applications simultaneously. The demonstrations performed at the J.J. Pickle Research Center show the portability of the system and ease of configuration and installation in new locations. The CV demonstrations were presented with the context of how they could be utilized by TxDOT and integrated into their existing traffic management centers and standard operations. The CTR researchers investigated the possibility of using inexpensive CV retrofits for traffic monitoring without relying on traditional GPS systems.

Section 2. Demonstration of Technology: SwRI

SwRI operates in San Antonio, Texas using an on-site test track as well as local roads and highways as needed. The SwRI research team developed a number of demonstrations and technologies that were showcased throughout this project. The demonstrations included a subset of applications described by the USDOT Connected Vehicle program, focusing on those that have higher potential for near-term deployment and immediate safety benefits, as well as concepts for new applications that could be used by local and regional traffic management agencies. The applications included core capabilities ranging from vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications, advanced onboard sensors, infrastructure-based algorithms for vehicle trajectory detection, and new methods for communicating with vehicles that would normally be out of range of DSRC infrastructure.

2.1 Definitions

2.1.1 Emergency Vehicle Alert

Emergency vehicles, like ambulances, police cars, fire trucks, and construction vehicles, broadcast an Emergency Vehicle Alert (EVA) when they are activated. CVs receiving the EVA will analyze it to determine the emergency vehicle's direction of travel, distance, and speed, which in a real-world deployment would enable the driver, or an automated vehicle control system, to take an appropriate action, such as slow down, pull over, or continue with no change. The portable onboard device (POD) can send and receive EVA messages and each POD can be individually configured as an emergency vehicle, such as an ambulance, police car, or a fire truck. A receiving

POD determines from the emergency vehicle's basic safety message (BSM) the location, direction, and speed of the emergency vehicle. Figure 2.1 depicts an EVA demonstration.

Figure 2.1: EVA Demonstration

2.1.2 Electronic Emergency Brake Lights

A CV will broadcast an "emergency braking" message to other vehicles when the system detects a deceleration greater than a defined threshold (Figure 2.2 provides an example of such a message). This message is intended to warn other CVs that are located behind the vehicle, so they may take immediate action to reduce their speed. This application is intended to prevent the kind of sudden traffic compression, and subsequent crashes, we see in today's non-CV traffic systems.

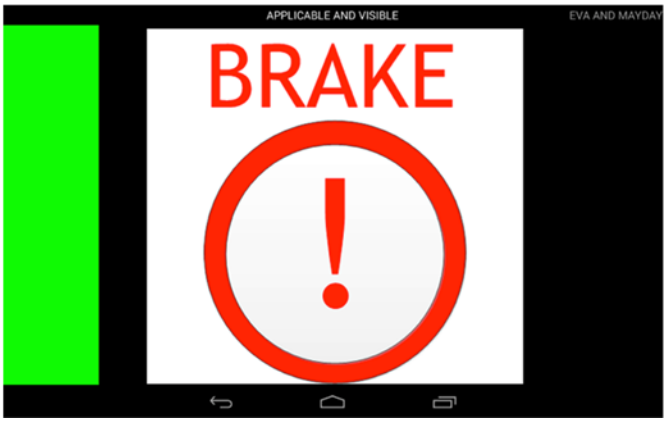

Figure 2.2: SwRI Tablet Display of 1.1.2 Electronic Emergency Brake Light Message

2.1.3 Static Wrong-way Driving Detection

CV systems have the ability to detect wrong-way drivers (WWD) using information reported by a vehicle's BSM. One way to do this is for a communications device installed in the infrastructure—roadside equipment (RSE)—to check the reported direction of a vehicle against the previously defined correct direction for traffic on a segment of road. SwRI can simulate a vehicle entering a road segment going the wrong direction using our San Antonio test track, SwRI's RSE, and vehicles equipped with PODs. The RSE receives the vehicle's BSMs and

compares the reported direction against the road segment's correct direction, in a process often referred to as *geo-fencing* or *geo-coding*. The RSE can then broadcast a Roadside Service Announcement (RSA) specifically to the WWD vehicle, as well as other vehicles within communication range (Figure 2.3).

Figure 2.3: CV WWD Messages Sent by RSE

2.1.4 Intelligent Message Propagation

This CV application would enable vehicles or infrastructure devices (such as RSEs) to pass along (propagate) messages they have received, which would be very useful in a scenario where RSE coverage is sparse or otherwise unavailable, and would enable CVs to continue to be informed of important events without RSE coverage. V2V message propagation is also viable for this application. This application would be applicable over large geographic areas with many vehicles, enabling a message to rapidly move from vehicle to vehicle. The final use of the message would depend on the message content, and could be consumed by individual vehicles, such as in the case of a weather-related warning, or could be consumed by an RSE, as in the case of a stranded motorist. In Figure 2.4, a simulated CV traffic system is shown with the effective DSRC (dedicated short-range communication) range of vehicles shown in red and that of RSEs shown in green. This illustrates how a smart transport system could be enabled without a 100% coverage of RSEs, as long as there is sufficient market penetration of CVs.

Figure 2.4: Message Propagation between RSEs

2.1.5 Road Condition Monitoring

According to current estimates, potholes cause approximately $6.4 billion in damage annually, making timely detection and repair of degraded roadways a significant concern for citizens and governments alike. Current methods for detection of poor road conditions consist of manual surveying, which is limited by the available resources of a traffic management entity. While the prevalence of smartphones has increased the ability for individuals to report road condition issues, the use of CV communication protocols presents a unique opportunity to enable vehicles to identify regions of pavement that require immediate maintenance, and to observe trends in pavement conditions over time. The necessary technologies to accomplish this, such as accelerometers, GPS-based localization systems, and CV DSRC are becoming more widely available, enabling new applications to be developed to enhance the collective situational awareness of the vehicles themselves, and of the traffic system as a whole. Figure 2.5 depicts evolution of the necessary equipment.

The research team has developed a method (the Dynamic, Distributed, Road Rating [DDRR] system) for utilizing incoming accelerometer and GPS data to quantify road roughness, which can be scaled across various spatial windows that reflect different aspects of road health. For example, a smaller spatial window will detect shorter-term anomalies in road condition, such as might be caused by a pothole or piece of debris in the road, while a larger window will detect more general roughness on a segment of road, which may indicate road surface deterioration.

Because the response of an individual accelerometer will be affected by the specific dynamics of a vehicle, including tire and suspension response, the DDRR system is first trained using the accelerometer data from a specific vehicle installation. This training is performed by driving the vehicle through a variety of speeds on smooth roads to identify the system's baseline response. Once completed, the vehicle is able to identify anomalous road pavement conditions, and communicate this data to other CV-equipped vehicles or to RSE.

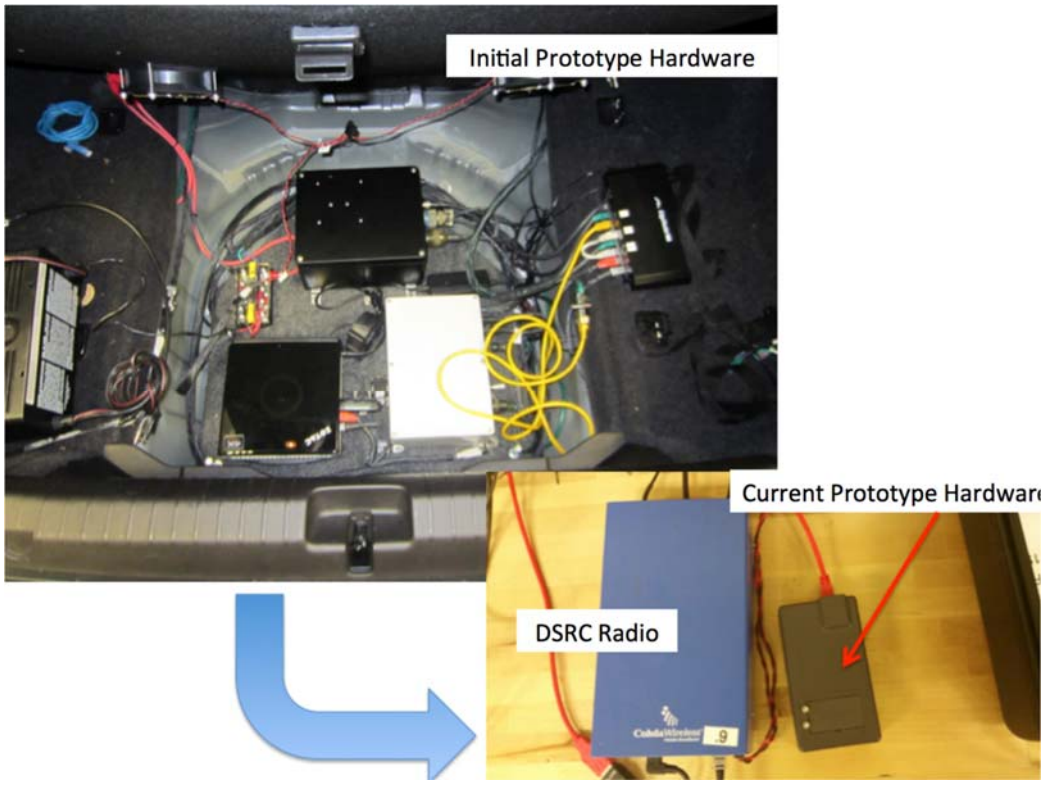

Figure 2.5: Hardware Evolution in Road Condition Monitoring

Data that has been received by another vehicle or an RSE can be used to illustrate the road conditions across a broader geographic area. The SwRI-developed method performs a clustering operation on collected road condition reports, which allows uniform display of roadway condition independent of traffic distribution. This clustered data can then be displayed using a tool such as an intensity-weighted heat map, as shown in Figure 2.6, or with individual events called out, such as in Figure 2.7.

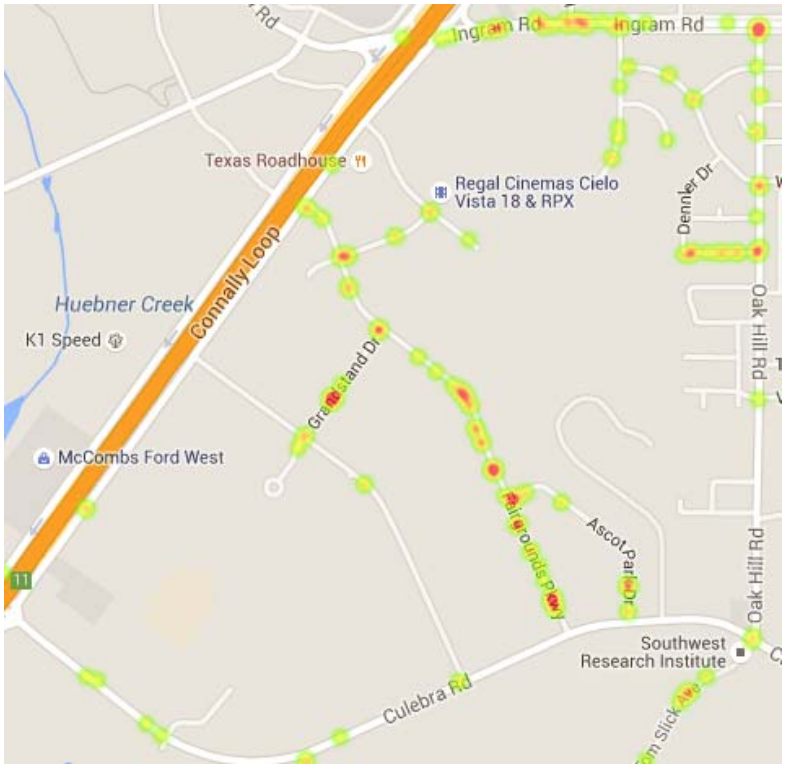

Figure 2.6: Heat Map Display of Road Condition in San Antonio, TX

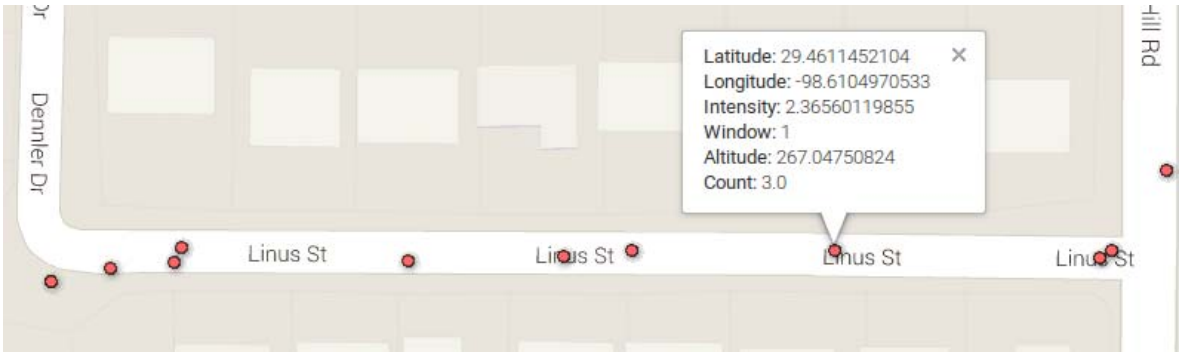

Figure 2.7: Precise Locations of Anomalous Events

2.1.6 Dynamic Wrong-way Driving Detection

Another use for the vehicle BSM is to enable an RSE to learn the location of lanes and their correct direction of travel by "listening" to vehicle BSMs as they pass through an area. Once this learning is accomplished, any CV traveling the wrong way in that section of roadway will be identified as a WWD, and the WWD warnings (as described earlier) could be initiated. This may involve the intelligent aggregation of basic vehicle state data such as GPS position, heading, and speed, passively collected by nearby infrastructure-based DSRC equipment using existing hardware solutions developed for CV deployments. This aggregated data could then be processed using learning algorithms and condensed into a set of sparse GPS waypoints that represents the lane-level roadway model. This set of waypoints can then be broadcast out by the RSE and

received by DSRC-equipped vehicles for use in numerous safety and mobility applications. The lane-level model can also be rebroadcast by vehicles to other vehicles that are not within range of the RSE, or could be broadcast and received using cellular communications, thus greatly expanding the number of vehicles that can benefit from the map data.

As vehicles repeatedly pass over lane segments, the stationary RSE collects the vehicles' BSM data, which will vary slightly from vehicle to vehicle as individual drivers may pass over a given lane segment in different positions within the lane, and due to small variations in GPS accuracy. However, the more frequently vehicles pass over the same lane segment, the more data the learning algorithm has to analyze, and the faster it can converge on a steady-state model of the lane (Figure 2.8 and 2.9). This iterative process results in an increasingly accurate representation of the centerline of a lane segment, which can be updated dynamically simply through the altered behavior of drivers. Algorithms could then detect this altered behavior after a threshold of vehicles have traversed the same segment, and the lane model can be updated and rebroadcast quickly without centralized control of the process.

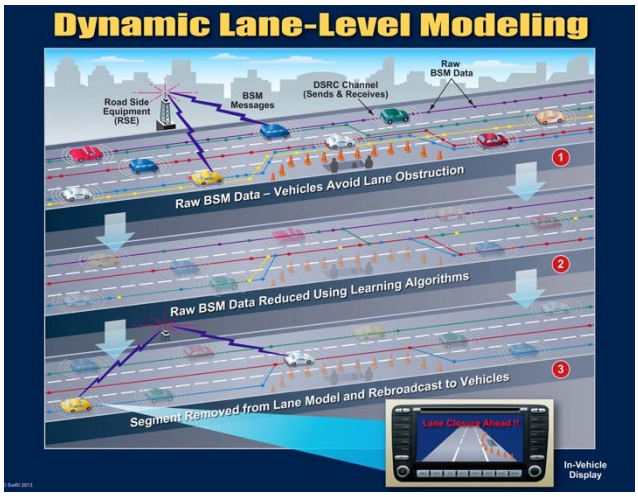

Figure 2.8: Lane Learning Concept

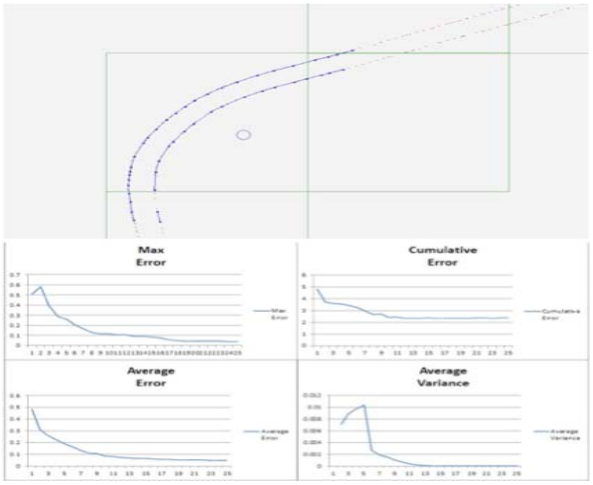

Figure 2.9: RSE Algorithm Output

2.2 Demonstrations

A series of demonstrations were held during the project to enable attendees to experience these connected and autonomous vehicle applications in a hands-on way, and to provide feedback to the research team on their experiences. These demonstrations are described below.

2.2.1 Winter 2015, J.J. Pickle Research Campus

In December 2015, the project team organized a demonstration of the applications described above at the J.J. Pickle Research Campus in Austin, TX. A quarter-mile stretch of road was closed off to normal traffic on the campus's south side (Figure 2.10) where the team conducted demonstrations to an audience of TxDOT staff and UT Austin faculty and staff not associated with the project. These demonstrations enabled the attendees to ride in CVs during the demonstrations to view first-hand how various CV applications might be implemented.

Figure 2.10: Winter Demonstration Venue Showing J.J. Pickle Research Campus (Inset), and Detail Location of Test Road and Temporary RSE

Attendees who were not riding in vehicles could view the demonstrations from a safe viewing area, and could see various DSRC hardware as well as a large TV screen that showed the real-time locations of the vehicles on a Google Earth map overlay.

2.2.2 Summer 2016, SwRI and San Antonio Roadways

In June 2016 the project team scheduled a second set of demonstrations, this time in San Antonio, Texas; however, due to inclement weather, this demonstration was rescheduled for later in the summer. These demonstrations will enable the team to highlight the installed base of RSE devices in San Antonio, with one located on SwRI's campus and three installed along Interstate 410 between Culebra Road and US 281, as shown in Figure 2.11. Specifically, the road condition monitoring, message propagation, and dynamic WWD detection and alert demonstrations will be able to take advantage of these RSEs, and demonstrate the power of these CV applications in a larger geographic area than was possible during the winter demonstration.

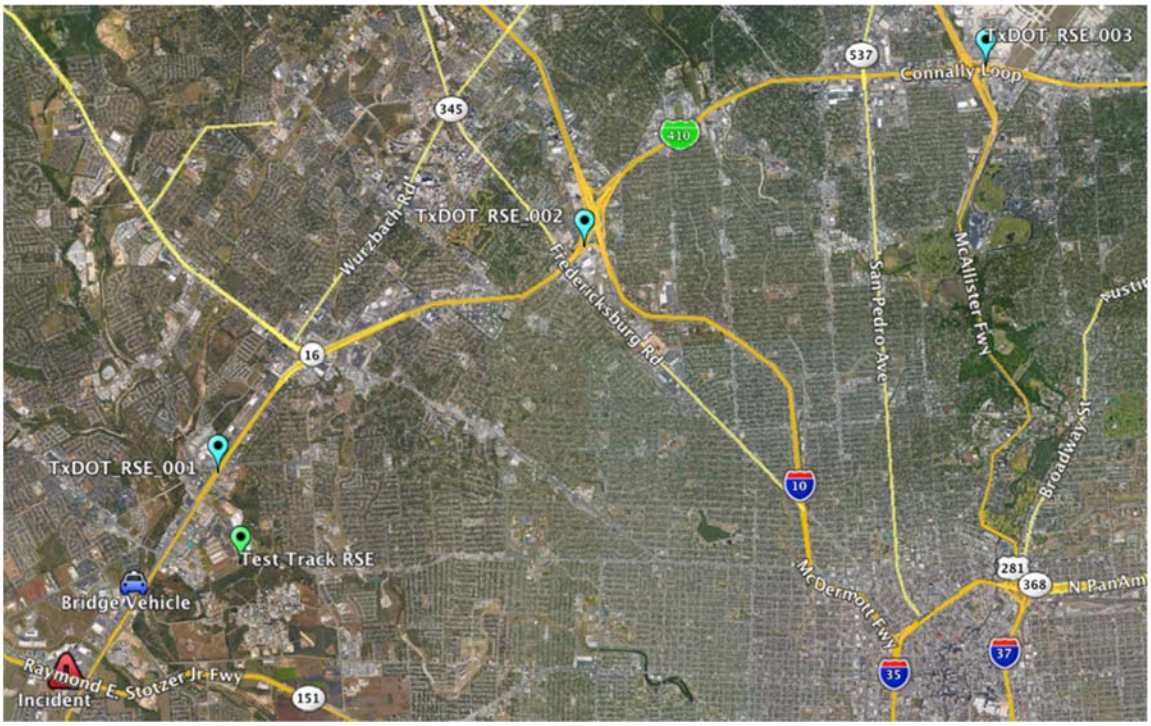

Figure 2.11: Spring Demonstration Venue Showing the Campus of SwRI and a Portion of Interstate 410 Instrumented with RSEs

2.2.3 On SwRI's Test Track: Dynamic WWD Detection and Alert with AV Safe Stop

SwRI will demonstrate a new method for implementing WWD detection using machine-learning algorithms, and will show how an automated vehicle can be safely stopped before becoming a hazard to right-way drivers. To accomplish this, SwRI will first use two or three vehicles that are equipped with the SwRI-developed system, which will be transmitting BSMs at 10hz, to drive the correct direction on our test track and through our four-way signalized intersection (Figure 2.12). The RSE located at the SwRI test track will be running a SwRI-developed machine-learning algorithm, essentially listening to the BSMs and learning the location of lanes and their correct direction of travel. Once this learning is accomplished, any CV traveling the wrong way will be identified as a WWD, and the WWD warnings will be initiated.

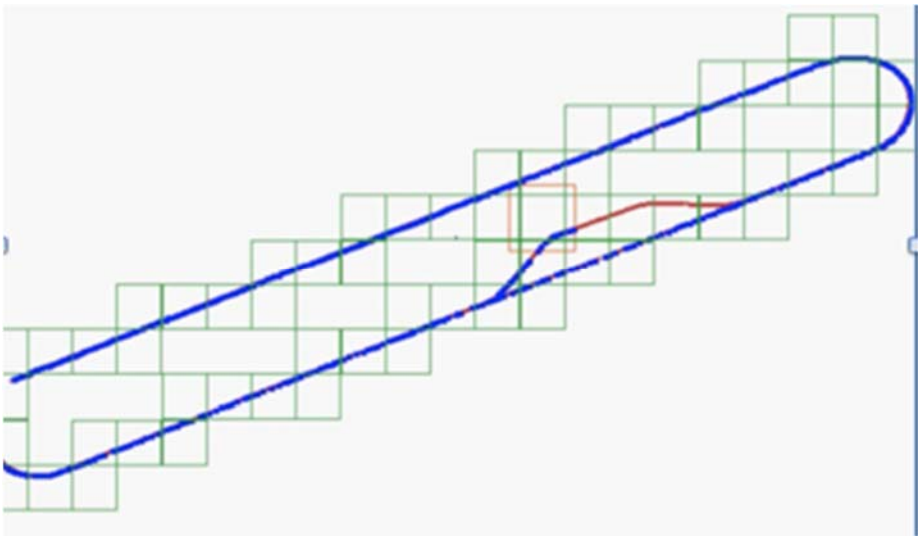

Figure 2.12: Dynamic Lane Learning at SwRI's Test Track

SwRI will also showcase one of its fully autonomous vehicles (AVs), a Class VIII Freightliner (Figure 2.14), during the dynamic WWD demonstration. The Freightliner will be sent along a route as a WWD (Figure 2.13), and once the local RSE detects this and warns the vehicle of its WWD status, the vehicle is autonomously brought to a controlled (safe) stop before it can enter the primary route for "right-way" drivers. This simulates an autonomy-capable vehicle approaching a highway the wrong direction on an exit ramp, either in an autonomous driving mode or under human control. Upon notification by the RSE, which sends out a trusted and verified message, the vehicle will slow and stop prior to entering the main lanes of the highway.

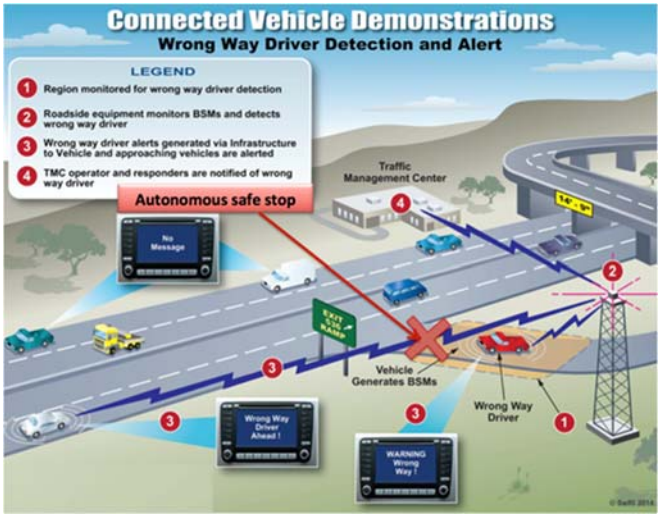

Figure 2.13: SwRI's Autonomous Freightliner Stops Once Identified As A WWD

Figure 2.14: SwRI Autonomous Freightliner

Additionally, at the SwRI test track, a large TV screen will display a Google Earth map showing semi-live updates on from mobile (off-campus) demonstrations, including road condition monitoring and message propagation. This display will give attendees a sense of the type of data

that could be available to an entity such as TxDOT, with even sparse deployment of CVs and RSEs.

2.2.4 On & Around Loop 410 in San Antonio

SwRI will use the installed base of RSEs in San Antonio to demonstrate road condition monitoring and message propagation. The road condition monitoring demonstration will use a team of vehicles to take participants onto San Antonio streets and I-410. In each vehicle, an Android tablet will display the real-time data on screens labeled “Rough Roads” and “Potholes” (Figure 2.15). Roughness events that exceed a threshold will be cached onboard until the vehicle comes within range of an RSE, at which time the data will be sent to the RSE and forwarded on to SwRI computers at the test track and displayed as a heat map of locations and severity. This data could be shared with other vehicles to warn or advise of rough roads, and would be very valuable to TxDOT for real-time maintenance awareness.

Figure 2.15: Road Condition Monitoring Tablet and Heat Map Displays

Another demonstration will involve message propagation. This CV application enables vehicles or RSEs to pass along (propagate) messages they have received. This capability would be quite useful in a scenario where RSE coverage is sparse or otherwise unavailable, and would enable CVs to continue to be informed of important events without RSE coverage. This application is best demonstrated over large areas with many vehicles; however, we will demonstrate it by driving one CV into an area that is out of range of the SwRI RSE, and a second CV will be positioned within range of the first (hidden) CV. This second (bridge) vehicle will relay (propagate) messages from the hidden CV to the RSE at the SwRI test track, and its message will be displayed on the TV over a Google Earth map overlay (Figure 2.16), along with message propagation metadata, such as the number of hops taken (in this case just one) and the time it took to propagate from source to destination.

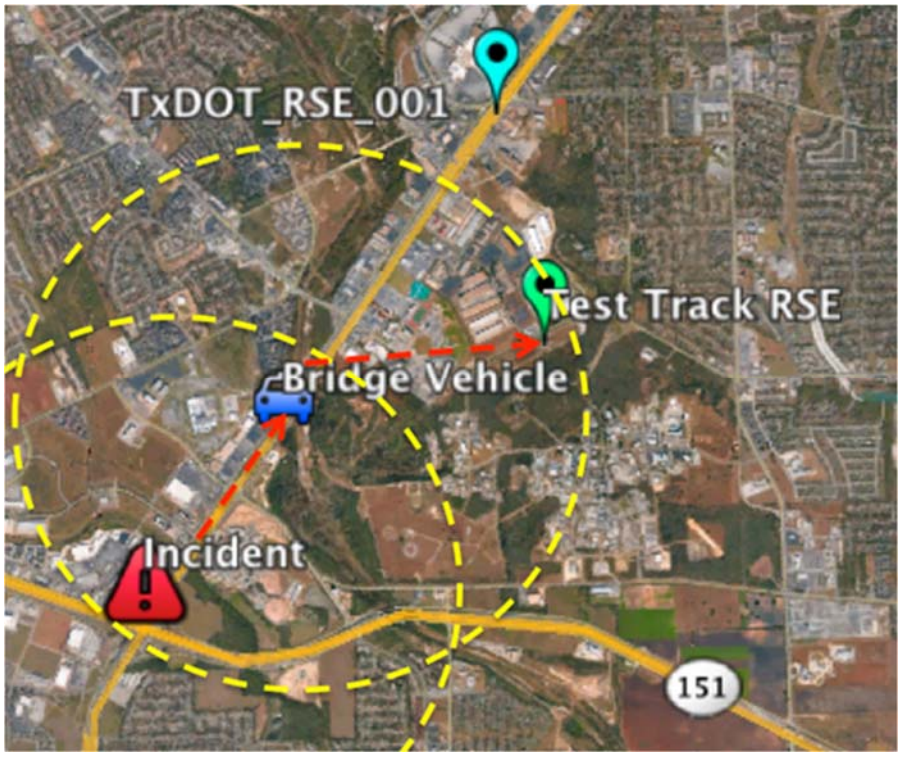

Figure 2.16: Message Propagation Demonstration Configuration

Section 3. Demonstration of Technology: CTR

3.1 Background

Most car navigation systems estimate the car position using satellite-based positioning systems, such as the Global Positioning System (GPS). Other satellite-based systems are available, such as the GLONASS system, or the upcoming Galileo System, though such systems are not currently offering worldwide coverage. Figure 3.1 depicts the GPS satellites in orbit.

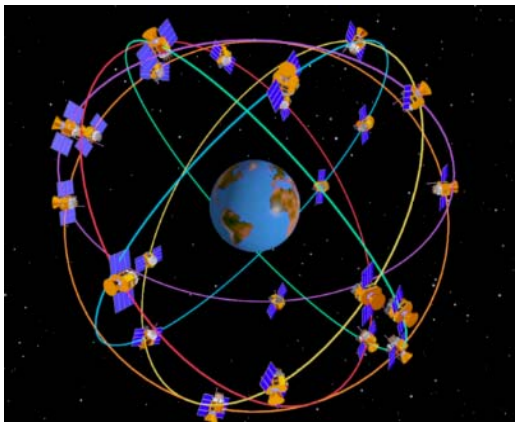

Figure 3.1: Illustration of a Constellation of GPS Satellites Orbiting the Earth

GPS positioning systems operate as follows: a set of satellites transmit pulses at regular time intervals, which can be received by a GPS receiver. If four satellites signals are simultaneously received by the user, the user can determine its position and time by solving a system of four equations with four unknowns, where the equations correspond to the times required for the signals of the satellites to reach the GPS receiver, and the unknowns correspond to the position on Earth (longitude, latitude and altitude), as well as the current time.

GPS satellites operate on two communication channels (L1 and L2), operating at a very high frequency, on the order of 1GHz. The L1 channel carries the navigation message, which is transmitted at a very slow rate of 50 bits per second (bps, or Baud). It is a 1500-bit sequence, and therefore takes 30 seconds to transmit. This navigation message includes information on the broadcast ephemeris (satellite orbital parameters), satellite clock corrections, almanac data (a crude ephemeris for all satellites), ionosphere information (which is used to correct the delays received by the receiver in function of the state of the ionosphere, an atmospheric layer located between 60km and 1000km altitude), and satellite health status.

While GPS systems are relatively inexpensive and accurate (up to tens of meters in usual conditions), they have several drawbacks for traffic sensing applications:

1) The positioning information is affected by random noise, particularly in urban environments (canyon effect). This random noise is caused by the unwanted reflections of the satellite signals on buildings, which affect the accuracy at which one can precisely time when the signal of each satellite was received, and therefore causes positional errors. In urban environments, these errors can be on the order of tens of meters, which can cause, for example, a vehicle equipped with a GPS to appear to be driving on another street. This can result in a loss of precision for traffic purposes: while the mapping of the vehicle to the road network is usually correct, it may be that the fluctuations in the estimated position (from the GPS measurements) cause high uncertainty in travel time estimates between two consecutive intersections. Similarly, when monitoring traffic in urban environments, the GPS uncertainty prevents one from accurately distinguishing vehicles stopped in traffic, or parked vehicles (such as vehicles waiting for a passenger). Higher-resolution GPS systems are available; for example, Real-Time Kinematics (RTK) GPSs use measurements of the phase of the GPS signals emitted by satellites to pinpoint the position of a receiver with greater accuracy. As of 2016, however, these devices cost hundreds of dollars, and require minutes to tens of minutes to properly lock on GPS satellites.
2) While the generation of absolute position data (longitude, latitude, and altitude) is desirable from a positioning standpoint, it inherently affects the privacy of the user when part of a traffic monitoring system. Indeed, classical traffic monitoring architectures (such as the architecture used in the Mobile Millennium experiment at UC Berkeley) for GPS-based traffic sensing rely on GPS position measurements sent by users to a given centralized server, as illustrated in Figure 3.2. As this figure demonstrates, the traffic data generated by vehicles (in the form of velocity and position measurements) is first sent to a third-party server (over the cellular network), which attempts to anonymize the data (for example, by stripping the phone number associated with the GPS position information), and subsequently transmits this data to computer servers that perform the traffic state estimation (using possibly other traffic feeds, such as from fixed traffic sensors or other sources of traffic information). The major issue associated with this architecture is that the third party has access to all information about the user, and therefore has to be trusted.

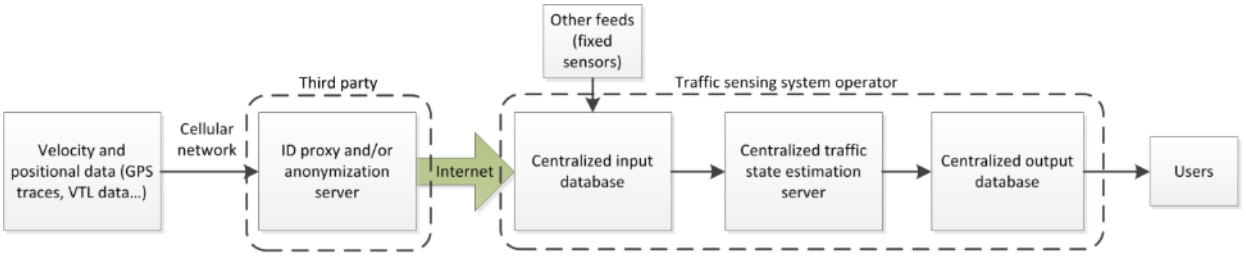

Figure 3.2: Architecture of Classical Traffic Monitoring Systems (probe-vehicle based). In this system, traffic measurement data is sent to an anonymization server, which holds sensitive information.

Figure 3.3 outlines a different type of traffic monitoring architecture based on a short-range wireless radio network. In this architecture, the data generated by vehicles is processed in a distributed manner by the fixed radio nodes themselves.

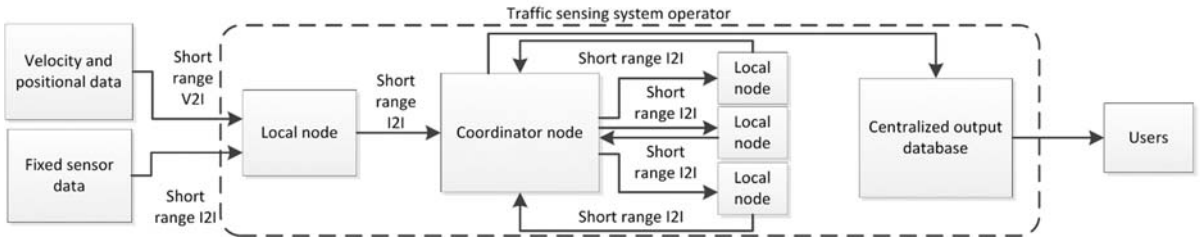

Figure 3.3: Architecture of a Distributed Probe-Based Traffic Flow Monitoring System that Guarantees User Privacy.

The advantage of such a system is that privacy is guaranteed by design, since only a distributed attack on the radio nodes would allow an adversary to gain information on the location of users. A similar concept is reflected in the time-varying IDs attributed to CVs; these IDs make tracking a CV more complex, since an adversary has to compromise all nodes along the path of the CV to track it.

Bluetooth or WiFi readers are widely used across the United States and the world to generate traffic measurements. They operate as follows. A vehicle carrying a Bluetooth or WiFi-enabled device (for example, a Bluetooth-enabled cellphone, or a WiFi-enabled tablet) drives between two different readers. Each reader captures the MAC (Medium Access Control) of the device by performing a scan. The MAC is unique to each device. Therefore, the operator of the sensing infrastructure can match the MAC addresses collected by the readers, and determine the travel time required to go between one reader to the other.

The main issue associated with Bluetooth or WiFi readers is their inherent tradeoffs. The devices cannot be installed too far apart from each other, as the probability of matching vehicles decreases when the distance between readers increases (since vehicles are less and less likely to take the route between the two readers). A notable exception is highways, since most users can only take one route between two readers. Similarly, Bluetooth or WiFi readers cannot be installed too closely from each other, as this would result in added uncertainty, due to the detection range of the Bluetooth or WiFi signals, in the order of tens of meters.

Thus, the proposed IMU system can interface directly with Bluetooth readers, providing an additional and complementary data feed to this system

3.2 Inertial Measurement Units

An Inertial Measurement Unit, or IMU, consists of the combination of an accelerometer, a gyrometer (or gyroscope), and possibly a magnetometer, in a single device. IMUs are commonly used in aerospace engineering to estimate the position of aircrafts or spacecrafts, by monitoring the accelerations and rotations of the vehicle in which the IMU is located. IMUs are also used in connected and autonomous vehicles to monitor their acceleration and attitude with respect to the ground.

The accelerometer of an IMU measures the proper acceleration that is the acceleration of an object with respect to a free-falling frame. The proper acceleration (sometimes referred to as *g-force*) is different from the actual acceleration of the object (sometimes called *coordinate accelerations*). In this project, we are not interested in matching the accelerations to causes (external forces), since we only want to reconstruct vehicle trajectories

The gyrometer (or gyroscope) of an IMU measures the rate of rotation of an object with respect to an inertial frame. Newtonian mechanics postulate that all inertial frames are in uniform translation with respect to each other, and therefore have no rotation motion with respect to one another. Such frames are approximated by frames that use reference points that are very far away from us (for example, stars or galaxies). The gyroscopes measure the rate of rotation of an object with respect to these frames, by measuring the Coriolis pseudo-force caused by the rotation on a test object.

The magnetometer is a device that monitors the direction and amplitude of the local magnetic field, and can therefore be used as a directional reference by tracking the direction of the magnetic North. Given that vehicles are built with high amounts of steel, which is ferromagnetic (and thus strongly perturbs magnetic field lines), the measurements of the magnetometer are in practice too unreliable to be used as a directional reference.

3.3 Fabrication of a Bluetooth IMU Device

To facilitate the integration of an IMU with a vehicle, we choose to build our own IMU system using hardware components, integrated in a Printed Circuit Board (PCB). The objective was initially to use the IEEE 802.15.4 RF protocol to transmit position data to a given wireless sensor network, and as a result, the system has a slot for an 802.15.4 XBee transceiver. An early prototype of the system is shown in Figure 3.4:

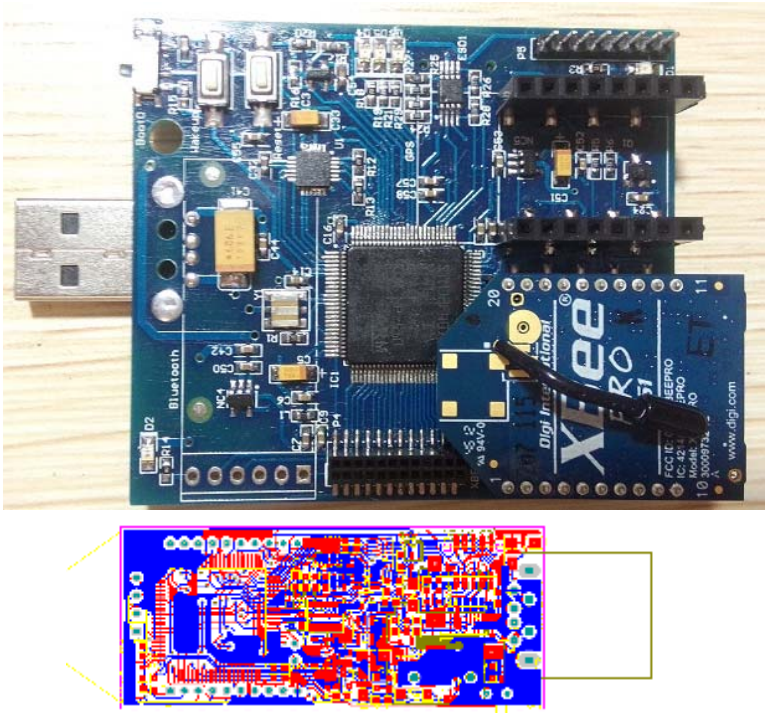

Figure 3.4: Top: Early IMU Prototype. Bottom: Second Iteration of the PCB layout.

The final iteration of the IMU prototype is shown in Figure 3.5. This figure also shows the different peripherals connected to the main processor.

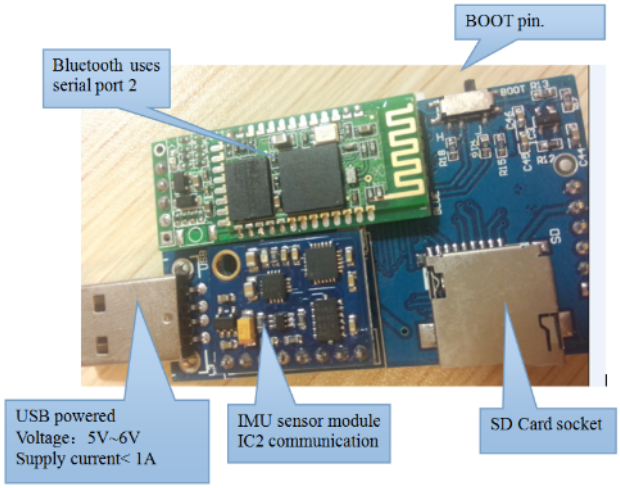

Figure 3.5: Bluetooth, IMU, and SD Card Peripherals of the Developed Sensor

To validate the performance of the sensor in trajectory reconstruction, we also included a GPS system (which is only used for validation). The GPS has its own antenna, and also communicates to the main processor using serial communication. It is shown in Figure 3.6.

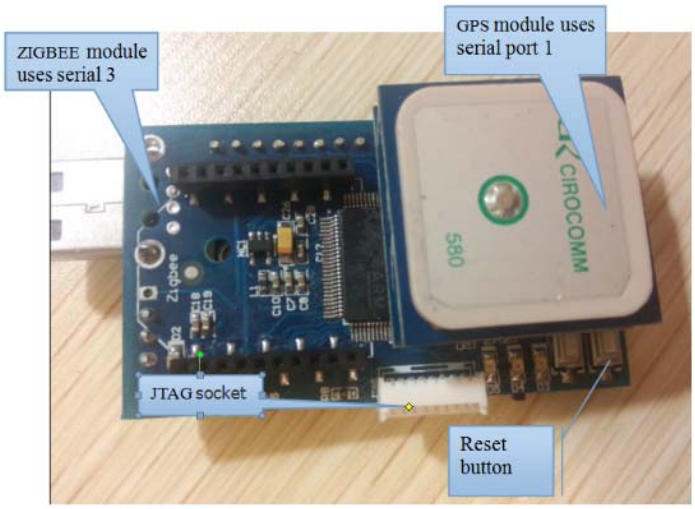

Figure 3.6: GPS and JTAG Programming Interface of the Sensor

To program this sensor, we use a JTAG (Joint Test Action Group, which is an electronics industry association formed in 1985 for developing a method of verifying designs and testing printed circuit boards) interface. JTAG interfaces are commonly used for developing printed circuit boards, and have standard connectors for programming the device. The JTAG interface and JLink programmer used to upload the code to the memory of the microcontroller is shown in Figure 3.7.

Figure 3.7: JTAG Programming System. Left: RS232 Interface. Right: JLink Programmer

3.4 Validation of the Different Components

The second set of activities consists in developing software to interface with the sensor, and communicate with its different subsystems (for example, Bluetooth, GPS, and IMU). This requires the development of software libraries. These libraries allow the microcontroller to establish a connection with its peripherals, retrieve the data they generate (for the GPS and IMU), configure their performance characteristics (for example, the rate at which they send measurement data or their scales), and outputs this data (such as by sending them to a Bluetooth-enabled device, or by storing them in a micro SD card).

Since embedded systems have an emphasis on performance and low cost (with respect to other consumer electronics), they tend to be unreliable, which requires an intensive debugging process. The different components have subsequently been tested by installing the device in a vehicle, connecting it either to a free USB port, or to a USB car charger. An example of installation is shown in Figure 3.8.

Figure 3.8: IMU Device Installed in a Vehicle, with Power Supplied through a USB Car Charger.

The Bluetooth connectivity was subsequently tested by installing a Bluetooth terminal application (in the present case the BT Simple Terminal app developed for Android) on a smartphone (Samsung Galaxy Mega 2), and pairing with the device. The default pairing code chosen (1234) is static for simplicity, although more sophisticated and secure pairing schemes could be created.

The inertial measurement data consists in a vector with six components:

$$a = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix},$$

which corresponds to the vector of proper acceleration measured in the set of coordinates defined by the IMU sensor.

$$g = \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix},$$

which corresponds to the rotation vector, measured in the set of coordinates defined by the IMU sensor.

Note that the rate at which data is generated by the sensor is a function of the dynamics that we want to track. For land vehicles, the spectrum of the accelerations and rotation rates contains frequencies that are relatively low, on the order of a few Hz. Therefore, we chose a sampling rate of 10Hz, which is sufficiently high to cover all significant frequency components of the signal (by Shannon's sampling theorem). The sampling rate should also be as low as possible,

since the random noise affecting the signal increases with higher sampling rates. Figure 3.9 illustrates the reception of data on a smartphone over Bluetooth.

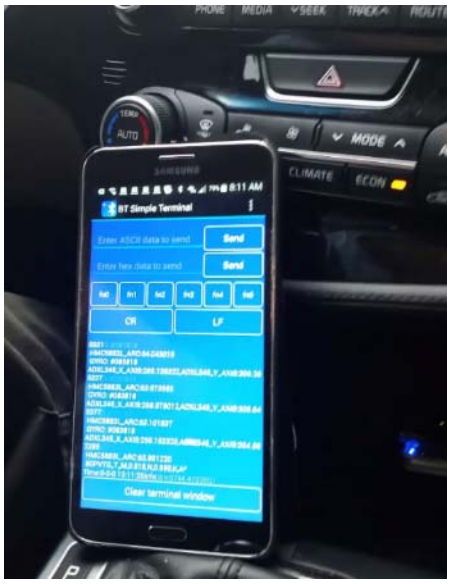

Figure 3.9: Illustration of Inertial Data Reception on a Bluetooth-Enabled Smartphone

3.5 Inertial Data Validation

We conducted some tests to validate the performance of the IMU component of the system, by performing a few checks:

- The norm of the acceleration vector $a = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}$ should be close to the value of g (acceleration of gravity at the surface of the Earth)
- We align each of the three axes of the IMU in the direction of the vertical, to check that each of the axes has a correct acceleration measurement. The variability of the acceleration measurement between axes is caused by factory calibration and accelerometer bias.

The norm of the acceleration vector is shown in Figure 3.10. As this figure illustrates, the norm is very close to the acceleration of gravity g (about 980 cm/s^2), and within the 2% error specified in the IMU parameters. The x axis corresponds to the time sample, over an experiment time of 75 seconds (with 10 measurements per second).

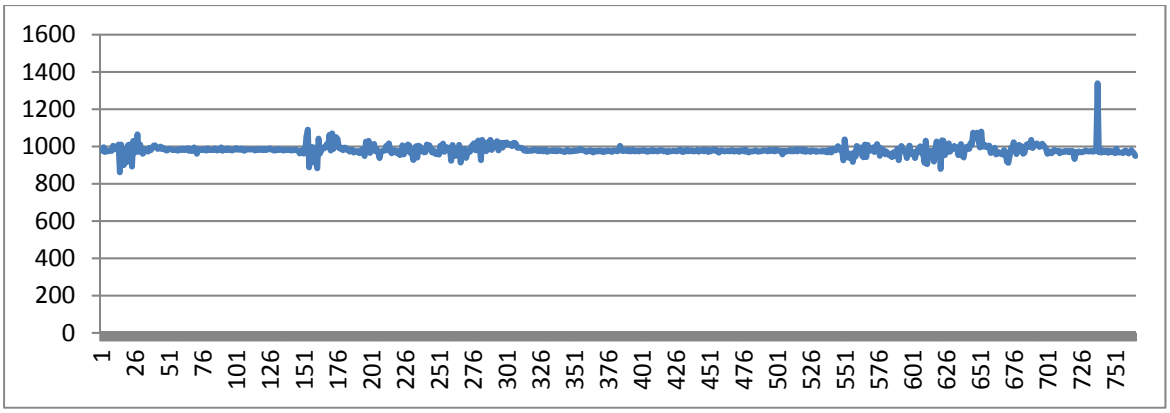

Figure 3.10: Norm of the Acceleration Vector (units: cm/s^2).

As this figure demonstrates, the proper acceleration is always very close to 1000 cm/s^2, which corresponds to the acceleration of gravity on Earth.

Figures 3.11 and 3.12 show examples of raw acceleration and rotation rate measurement data, obtained from the accelerometer and gyrometer. Since these results are obtained in the set of coordinates of the IMU device, they are difficult to interpret.

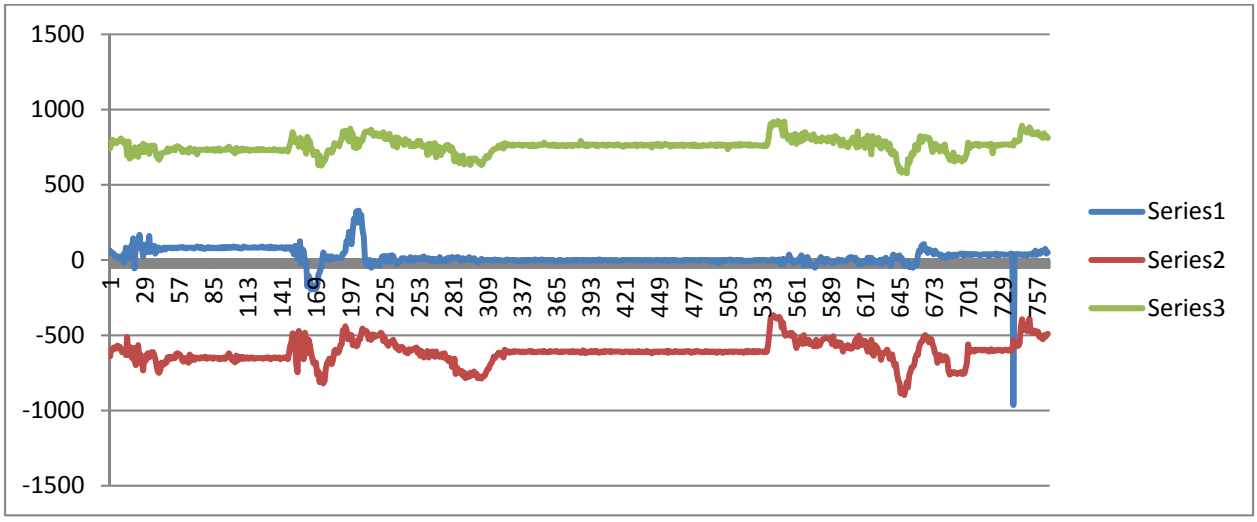

Figure 3.11: Acceleration (units: m/s^2) along the Three Axes of the Accelerometer during a Car Trip.

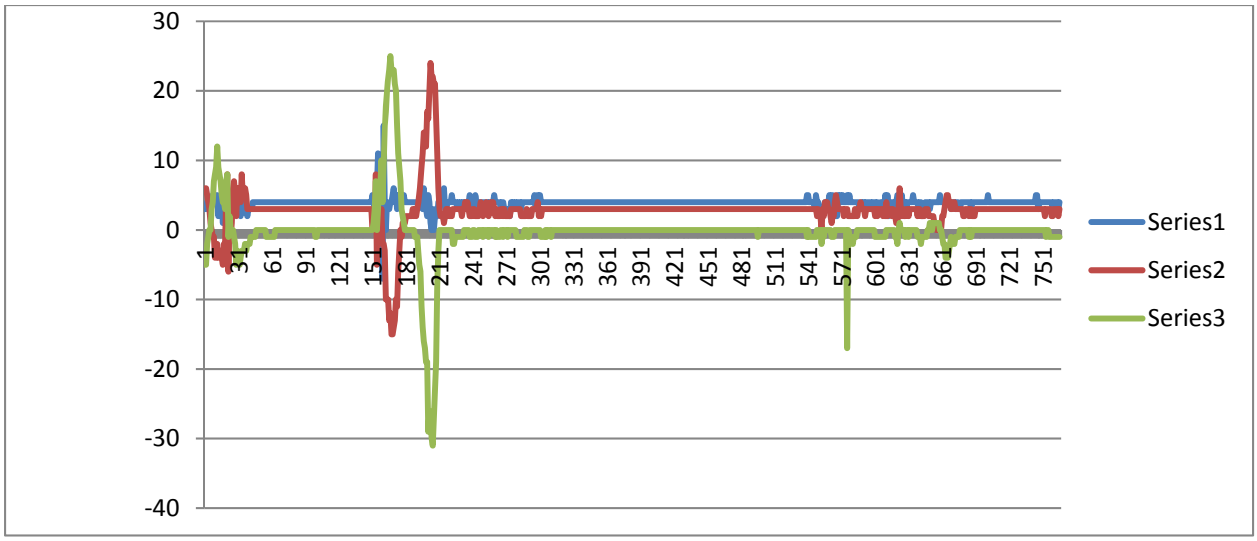

Figure 3.12: Rotation Rate Measurement Data (units of $0.01 \circ / s$)

3.6 GPS Free Auto Calibration of IMU Onboard Vehicles

In Figures 3.11 and 3.12, the IMU is not aligned with the coordinate axes of the vehicles, which are defined as follows:

- Longitudinal axis: x axis
- Lateral axis: y axis
- Vertical axis: z axis

The axes of the IMU are not necessarily aligned with the aforementioned axes. The relationship between the coordinates of the acceleration and rotation rate vectors in the vehicle axes and in the IMU axes is encoded by a rotation matrix R:

$$\begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix} = R \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}$$

where A corresponds to the coordinates of the acceleration vector in the vehicle frame, and a corresponds to the coordinates of the acceleration vector in the IMU frame. This misalignment is one of the main issues of retrofitting an IMU to a vehicle. Unlike a GPS, an IMU requires calibration, which makes the retrofit too complex. We thus investigated a way to perform this calibration automatically.

3.7 Procedure

Let $R_{\underline{\underline{c}}}(t)$, $R_{\underline{\underline{s}}}(t)$, and $R_{\underline{\underline{s}}}(t)$ be the rotation matrices transforming, respectively, the vehicle coordinates into the ground coordinates, the IMU sensor coordinates into the ground coordinates, and the IMU sensor coordinates into the vehicle coordinates. Our objective is to determine $R_{\underline{\underline{s}}}$,

which is assumed here to be constant ($R_{\underline{s}}{}_{\underline{c}}$ is only representing the coordinate change between the IMU and the vehicle, and unless the IMU is rotated with respect to the vehicle, this transformation remains constant). Since we do not have GPS or magnetometer data, $R_{\underline{s}}{}_{\underline{g}}$ (which corresponds to the mapping between the IMU coordinates and the ground coordinates) cannot be determined univocally, though this does not affect the self-calibration principle.

Using the above definitions, we have that $R_{\underline{s}}{}_{\underline{c}}(t) \times R_{\underline{c}}{}_{\underline{g}}(t) = R_{\underline{s}}{}_{\underline{g}}(t)$ (by the composition of rotations). We assume that the pitch and roll attitude of the vehicle with respect to the ground is most of the time zero, given that most of the time, the vehicle is situated flat on the surface of the Earth; that is, the vehicle does not have any roll angle (or tilt with respect to its longitudinal axis) or pitch angle (with respect to its lateral axis).

With this assumption, we have that $R_{\underline{s}}{}_{\underline{g}}(t)$ is (on average) a rotation matrix of a pure yaw, of the form:

$$\begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Determining $R_{\underline{s}}{}_{\underline{g}}(t)$ (up to a rotation with respect to the z axis of the Earth frame) can be done by fusing (combining) the accelerometer and gyrometer data, using a complementary filter or a Kalman filter. Note that since no heading measurement is assumed to be available, this rotation matrix will be known up to a rotation around the vertical direction (z axis of the Earth frame). While the IMU contains a magnetometer, which could be used to obtain the heading of the vehicle, the presence of metal in a car greatly affects the accuracy of the readings of this device, and we chose to ignore its measurement data for the present application. Therefore, the two above equations do not allow us to determine the attitude of the device with respect to the vehicle $R_{\underline{s}}{}_{\underline{c}}(t)$ uniquely. To determine $R_{\underline{s}}{}_{\underline{c}}$ uniquely, we can leverage the residuals of the acceleration measurements. Indeed, the proper acceleration of the vehicle will be (in the frame of the vehicle, neglecting the Coriolis acceleration due to the rotation of the vehicle around its z axis[1]):

$$\begin{bmatrix} a_x = \dfrac{dv}{dt} \\ a_y = v g_z \\ a_z = g \end{bmatrix} = R_{\underline{s}}{}_{\underline{c}} \begin{bmatrix} a_X \\ a_Y \\ a_Z \end{bmatrix}$$

where a_x, a_y, a_z, a_X, a_Y, a_Z, and $g_z$respectively denote the acceleration components in the vehicle coordinates, the acceleration components in the sensor coordinates, the velocity of the vehicle in the Earth frame, and the rotation rate of the vehicle along the z axis in the vehicle coordinates.

[1] The Coriolis acceleration is on the order of $v.\omega$, where ω is the yaw rate of the vehicle and v is the speed of the vehicle (in the Earth frame). For usual vehicle speeds and yaw rates, the effect of the Coriolis acceleration is negligible.

Since the gyro measures the rate of rotation, we use the following approach: if the rate of rotation is approximately zero[2], the second term in the above equation is approximately zero, which gives us an additional measurement constraint, enabling us to compute the rotation matrix $R_{\frac{s}{c}}$.

We validated the performance of this algorithm in reconstructing the correct value of $R_{\frac{s}{c}}$ by computing the acceleration in the Earth frame. The results are shown in Figure 3.13. As this figure demonstrates, the algorithm correctly converges to a state in which both a_x and a_y are zero, as expected.

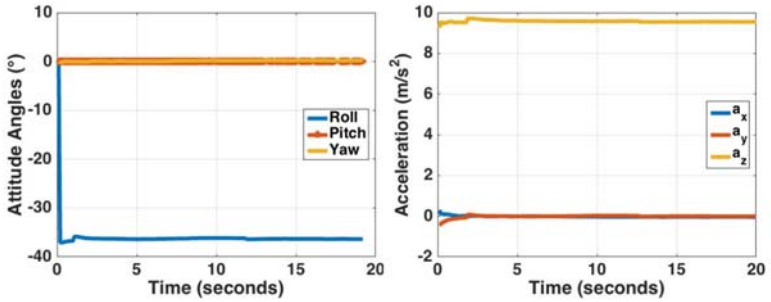

Figure 3.13: Convergence of the Attitude Angle Estimates (attitude of the IMU device with Respect to the Vehicle) Derived from the Rotation Matrix $R_{s/c}$

The results outlined in this section address a major issue in retrofitting classical vehicles into CVs. While radios, computational equipment, and absolute positioning devices (GPS) can quickly be added to a vehicle, IMUs (which are critical for some CV applications, such as hard braking detection or incident detection) are difficult to install, since their axes need to be aligned with the frame of the vehicle, which is a cumbersome process. The automatic calibration procedure shown earlier allows for quick installation of an IMU device (a component of the CV hardware suite) in a vehicle, which paves the way to CV retrofits that could be installed by the drivers themselves, and can significantly accelerate the deployment of CVs on roads.

[2] To check if the rotation rate is approximately zero, we are thresholding for the norm two of the rotation vector in the device frame. The rotation of the Earth is negligible with respect to the measurement uncertainty of the gyrometer, and thus we can assume that a fixed object on Earth is associated with a zero rotation vector.

Additional Files Related to This Product:

Demonstration Videos and Transceiver Data from the SwRI and CTR Research Teams

Demo Videos
Following are links to the video files for each of the CV applications demonstrated during Phase 1.

- Emergency Vehicle Alert (CV_EVA.mov):
https://drive.google.com/file/d/0BxcDb0raOPUgZzlJbWhtaXBEWXM/view
- Electronic Emergency Brakes Lights (CV_EEBL.mov):
https://drive.google.com/file/d/0BxcDb0raOPUgNFVnQ3pqanJId28/view
- Wrong-way Driving Detection (CV_WWD.mov):
https://drive.google.com/file/d/0BxcDb0raOPUgdmxoMV9WSzNGcWM/view
- Intelligent Message Propagation (PropagationOn410_01.mp4):
https://drive.google.com/file/d/0BxcDb0raOPUgWWhXTU1XSU5pVVk/view
- Road Condition Monitoring (RCM Demo.mov):
https://drive.google.com/file/d/0BxcDb0raOPUgV2pqXzQzTWgyU0E/view
- Dynamic Lane Learning (Lane_Learning_Track_Intersection (Converted).mov):
https://drive.google.com/file/d/0BxcDb0raOPUgTXJjcHlBem9zMkU/view

Transceiver Data
The provided folder of three standard Matlab files (Trip_013.mat, Trip_014.mat, Trip_015.mat) contains the following matrices.

1. ACC, corresponding to the raw acceleration data in the sensor frame, in standard gravity units (g). This data can be converted in m/s^2 by multiplying it with 0.0098. This data contains three columns, and each row corresponds to the time step since the beginning of the experiment. The time difference between each row corresponds to the value stored in the variable TT.
2. GYR, corresponding to the raw rotation rate data in the sensor frame, in $°/s$. This data contains three columns, and each row corresponds to the time step since the beginning of the experiment. The time difference between each row corresponds to the value stored in the variable TT.
3. GPS, corresponding to the latitude and longitude data, multiplied by 100. This data can be plotted in Matlab using the geoshow command, applied to the matrix GPS/100 (pointwise division of the matrix GPS by the factor 100).
4. TT, corresponding to the time step associated with the experiment.

These data files were obtained by pairing a laptop computer to the IMU/GPS device developed as part of Phase 1 of Project 0-6838, and recording the output of the IMU device.