Wireless Vehicular Blind-Spot Monitoring Method and System

Progress Report

Department of Electrical and Computer Engineering
University of Manitoba

Prepared by:
Chen Liu
Xiaodong Xu

Faculty Supervisor:
Prof. Robert Mcleod

Group Number: Group 6

Course Number: ECE4600
Submission Date: Jan 14, 2013
1 Introduction

The vehicle blind-spot monitoring system we proposed is a novel Smartphone-based platform to help drivers detect objects within a predefined perimeter of the vehicle. The purpose is to detect and monitor objects encroaching upon a vehicle’s blind-spot area and warn the driver if a potential collision is detected. This blind-spot monitoring system is designed to be used with vehicles that do not have a factory-equipped blind-spot detection system while leveraging the Smartphone as the user interface. Therefore, it can be easily installed on any type of car to enhance safety. During this reporting period, we have finished the initial prototype of the entire system, which means our project is ready for initial testing. Our progress is aligned with our original projections, but it is slightly behind schedule (reason will be explained in Progress up-to-date section). We need to speed up our work in order to make up some time in the next two months to meet the original schedule.

2 Progress Summary
This section summarizes progress to date and decisions made during the development period.

2.1 Decisions

2.1.1 Display and Control Unit

We decided to adopt the modern Smartphone as the display for the monitoring and warning system of the project. Since the Smartphone is widely used nowadays, there are three benefits to using it as our display: 1) The Smartphone’s powerful graphical display functions can facilitate and enhance the blind-spot area monitoring. 2) The Smartphone can provide powerful processor and comprehensive application programming interface (API) to support the algorithm development. 3) The cost of the project is reduced substantially resulting from the fact project does not need to include a specific display and a central control unit.

For the choice of Smartphone, we selected iPhone 4S as our target platform for two primary reasons: 1) Due to budget limitations, we were restricted to use our available Smartphones for the project. In this case, iPhone 4S and Samsung Galaxy Nexus are only options we have. 2) As the project requires the Smartphone to support Bluetooth Low Energy (the reason
for choosing Bluetooth Low Energy will be explained in Wireless Technology section), iPhone 4S was selected as our only choice for now.

2.1.2 Wireless Technology

As our project utilizes the wireless communication between the sensor unit and the control/display unit, the choice of the wireless technology becomes an important decision to make.

For a project based on wireless sensor, the most popular wireless technologies are Bluetooth, IEEE 802.11(Wi-Fi) and ZigBee. Among these options, only the ones that are supported by Smartphone can be considered, since the Smartphone should be able to transmit/receive the wireless signal independently. Therefore, we ruled out the option of ZigBee because it’s not built-in to any Smartphone at present. In addition, as our blind-spot detection system is not embedded into the vehicle, an external power source, such as a battery, is required. In order to achieve the goal for less frequent battery replacement, the power consumption of the system is a critical issue. Another fact is that the communication does not require a high bandwidth as we are only transmitting simple data via the connection. Based on these criteria we decided to choose Bluetooth over IEEE 802.11(Wi-Fi) as it consumes less power than IEEE 802.11 while providing us enough bandwidth for data transmission. In addition, we found that the Bluetooth Standard 4.0 includes a Low Energy profile, which is much more power efficient (it can run for years with a coin cell battery) than Bluetooth v.2.1 [1]. Hence, we decided to utilize Bluetooth Low Energy (BLE) to implement the wireless communication.

2.1.3 Distance Sensor

According to our research findings, there are several sensors being used to measure distance to an object. The most common object distance sensors are infrared sensors, microwave sensors, radar sensors, and ultrasonic sensors. The infrared sensor performance will be interfered with direct and indirect sunlight. In addition, depending on the object’s surface, color and shade the reading may be different [2]. Hence, the inaccurate distance measurement of infrared sensor cannot meet the requirement for conveying correct information to a driver-operator to ensure safe vehicle operation. For the microwave sensor, the sensor may detect undesirable movements due to its strong penetration [3]. These detected undesirable movements will interfere with blind spot monitoring results and may trigger false alarms. The radar sensor is far more expensive compared to other methods, and was not selected as this violates our design objective of low cost.
Compared to other options, the ultrasonic sensor best matches the requirements for our project due to the following reasons: 1) Discrete distances to moving objects can be detected and measured. 2) It is less affected by target materials and surfaces, and it is not affected by color and sunlight. 3) It has resistance to external disturbances such as vibration, infrared radiation, ambient noise, and EMI radiation [4]. 4) Ultrasonic transducers are also of low cost and are widely available. Therefore, it was decided to utilize ultrasonic sensor as our distance sensor.

2.2 Progress up-to-date

2.2.1 Software on iPhone – status completed
In order to design and implement a graphical user interface (GUI) on iOS for the monitoring and warning system, Xiaodong Xu learned Objective C and iOS programming. During this time, we realized that Objective C is a very different programming language in regards to its syntax compared to C and C++. This resulted in more time being spent on getting familiar with Objective C. This is one of reasons why our progress is behind schedule. Furthermore, the Bluetooth Low Energy is a new technology on Smartphone so that the related documentation is lacking. To address this documentation problem, Xiaodong first looked into the related Apple iOS API libraries to gain a general understanding for how to use Bluetooth Low Energy in iOS programming, meanwhile, Xiaodong asked his supervisor for help in order to get more references. Based on his practice and exploration, he figured out a solution to establish a wireless communication through Bluetooth Low Energy on the iOS platform. In addition, he built an application, with a simple user interface, which implements our collision prediction algorithm.

2.2.2 Real-time Objects Approaching Algorithm Development – in progress
For this project, designing a reliable collision prediction algorithm is very important. As we know, modeling a driver’s driving behavior while changing lanes can be difficult. A master’s thesis done by Shannon Hetrick, which examined some different side collision warning systems based on 5 different methods. We found that the best way to avoid a side collision is to activate its turning signal while changing lanes. This can help driver to avoid most of the side collisions. And the second best way to warn the driver is the Time-to-Line Crossing rule, basically it estimates the time to collision based on the lateral velocity of the vehicle and then warns the driver when the time-to-collision drops below a certain threshold [5].
After we studied Shannon’s research, we agreed the Time-to-Line Crossing rule can be a good algorithm for our project. We decided to use this algorithm for predicting whether a potential collision could happen. In our project, if any potential collision is detected, an audio alarm will be triggered in order to alert the driver. At present, we have built our algorithm for the project and start testing it based on our prototype system.

During this reporting period, the algorithm has been simulated using a simple Matlab program. For the next reporting period, the iOS application, which implements the algorithm, will be used for the field test.

2.2.3 Sensor Box Initiate Integration – status completed

We wanted to test the performance of ultrasonic sensor on a real car and the reliability of our algorithm as soon as possible, in case any modification is needed afterwards. We decided to build a prototype on an Arduino development board (an open source microcontroller development board) first, as it is easier to interface with the ultrasonic sensor. Moreover, a plug-and-play Bluetooth Low Energy Module is available for Arduino so that we can easily utilize the Bluetooth function for our testing. In addition, the iPhone application created for this task can be reused later, as the application should be able to interface with any Bluetooth Low Energy Module. However, building the Arduino based prototype is not in our original plan so that we revise our Gantt chart to include this task.

To build the prototype, Chen Liu interfaced the ultrasonic sensor with Arduino Board. He then programmed the board to read in the input from our ultrasonic sensor and show the measurements in human-readable format. The test result verified that the ultrasonic sensor is capable of detecting the presence and proximity of objects accurately.

To establish a Bluetooth connection between the sensor prototype and the Smartphone, we used a driver program provided by the Bluetooth Low Energy Module Manufacturer. As a result, our iOS application established the connection with the Arduino system successfully.

At this point, the sensor box prototype is completed.

2.3 Problem

For the performance of ultrasonic sensor, initially we were concerned that the Doppler effect may affect the object distance measurement, as two cars can move towards each other when changing lanes. In order to solve this problem, first of all, we calculated how much deviation could be caused by the Doppler effect. The calculation result indicated the effect on
our project was minimal and could be neglected. Subsequently, we did a field test, which confirmed our calculation was correct.

Another problem was that initially we couldn’t find a way to measure the lateral speed of the vehicle. We had a discussion about this and conducted research about the vehicle lane-changing model. We found that the lane-changing behavior can be modeled as a uniform motion after the initial acceleration period. Therefore we decided to find the difference between each measured distance first, which is the distance we moved between each sample, then divide it by the time interval between samples. This result is the instantaneous velocity of the vehicle. Then we can find the moving average value of the instantaneous velocities, and this will give us a velocity that closes to the uniform motion velocity.

The method of mounting our sensor box to the car is a critical problem that has not been solved so far. We planned to ask some professionals within the Engineering department. In addition, we will investigate the mounting method that used by current similar car accessories.

### 3 Future work

1. Conduct more tests on Arduino based prototype in order to adjust our detection algorithm.
2. Interface the connectBlue Bluetooth Lower Energy kit with ultrasonic sensor.
3. Test the new detection system against iPhone application.
4. Custom PCB board if time permitted.
Reference


## Appendix A – Parts Ordered and Received

<table>
<thead>
<tr>
<th>Part Name</th>
<th>Distributor</th>
<th>Status</th>
<th>Expenditure (CAD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CB-OLP425I-26-A</td>
<td>Digi-Key</td>
<td>Received</td>
<td>59.97</td>
</tr>
<tr>
<td>X-Band Motion Detector</td>
<td>Digi-Key</td>
<td>Received</td>
<td>34.99</td>
</tr>
<tr>
<td>Bluetooth Low Energy Shield</td>
<td>Seeedstudio.com</td>
<td>Received</td>
<td>35.00</td>
</tr>
</tbody>
</table>