Bus Rapid Transit Technologies: A Virtual Mirror for Eliminating Vehicle Blind Zones

Final Report
Volume II

Michael Knoll Sergi
Intelligent Vehicles Lab
Department of Mechanical Engineering
University of Minnesota

CTS 04-12

Bus Rapid Transit Technologies: A Virtual Mirror for Eliminating Vehicle Blind Zones

Final Report
Volume II

Prepared by:
Michael Knoll Sergi
Intelligent Vehicles Lab
Department of Mechanical Engineering
University of Minnesota

December 2003

Center for Transportation Studies
University of Minnesota

CTS 04-12

Acknowledgements

The authors wish to acknowledge those who made this program possible. First, the University of Minnesota Center for Transportation Studies, the University of Minnesota Intelligent Transportation Systems Institute, and the US Department of Transportation Research and Special Programs Administration provided much of the financial support for this project. Second, Metro Transit supported this work by providing the "Technobus" research vehicle and much of the computer and electronic equipment found on board. Aaron Isaacs and Steve McLaird of Metro Transit have constantly supported the project and have provided Mn/DOT drivers for many Technobus demonstrations. Lenny Pawelk and his staff at the Heywood Garage have answered all of our technical and operational questions, and have kept the bus fueled, cleaned, and maintained. In two years of service, the bus has yet to let us down. We greatly appreciate a bus we can rely upon.

Thanks are also due to Mn/DOT whose innovative use of DGPS allowed the IV Lab to use the Trimble Virtual Reference Station DGPS system during the development of the lane assist systems on the Technobus. Moreover, Mn/DOT provided traffic support during the "History Channel" filming, resulting in a safe environment under which to document the lane assist system.

Finally, Trimble has supported this research through its provision of an IV Lab mirror site to Mn/DOT's VRS system. VRS provides high performance DGPS operation throughout the Twin Cities Metro area, allowing the IV Lab to demo its system anywhere in the Metro area.

Table of Contents

List of Figures

Executive Summary

The FTA has identified the concept of Bus Rapid Transit as a means to increase the efficiency of transit operations while maintaining transit's proven safety record. According to the FTA website www.fta.dot.gov, "BRT combines the quality of rail transit and the flexibility of buses. It can operate on exclusive transitways, HOV lanes, expressways, or ordinary streets. A BRT system combines *intelligent transportation systems* technology, priority for transit, cleaner and quieter vehicles, rapid and convenient fare collection, and integration with land use policy."

Because of the limited right-of-way available to build new (and possibly dedicated) lanes for BRT operations, the FTA has identified lane assist as an emerging technology which will enable deployment of BRT systems. The premise behind lane assist technology is to increase the safety of BRT vehicles as they operate in the more unique environments, such as narrow lanes. Lane assist technology will allow BRT vehicles to operate at the desired higher operating speeds while maintaining the safety of the passengers, BRT vehicle and the motoring public.

Metro Transit and Mn/DOT at the present time are cooperatively operating a BRT-like capability throughout the Twin Cities metro area. Buses operate in HOV lanes, on specially designated road shoulders (albeit at speeds significantly lower than limits posted for the adjacent highway), and are provided metered ramp by-pass capabilities in certain locations. At the present time, Metro Transit has 118 shoulder miles approved for BRT; approximately 15 to 20 miles of approved shoulder miles are added annually. These shoulders are considered by the FTA to be Lateral Guideways. These BRT like-capabilities, and others, provide the transit passenger faster, more efficient service when compared to traditional transit methods.

Although the bus-only-shoulder policy continues to be a very successful program, emerging driver assistive technology developed at the University of Minnesota can be

used to solve problems associated with the bus only shoulder program. For instance, most of the shoulders on which transit buses operate are no more than 10 feet (3.05 m) wide; a transit bus measures 9.5 feet (2.9 m) across the rear view mirrors. These narrow lanes require that a driver maintain a lateral error of less than one-half foot (0.15 m) to avoid collisions. This is a difficult task under the best conditions, and degrades to impossible during conditions of bad weather, low visibility, high traffic congestion, etc.

In addition to maintaining the desired lane position, a driver also has to merge into traffic when the bus only shoulder area ends or a left exit is required. Although theoretically the bus has the right of way in such a situation, many times the driver has to "fight" for his or her position. This also adds considerable stress to an already difficult task.

The primary objective of this work was to equip a Metro Transit bus with driver assistive technology which will enable a driver of a Metro Transit bus to better guide a transit bus on a narrow shoulder, especially under difficult conditions. This driver assistive technology was optimized for the bus driver. The technology associated with the primary objective will be aimed primarily at the lane keeping and forward collision avoidance tasks. This objective was met, and is the focus of Volume I.

The secondary objective is to investigate the Virtual Mirror as a technique for side collision warning and avoidance for transit applications. The virtual mirror has been implemented using existing geospatial database tools and DGPS as a range sensing device; however, for practical applications, LIDAR or similar ranging sensors will have to be used. A Virtual Mirror which utilizes LIDAR sensors was developed, and is the focus of Volume II.

The third objective will be to develop long term relationships with Metro Transit, the Federal Transit Administration, bus manufacturers, and technology providers to develop and implement strategies to improve transit operations. For instance, improving the ability of a bus driver to merge into and out of traffic is a high priority. Improved bus

guidance technology will make bus only shoulders a viable alternative throughout the country. Progress towards meeting this objective has been made, but considerable effort will have to be expended to make lane assist technology ubiquitious throughout the transit industry.

1. Introduction

1.1 Background

Mirrors are used to assist drivers in making critical maneuvering decisions by expanding the available field of view, thereby allowing the driver to maintain concentration on the road. Several types of mirrors are commercially available, including the standard planar mirror found on all vehicles in the United States, convex mirrors, and other non-planar mirrors. The National Highway Traffic Safety Administration (NHTSA) has created a set of federal standards concerning the type, properties, and location of mirrors for various types of vehicles in the United States [1]. For example, passenger cars must have a mirror of unit magnification (planar) as the inside and driver-side rearview mirrors. Depending on the field of view provided by these mirrors, a passenger side mirror may be optional and can be either planar or convex. Buses and other heavy vehicles must also comply with the standards defined in [1].

While mirrors offer the driver greater visibility of the current surroundings, there are limitations to their use. Planar mirrors provide a relatively small field of view that creates blind zones, areas around the vehicle in which the driver cannot see when using the mirror. These blind zones create the need for the driver to check the mirrors often to determine if a vehicle has entered a blind zone or to turn their head to directly view these areas while driving, increasing the potential for accidents. Non-planar mirrors offer a larger field of view to the driver, but the image is smaller and often distorted, which can lead to the overestimation of distance even when the driver has experience using non-planar mirrors.

Precipitation and darkness create low visibility situations that reduce the effectiveness of mirrors. In these situations, optical mirrors are either less effective or rendered useless, since the driver cannot see road markings or other vehicles. Additionally, glare from other light sources, such as vehicle headlights, can cause discomfort and greatly reduce the usefulness of mirrors [2][3]. Prism and electrochromic mirrors attempt to reduce glare

without further reducing the usefulness of the mirror, but these devices are not always effective [3][4].

There are also issues associated with optical mirrors that do not directly affect the driver. Cresswell and Hertz [5] studied the affects of mirrors on aerodynamic drag of trucks and found that using standard, commercially available truck mirrors caused a 5-10% increase in drag for a typical truck, which translates into a significant increase in fuel consumption over time.

Potential solutions have been examined to address the limitations of standard mirrors that involved the use of a periscope and mirrors or a rear-looking video camera. Pilhall [6] discussed the use of a periscope with a set of mirrors located on the roof of the vehicle. This provided the driver with a wide, unobstructed view, but due to the large size of the mirrors, it suffered from a large increase in the aerodynamic drag of the vehicle. The paper also described the use of a wide-angle, backwards-looking camera placed on the rear of the vehicle with a monitor located inside the vehicle. This also provides an unobstructed view for the driver, but to maintain an undistorted view would require a very large display in order to cover a wide field of view. These systems also suffer in low visibility conditions as they were designed only to address blind-zone problems.

The virtual mirror, initially proposed in Garlich-Miller and Donath [7] with respect to sensor fusion for object shape recovery and later developed in Pardhy *et. al* [8], is a computer-generated display that addresses the limitations of standard optical mirrors. It duplicates the useful properties of a physical mirror and draws the view on a display panel located inside of the vehicle.

As the display does not rely on the physical reflectivity of a mirror, glare from other light sources does not exist. The virtual mirror uses data from position and range detection sensors to draw the display, thus overcoming other limitations present with optical mirrors. Low visibility conditions do not affect the display nor do physical objects that block the line of sight of the driver. For example, mirrors typically have portions of the

vehicle in the reflected image as seen by the driver. Side rearview mirrors often have the rear corner of the vehicle visible to use as a visual reference, and inside rearview mirrors have a portion of the view blocked by the rear of the vehicle. Since the computer controls how everything in the virtual mirror is rendered, the host vehicle can be drawn semi-transparent so that objects and road markings can be seen in the mirror when they otherwise would not, and the vehicle would still be partially visible to be used as a reference. Therefore, in both normal and low visibility driving conditions, the virtual mirror would provide more information to the driver than a conventional optical mirror. In the event of a severe or total loss of visibility outside of the vehicle, the virtual mirror, combined with a heads-up-display [9], could allow a person to continue to drive with a reasonable amount of safety.

It is important to note that the use of a virtual mirror in place of an optical mirror has considerable operational benefits as well. If the sensor for the virtual mirror protrudes from the side of the bus less than the optical mirror, it in effect makes the bus narrower. Making the bus narrower is equivalent to making a lane wider. This can facilitate bus operations on a narrow lane, and reduce the right-of-way needed for a transit agency to provide operation.

The virtual mirror can be “virtually” moved and re-oriented to view areas that would be impractical for a real mirror. If the driver wishes to eliminate the blind zone along the side of the vehicle, the virtual mirror could be “moved” so that it would be near the front corner of the car. This would provide a much better view of the surrounding, but it would be impossible to place a real mirror in such a location. The various parameters of the virtual mirror, such as the size, can also be adjusted to create the view seen by a very large mirror, impossible to mount on a car, which would provide a wider field of view.

Furthermore, due to the virtual mirror display panel being located inside of the vehicle, there will be a reduction in aerodynamic drag as compared to a vehicle with standard rearview mirrors.

To provide a driver with sufficient data to maneuver safely, information concerning the surrounding environment must be gathered via a set of sensors to be displayed in the virtual mirror. Data pertaining to road markings are based on global positioning system (GPS) surveying and stored in a geo-spatial database [10]. Dynamic objects such as other vehicles must be detected in real-time through the use of range sensors. As the driver relies on the visual perception of the area, cameras may seem like a logical choice. However computer vision algorithms often require powerful hardware to perform in real-time, and suffer from issues with robustness and accuracy. RADAR sensors emit radio-frequency pulses to determine the approximate location and relative velocity of objects based upon the Doppler effect or other physical phenomena. While these sensors are reliable in a variety of environmental conditions, they often have a limited field of view and poor angular resolution. LIght Detection and Ranging (LIDAR) range sensors emit relatively small beams of light that provide an accurate distance measurement and the capability of small angular resolutions. Commercial scanning laser sensors are available that are capable of up to 0.25-degree angular resolution with a distance resolution of several centimeters.

There are three basic types of LIDAR: DIfferential Absorption Lidar (DIAL), Doppler LIDAR, and range finders. DIAL and Doppler LIDAR sensors are commonly used to measure chemical concentrations and wind velocity in the atmosphere, relying on the absorption and reflection of varying light wavelengths. Range sensors typically use the time for the light to travel to the target and back to determine the distance. Our focus is with range finder type LIDAR for this project.

LIDAR sensors have been used for many years as ranging sensors on autonomous mobile robots to provide data on the surrounding environment for obstacle avoidance and mapping [11][12][13]. Using various techniques such as Kalman filtering, iconic matching algorithms, and sensor fusion methods to achieve various tasks, these sensors have proven to be robust and accurate for these goals. In recent years these sensors have begun to be applied to automotive applications. Osugi *et. al* [14] incorporated a laser range sensor into an adaptive cruise control system to eliminate unexpected changes in

acceleration due to sensor errors. They developed a three-dimensional sensor and used a scheme that groups data based upon relative distance, angle, and speed to determine the distance to the forward-most vehicle. Kirchner and Ameling [15] installed a laser sensor on the front bumper of a car for vehicle detection using Kalman filtering and also road boundary detection using a model-based approach. Ewald and Willhoeft [16] developed a scanning laser sensor with a built-in signal processor that detects objects via segmentation and tracks them using a Kalman filter.

An algorithm was developed that uses spatial segmentation to group LIDAR data into clusters. The statistics of these clusters are used to determine the presence and location of nearby vehicles in order to relay this information to the driver via the virtual mirror.

1.2 Report Layout

Thus far the need for and the concept of the virtual mirror has been discussed. The remainder of this report documents the how the system has been implemented and the experiments to analyze the accuracy of the virtual mirror and the vehicle detection subsystem.

Chapter 2 outlines the mathematical methods and transformations needed to generate the display. Sections 2.1 and 2.2 describe the primary transformations necessary to create the 2-dimensional representation of the view seen in the mirror from the 3-dimensional data provided from the sensors and other systems. Section 2.3 goes on to describe the method used to create a stencil effect to mask areas on the display that are outside of the mirror boundaries. While a driver may not require this if they desire that the view encompass the entire display panel, it is necessary to show that the virtual mirror is capable of accurately recreating the view seen by a digital camera during experimentation.

The sensors and systems used to form the virtual mirror system are described in Chapter 3. The database containing the static roadway information, the position and heading sensors, and the range sensors that were used are described in detail in Sections 3.1, 3.2, and 3.3. Section 3.4 explains the rationale for using the OpenGL API to implement the

graphics routines to create the display, and Section 3.5 provides an overall description of the virtual mirror and the hardware used.

In order for the virtual mirror to generate the view, the parameters consisting of the position and orientation of the mirror and viewing location must be known. For the experiments performed a camera was mounted to represent the viewing location, which requires a method to accurately determine these parameters. Chapter 4 contains the information regarding the need for a reliable method of determining these parameters, the equipment configuration used, and the algorithm implemented to search for these parameters.

As previously stated, it is necessary that there be a system to locate nearby vehicles in real-time as this information cannot otherwise be known beforehand. Chapter 5 describes in detail the algorithm used to determine the presence and location of vehicles using a LIDAR range sensor. Sections 5.1 and 5.2 deal with the initial segmentation of the data while Section 5.3 explains how the segmented data is used to determine the location of potential vehicles. Furthermore, a form of vehicle tracking for successive sets of data is need and explained in Section 5.4.

The experiments performed to analyze the performance of the virtual mirror and the LIDAR-based vehicle detection algorithm are detailed in Chapters 6 and 7, respectively. First the equipment is described for each set of experiments, followed by an explanation of component synchronization and compensation for latencies. The chapters end with an explanation of how the analysis was performed on the data and the final results of the experiments.

The report concludes with Chapter 8, which describes the achievements of the virtual mirror in its current state. There is also a discussion of the limitations of the studies performed and of the sensors currently used with the system, with proposals for future components and experiments.

2 Geometric Transforms

The virtual mirror computes the appropriate viewing transformations needed to create a mirror-like display that integrates the position and orientation of both the driver's eyes and the mirror relative to the DGPS antenna on the vehicle. Using the DGPS position, the data retrieved from the geo-spatial database, and the information from the sensing systems around the vehicle, the surrounding environment can be rendered in the display using these viewing transformations. The OpenGL graphics library (www.opengl.org) was used to supply a robust set of graphics routines that allowed for hardware acceleration, thus leading to significant reductions in the rendering time of the display. In addition, the system involved less CPU overhead, affording more CPU time for other tasks on the system to run efficiently.

The graphics computations involved in creating the virtual mirror display can be divided into three major areas: the forward view transformations, the mirror transformation, and stenciling.

2.1 Forward View Transformations

The forward view transformations involve the perspective projection, clipping, and mapping to the two-dimension viewport (i.e. the screen). OpenGL provides routines that perform the computations used in the forward view transformations, thus the detailed mathematical equations will not be included. For details of the steps involved in the forward view transformation, see Foley *et. al* [17].

2.2 Mirror View Transformation

The mirror transformation performs the reflection of three-dimensional vertices about the defined mirror surface. Rogers and Adams [18] discuss the theory and mathematics involved in reflection. A reflection in two-dimensional space about an axis is the equivalent of performing a 180-degree three-dimensional rotation about the axis and mapping it back into two-dimensional space. The result, for a pure reflection, is a transformation matrix with a determinant of –1. For the reflection about one of the primary axes, the resulting matrix is the equivalent of the negative scaling in the direction

of the reflection. For example, the reflection across the *x*-axis in an *x*-*y* coordinate system would result in identical *x* coordinates, but the *y* coordinates would be reversed in sign (refer to Figure 1). This analogy can be extended to three-dimensional reflection. The reflection about a plane formed by two of the three primary axes can be accomplished by a negative scale transformation across the mirror plane, shown in Figure 2.

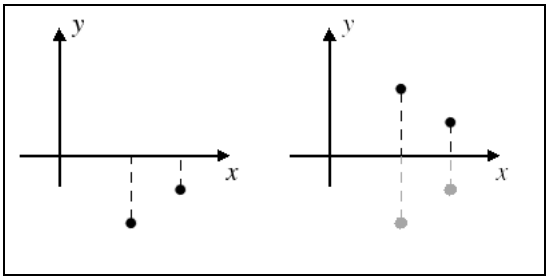

Figure 1 An example of the two-dimensional reflection of two points across the *x*-**axis.**

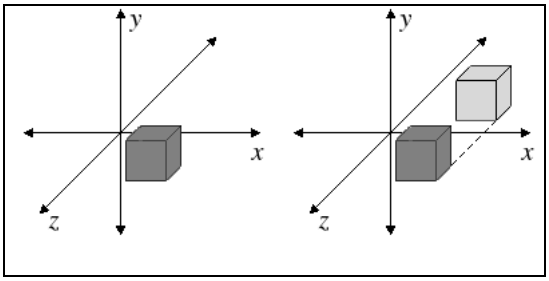

Figure 2 An example of the three-dimensional reflection of a box across the *xy*-plane. The *x* and *y* values remain unchanged while the *z* values change sign.

In order to mimic the view seen in a real mirror, a reflection matrix must be found that will perform the reflection on an arbitrary plane. To accomplish this, we translate and rotate the eye position such that it lies at the origin of a coordinate system defined in the mirror plane, such that the *z*-axis is coincident with the vector normal to this plane. This allows the use of the previously discussed method of scaling by –1 along the *z*-axis to perform the reflection. At this point, the inverse rotation and translation matrices are performed to return the eye to the original position, and the view seen by the observer will be the desired mirror result of the scene.

To compute this reflection matrix, we define z_{scale} to be a transformation that will scale vectors in the negative z-direction and M_{mirror} to be the transformation matrix that maps the mirror into the world coordinate system:

$$z_{scale} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (2.1)$$

$$M_{mirror} = T_{mirror} \cdot R_{mirror}$$

where T_{mirror} and R_{mirror} are the translation and rotation matrices that define the position and orientation of the mirror relative to the DGPS antenna.

Therefore the reflection matrix is defined as:

$$M_{reflection} = M_{mirror} \cdot z_{scale} \cdot M_{mirror}^{-1} \qquad (2.2)$$

The mirror transformation is accomplished by applying the reflection matrix, $M_{reflection}$, before performing the forward view transformation. The end result will be an image such as that shown in Figure 3. A camera image of the same view is shown in Figure 4. The view seen in the virtual mirror rendering is correct, but the lines of the road boundaries extend outside the area of the physical mirror in the view. This is remedied through the use of a stencil.

2.3 Stencil

OpenGL provides a set of stencil routines, however the video hardware used for the virtual mirror does not provide a hardware accelerated stencil buffer. Performing the stencil through software causes the time to draw the display to increase dramatically. Thus a method using the depth buffer was developed to produce the same effect.

The depth buffer is used by OpenGL to store a depth value for each pixel. The values in this buffer are used for hidden surface removal so that only the closest object is displayed at a given pixel location. To create a stencil effect, the depth buffer was filled with very small values for each pixel that did not correspond to a part of the mirror as seen in the view. To accomplish this, the vertices of a set of triangles were generated to distinguish between two sections on the screen: the pixels corresponding to the mirror seen in the view, and the remaining area surrounding the mirror (see Figure 5). The triangles are only added to the depth buffer and not actually rendered so the user will not see them. When an object is drawn, OpenGL will perform a depth test and any part of the object that is "behind" one of these triangles will not be drawn. By using the depth buffer and triangles a fast, hardware-accelerated stencil is created. Figure 6 shows the result of applying the stencil to the virtual mirror image.

One should note that the stencil is not always required. If the borders of the virtual mirror drawn lie outside of the edges of the view, then it would not be necessary. Also the user may choose to disable the stencil in order to simulate a much larger mirror, providing a larger field of view to better assess the surroundings.

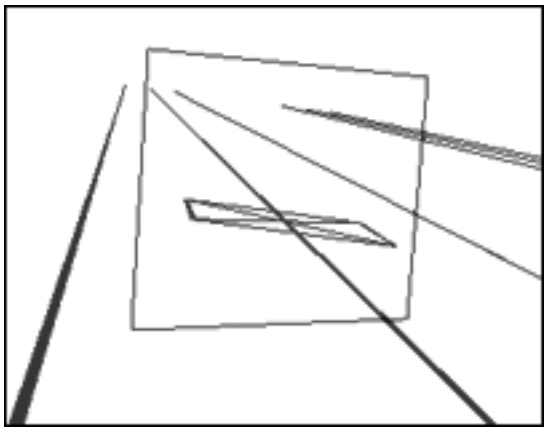

Figure 3 Virtual mirror rendering without using a stencil. The solid lines disappearing in the distance represent real lane markings. The large box in the center represents the edge of the mirror. The box with the X represents a calibration mark that is painted on the road.

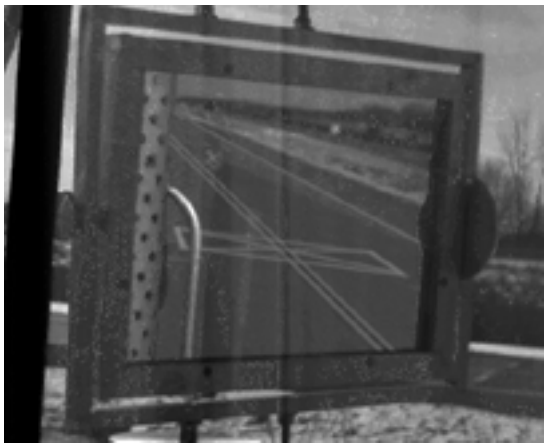

Figure 4 The camera image that corresponds to the virtual mirror image in Figure 3**.**

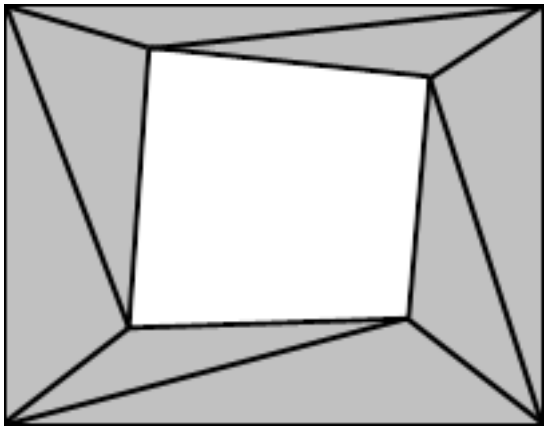

Figure 5 The gray triangles are used to stencil the image. The white section represents the surface of the mirror and anything that lies in that area of the image will be drawn.

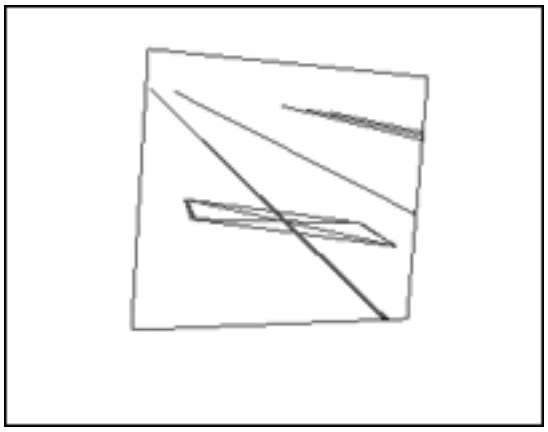

Figure 6 The virtual mirror rendering after using the stencil.

3 Implementation

For the virtual mirror to display the current surroundings at any given time, it must know the location of the vehicle and have some method of acquiring the location and geometry of the surrounding environment in real-time. To accomplish this, a geo-spatial database [10] provides information concerning static objects and markings on the roadway, a DGPS receiver is used to obtain the position and heading of the vehicle, and a scanning laser ranging sensor searches for nearby vehicles. Using this data the display is generated using an OpenGL application.

3.1 Geo-Spatial Database

The virtual mirror must render the view seen in the real world, thus requiring the use of a digital map or database that represents the location and attributes of various objects along the roadway. A database was used that stores road information as spatial objects that correspond to the geometry or geography of the surrounding area. This database is referred to as a geo-spatial database.

3.2 Differential Global Positioning System

The Global Positioning System (GPS) is a navigation system based upon a constellation of 24 satellites. Using radio signals transmitted from these satellites, the position of a GPS receiver can be accurately triangulated to within several meters. For aviation and boat navigation this degree of accuracy is sufficient, whereas vehicle-based applications require more accuracy.

To solve this problem, differential-GPS (DGPS) was created. DGPS involves a stationary GPS receiver that, based upon an accurate surveyed position and recent position calculations, communicates an error correction to other GPS receivers in the vicinity to greatly increase the accuracy of the measurements. For vehicle navigation assistance with driver interfaces such as a virtual mirror, a high-degree of accuracy is needed to ensure safe maneuvering using the display. We use a dual-frequency system based on carrier phase RTK that significantly reduces the effect of ionosphere and troposphere distortion

and wobble in the satellite orbit. The accuracies achieved with this system are on the order of a few centimeters ($1\sigma = 2$cm).

3.3 Light Detection and Ranging

The geo-spatial database is used to display the static surroundings of the vehicle at a given location, but other objects that may pose navigational threats must be detected. Various sensors, such as side-looking Radio Detection and Ranging (RADAR) units, were considered for vehicle detection on the side of a vehicle. Due to the high degree of spatial resolution and accuracy desired, a Light Detection and Ranging (LIDAR) sensor was used.

3.4 Computer Graphics

The previous implementation of the virtual mirror by Pardhy *et. al* [8] utilized a set of custom written graphics routines to render the display in the QNX Photon environment. This provided much flexibility in the customizability of the programming environment, but the time to generate the display was dependant upon the processor speed and other processes running on the same computer. This limited the display to containing only lines and line segments rather than polygons.

Thus the virtual mirror was recreated using an OpenGL library. OpenGL is a widely used 2D and 3D application programming interface (API) that provides a large, robust set of graphics routines as well as the ability to use video cards that allow for hardware-accelerated graphics computations. The end result is a display that is capable of rendering the road and road markings as filled polygons to create a more detailed and intuitive display with very small drawing times.

3.5 System Description

The virtual mirror we have implemented displays the view on liquid crystal display (LCD) panel mounted inside of the vehicle near the driver. An AMD K6-II 400Mhz computer running QNX Real-Time Platform 6.1 with a 3dfx Voodoo3 3000 video display adapter drives the display. The view is rendered based upon the current position and heading received from the DGPS unit. To achieve accurate results, Leica SR530 dual-

frequency GPS receivers were used in the test vehicle and at the DGPS transmitting reference station that results in an accuracy specified by a standard deviation of approximately 2cm. A SICK LMS-221 scanning laser sensor was mounted on the side of the test vehicle and an algorithm for the detection and tracking of vehicles in adjacent lanes using this LIDAR sensor was implemented. Figure 7 illustrates the system layout for the current implementation of the virtual mirror.

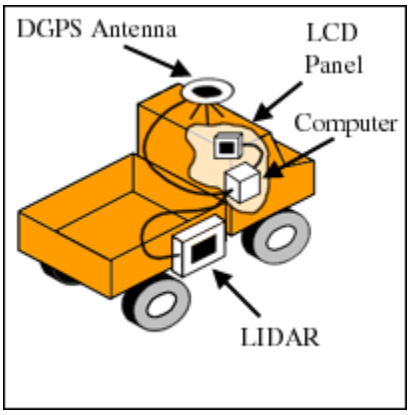

Figure 7 Overview sketch of the equipment set up for the virtual mirror. The computer inside the vehicle gathers data from the LIDAR and DGPS sensors and generates the virtual mirror display on an LCD panel.

The refresh rate of the virtual mirror is currently limited to the 10hz update rate of the DGPS receiver. The time to process the sensor data, query the database, and update the display is well below the 100-millisecond sampling time between DGPS positions.

4 System Calibration

To accurately render the view that would be seen in the real mirror, the position and orientation of both the viewing location and the actual mirror are needed. While these values can be measured, small errors in measurement can lead to non-negligible errors. In a previous implementation, these parameters were determined by trial and error, but this is extremely tedious. It still can be difficult to find the best values using a series of images as each image has been subjected to vehicle vibrations, thus a set of values that results in a perfect match in one image may not be accurate for another image. This requires a "best fit" of the parameters. Thus a method for accurately determining these values was developed.

4.1 Need for Accurate Calibration

A high-accuracy geo-spatial database containing various road features was used for the rendering of road geometry. Using the lane boundary markings and square calibration marks painted on the road, the lines in a camera image and the virtual mirror rendering are compared to determine how close the parameters (position and orientation) are to the actual parameters of the system. Figure 8 provides a layout of the relevant road markings with respect to the vehicle. When the estimated parameters are very close to the actual parameters, then the virtual mirror image will "line up" with the camera image. Figure 9 and Figure 10 show a sample image of the view seen in an optical grade passenger side mirror and the virtual mirror representation respectively. Figure 11 and Figure 12 show the virtual mirror display overlaid on the camera image of the actual mirror. Figure 12 is similar to Figure 11, but in this figure the parameters for each image are identical except the virtual mirror in Figure 12 has a 0.5-degree error in the rotation of mirror along the x-axis. This small error results in a large disparity between the virtual mirror rendering and the real mirror as seen by the camera. This illustrates the need for an accurate method of calibration.

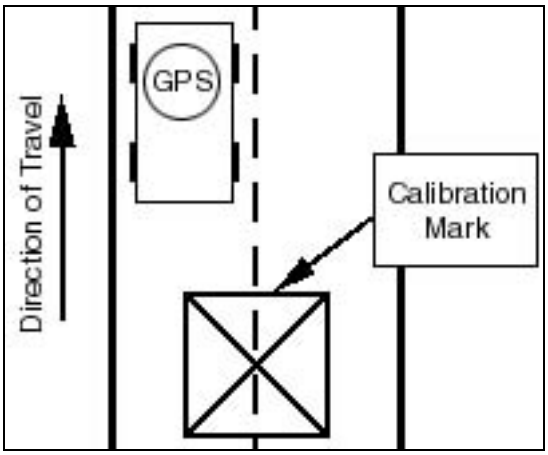

Figure 8 System layout of road markings and a calibration mark with respect to the vehicle whose position is sensed by DGPS.

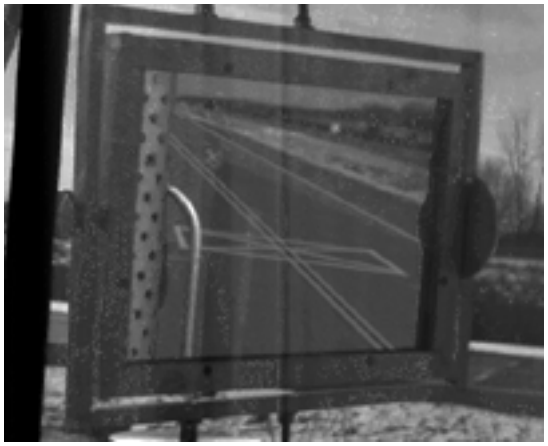

Figure 9 An example calibration image as seen a passenger side mirror.

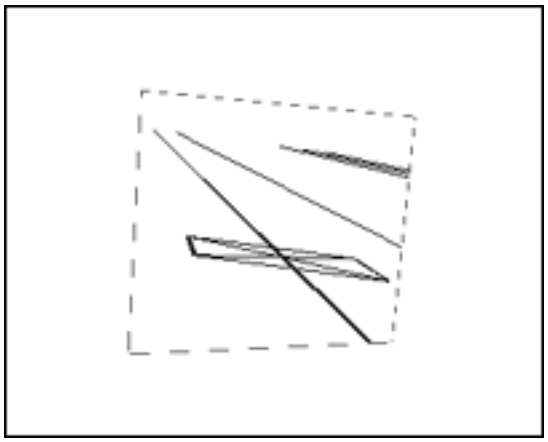

Figure 10 A virtual mirror display representation. This image contains black lines on a white background for ease of viewing, however the normal display renders colored lines (white or yellow).

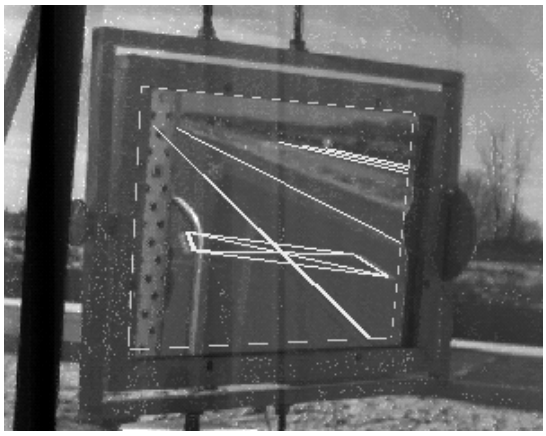

Figure 11 A virtual mirror display superimposed on a camera image. The virtual mirror lines are white in this image.

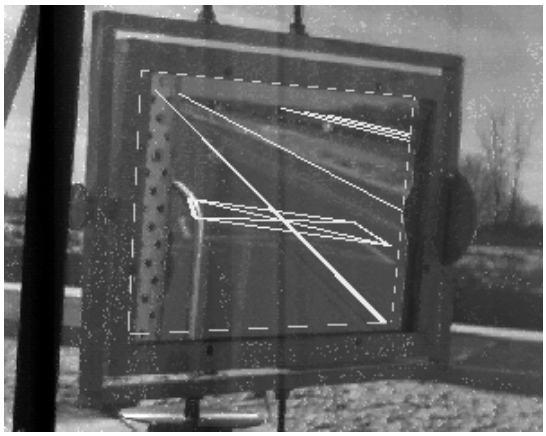

Figure 12 A virtual mirror display superimposed on a camera image in which there is a 0.5-degree error in the x-axis rotation of the mirror. This results in a large difference between the true and virtual images. The virtual mirror lines are white in this image.

4.2 Equipment Setup

During the system calibration, the calibration mark must be in the view of the camera. Two sets of images are used, and are classified as the "forward view" and the "mirror view" (Figure 13). The setup for the two views is identical except that the camera is oriented to look through the windshield for the forward view while the camera is rotated to see the reflection in the right-side mirror in the mirror view. The camera is mounted such that it is able to rotate in only one dimension, resulting in the position and

orientation of the camera for the two views to be identical except for one rotation parameter.

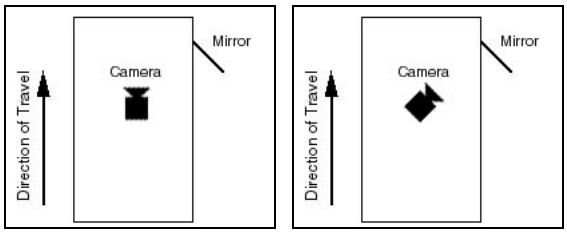

Figure 13 System layout of equipment inside vehicle in the forward view (left) and mirror view (right)

4.3 Data Collection

First a series of images are taken using the forward view at various distances from the calibration mark. The process is then repeated for the "mirror view". The road markings and calibration mark must be seen in image to perform this calibration. An example of a camera image of each view is shown in Figure 14 and Figure 15.

The forward view data is used to determine the camera parameters while the mirror view data is used for the mirror parameters. Since the camera must actually be placed in a different orientation for each view, an extra rotation must be included in the camera parameters. As the camera is mounted such that it can only rotate in one plane, the camera can be rotated about one axis without affecting the position or the other two rotation parameters. Therefore the camera parameters in the mirror view are identical to those in the camera view except that there is an additional rotation along the *z*-axis. The images and data for calibration purposes are collected when the truck is stationary to eliminate the effects of any latency in the system that could introduce error.

Figure 14 A forward-view calibration image

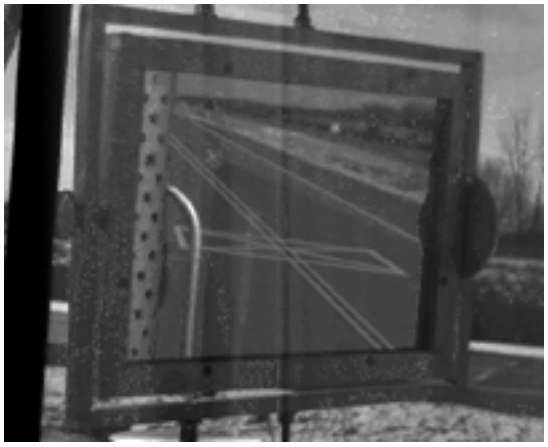

Figure 15 A mirror-view calibration image containing lane boundary lines and a calibration mark.

4.4 Calibration Algorithm

The Post-Processing Extrinsic Parameter System Initialization (PPEPSI) algorithm was developed to determine the desired parameters. Using a set of images and their respective DGPS data (position and orientation of the vehicle) key calibration points in the image are selected, as shown in Figure 16. The algorithm uses the pixel coordinates of the calibration mark corners and the equations of the lines formed by the lane boundaries as a comparison between the camera image and the virtual mirror rendering. The algorithm uses selected points on any visible lane boundaries. On MnROAD, the testing location, three lane boundaries were visible in the forward view. In the mirror only the two lane boundaries for the adjacent lane were visible in the mirror, resulting in two lane boundary

lines and the points for the calibration mark. These points were sufficient to determine the needed system parameters using the PPEPSI algorithm.

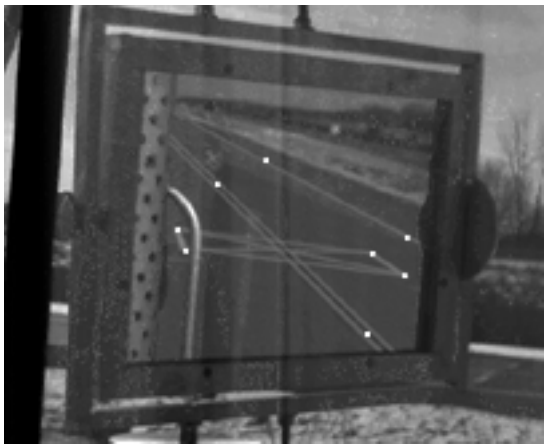

Figure 16 Mirror-view image with white spots representing the 4 corners of the calibration mark and points along the lane boundaries used to determine their line equations.

Beginning with a rough estimate of the position and orientation values, a bounded search is performed where the parameters are adjusted and a measure of the quality of the parameters is calculated. This measure is a weighted-sum of the difference between the equations of the lines formed by the calibration points in the image and the corresponding points rendered by the virtual mirror. Once the range of parameters has been exhausted, the values with the smallest average error measure over the set of calibration images are used to repeat the search using a smaller set of bounds around these values. Using this smaller set of bounds and adjusting the test parameters by smaller increments allows for higher resolution results. After several iterations, the final parameters are stored to use for the final analysis.

5 Vehicle Detection Algorithm

The overall algorithm used for LIDAR-based vehicle detection and tracking is broken into several key steps. Distance thresholds are applied in order to analyze only the data in relevant areas, the remaining data is divided into groups referred to as clusters, the clusters are tested for vehicle matches, and detected vehicles are added to a tracking database.

5.1 Thresholds

Using current LIDAR and DGPS data, some distance thresholds must be applied to remove unnecessary data. First the distance returned for each LIDAR beam is compared to the maximum distance detectable. If nothing was detected, the sensor returns the maximum value and thus that piece of data is ignored. Next the DGPS position is used to compare the sensor location to the road features. The geo-spatial database is queried for nearby road objects such as guardrails, mailboxes, and jersey-barriers that may be detected by the LIDAR sensor. If none of these objects are found to be close to the vehicle, the database is queried for the road shoulder. The perpendicular distance from the location of the LIDAR sensor to each of the mentioned road features is calculated, and the closest distance is used as a threshold. This has the effect of ignoring any sensor data that lies outside of the drivable area of the road where cars would not be present. Figure 17 illustrates an example of a vehicle in a two-lane road where the far edge of the road shoulder is used as the threshold distance.

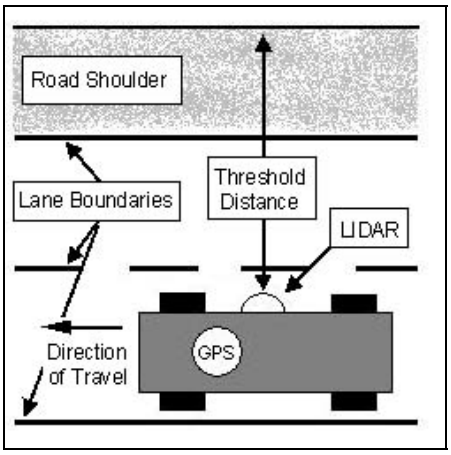

Figure 17 An example of a vehicle in the left lane of a two-lane road with the lane markings flagged. The threshold distance is the distance from the LIDAR sensor to the edge of the road shoulder

5.2 Data Clusters

After applying thresholds to remove data related to objects that are not on a drivable portion of the road, the remaining data points are broken into clusters based upon their location relative to each other. The program cycles through each of the data points and calculates the distance to nearby points. Any two points that are within a certain distance to each other are grouped into a cluster, and clusters containing two or more common points are merged together. Figure 18 shows an example of a set of sample data points and how they would be grouped into clusters. As the clusters are formed, a set of statistics for each one is generated containing various pieces of information that are used for object detection, such as the number of points in the cluster, the overall distance between the two furthest points, and the location in the field of view of the two furthest points in the cluster.

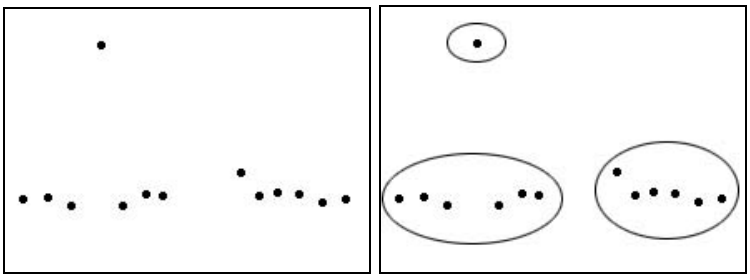

Figure 18 The left image shows a sample of raw LIDAR data while the right image shows how the data would be broken into clusters.

5.3 Vehicle Detection

From observations of the data as a vehicle passes the sensor, the shape of the data generally falls into one of three general cases that depend on where in the sensor field the vehicle lies. If the detection field of the sensor is divided into three sections, each of the general cases occurs when the vehicle is entirely in the left or right section, or at least partially in the center section. Figure 19 illustrates the three general cases and the sections of the sensor field that they correspond to.

The statistics generated for each cluster are compared to a set of criteria that is determined by the section in which the cluster resides. If most of the data resides in the central section, then the sensor should detect almost the entire length of the vehicle and thus provides a very accurate location of the front, rear, and side of the vehicle. However if the data resides in one of the side sections, it is possible that little of the side of the vehicle was detected while a reasonable portion of the front or rear of the vehicle was identified. This situation results in smaller clusters that provide a good estimate of the location of the front or rear corner of the vehicle.

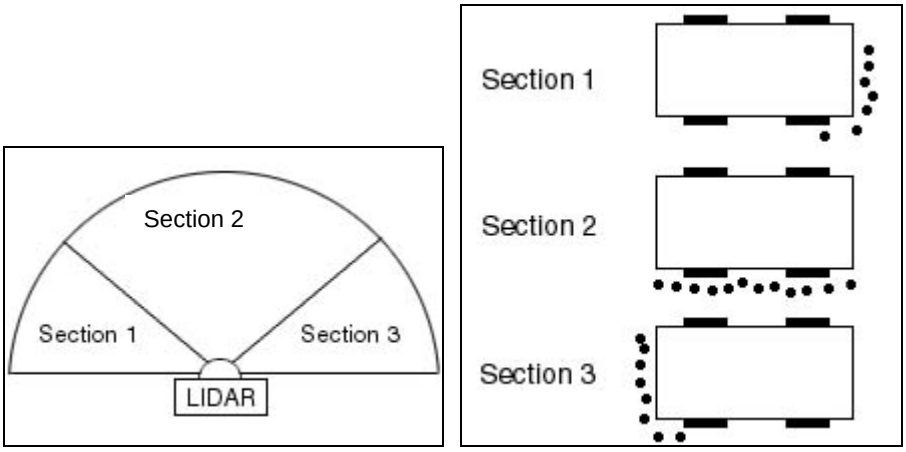

Figure 19 The left figure represents the area in which the LIDAR sensor detects objects divided into three sections. The right figure is an example of the general cases corresponding to each section. The spots represent typical LIDAR data as located relative to the sensed vehicle when that vehicle is located in either section 1, 2 or 3.

5.4 Vehicle Tracking

In order to extract more information about detected vehicles, such as the relative speed and heading, a method of tracking the vehicles must be implemented. Tracking is also

necessary to increase the reliability of the filter in different situations. Figure 20 shows an example where two vehicles are traveling in different lanes and are detected by the LIDAR sensor. In Figure 21 the vehicle in the far lane is seen passing the other vehicle, causing the far vehicle to be at least partially occluded. If the vehicle in the far lane is being tracked this situation can be detected and handled appropriately by using the estimated vehicle size and the detected corner. Note that should the far vehicle be at least partially occluded for the entire span of time in which it crosses the sensor field, the estimated location will always be inaccurate because the length of the vehicle will not be known. This is a limitation due to the line-of-sight nature of the sensor.

Another problem is caused when gaps are present in the data due to spurious reflections during a LIDAR scan. In such a case the criteria for determining the presence of a vehicle may not be met for that cluster. However, if the vehicle had been detected previously, the current position can be estimated and the filter could adjust the criteria to accommodate the situation.

A table of tracked vehicles is created to track objects from scan to scan. After vehicles are located from the current LIDAR scan, the position of each tracked vehicle in the table is extrapolated based upon the previous position, calculated speed, and calculated heading. If one of the current vehicle positions is within a certain tolerance of the position of an extrapolated tracked vehicle, then the data in the table is updated using the current vehicle information and the speed and heading are recalculated based upon the current and previous data. The special cases previously mentioned, such as a partially occluded vehicle, are also accounted for in this manner. If a tracked vehicle was not matched with a current vehicle, then the locations of the clusters that were not determined to be vehicles previously are checked. Any that are close to the location of where the vehicle should be are used to determine the current vehicle position and the relevant data in the tracking table is updated. If there is no match, then the data is updated using the extrapolated data and a flag is set so that other programs that use this data can warn the driver that it is an estimated position. If the vehicle is not reacquired within a few scans,

it is removed from the table. Once all tracked vehicles are accounted for, any other current vehicles that were previously undetected are added to the table.

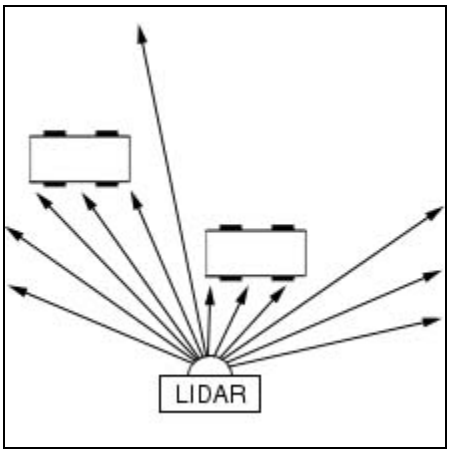

Figure 20 Two cars in different lanes detected by the LIDAR sensor

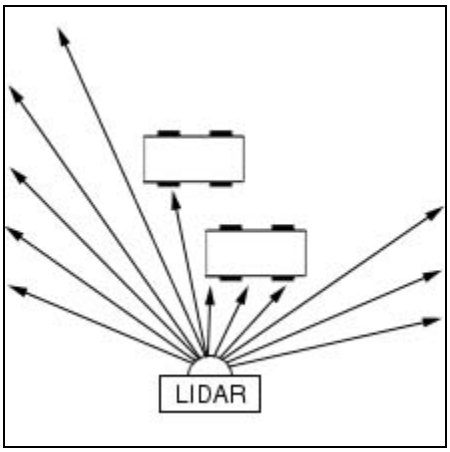

Figure 21 The vehicle in the far lane (furthest from the sensor) is partially occluded by the other vehicle, thus only the left half of the vehicle would be detected

After the tracking table has been completely updated, the vehicle position and orientation data is transformed into vehicle coordinates (based upon which side of the vehicle the LIDAR sensor is mounted) and transmitted via shared memory to other applications such as a virtual mirror or a heads up display.

6 Dynamic Virtual Mirror Performance

It must be proven that the virtual mirror is capable of displaying the view of the surroundings with a high degree of accuracy in order for a driver to make critical driving decisions using it. Therefore an experiment was designed that uses a high-resolution digital camera to represent a persons view of the side rearview mirror. By synchronizing the image capture with the position data from the DGPS receiver, the virtual mirror image is rendered and superimposed on the camera image. Any disparity between the features in the two images can be easily seen, and this is used to determine the accuracy of the virtual mirror representation.

6.1 Experimental Setup

The SAFEPLOW, a 2500 Series International truck shown in Figure 22, was equipped with a SR530 Leica DGPS receiver and used as the test vehicle. The Leica SR530 is a 24-channel dual-frequency receiver with on-board RTK. The manufacturer claims a position accuracy of 2cm. A Hitachi KP-F100 monochrome progressive scan digital camera, with a resolution of 1300 by 1034 pixels, was mounted such that it would capture images of the view in the passenger-side mirror as the experiment was run. The passenger-side mirror was replaced with a wider optical grade glass mirror to provide a larger field-of-view, allowing more of the road surface to be seen in the image. The camera provides a frame-on-demand feature that allows an image to be captured in response to an external trigger. Synchronization between the DGPS data and the image capture is critical to evaluating the accuracy of the system and was achieved by transmitting a trigger signal immediately upon receiving the data from the DGPS receiver.

Figure 23 illustrates the two computer systems that were used for the data collection: a Pentium 3/700mhz computer running Windows 2000 Professional to record digital images and an AMD K6-2/400mhz computer running QNX Real-Time Platform v6.1 to collect DGPS data and generate the synchronization signals.

Figure 22 The SAFEPLOW, a research vehicle used in the Intelligent Vehicles Laboratory

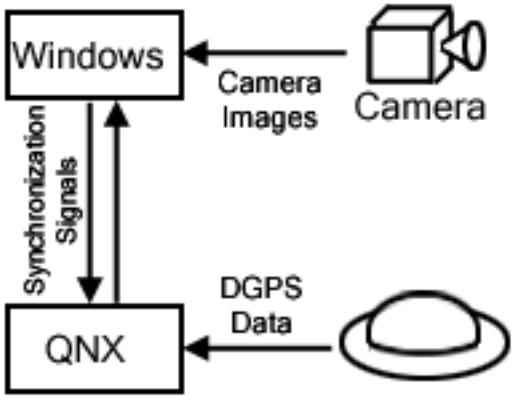

Figure 23 Overview of experimental system setup

6.2 Latency

An unknown in the system is the latency associated with the DGPS calculation. The DGPS receiver collects satellite data and then begins to calculate the position, which takes approximately 30 milliseconds according to the manufacturer. Since the serial transmission encompasses another 12 milliseconds, the image will be taken on the order of 42 milliseconds late. The camera shutter speed was set to 1/10000 second during the experiments, resulting in a negligible latency for the image capture relative to the other delays in the system. Therefore the assumption was made that the image was taken at the exact time the camera was triggered.

If the delay is known, the position can be projected using the DGPS position, speed, and heading. For this analysis the errors were calculated without projecting the position and the latency was estimated using the error and vehicle speed because it is not guaranteed to take exactly 42 milliseconds.

6.3 Analysis

The camera is triggered upon retrieval of the DGPS data, which occurs every 0.1 seconds. The virtual mirror renders the view for each data point, which is then superimposed on the respective camera image. Along with the lane boundaries and the calibration marks, a set of grid marks, as seen in Figure 24, is drawn near the calibration mark to determine the lateral and longitudinal errors. The longitudinal and lateral grid marks are spaced 0.25 meters apart, respectively. By counting the number of pixels between grid marks the width of a pixel in centimeters can be found in the area close to the calibration mark lines. Using this information and the number of pixels between the virtual mirror line and the line in the image, the lateral and longitudinal errors can be determined.

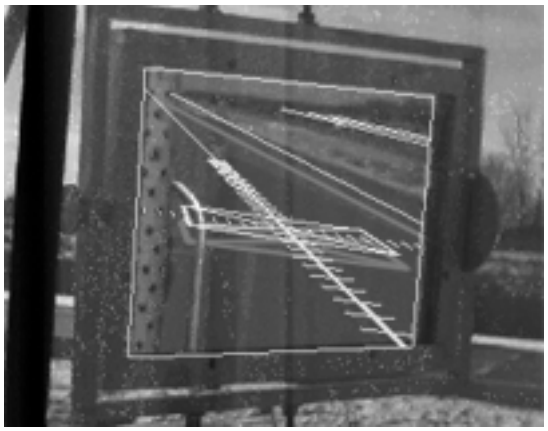

Figure 24 Grid marks drawn to calculate offsets (virtual mirror lines and grid marks are white)

For the purposes of evaluating the system, the errors are divided into longitudinal and lateral errors. Longitudinal is defined as being in the direction of travel of the vehicle while lateral is orthogonal to the direction of travel.

Experiments were performed at approximately 9.5 m/s (21mph) and 18 m/s (40mph) along a straight portion of the road to analyze the dynamic longitudinal and lateral errors. The speed of the host vehicle was calculated during the experiment using successive DGPS positions. Using the previously described method of analyzing the pixel sizes, the average longitudinal error over a series of experiments was found to be 0.40 meters at 9.5

m/s and 0.77 meters at 18 m/s. The lateral errors at 9.5 m/s and 18 m/s were determined to be 0.023 meters and 0.047 meters respectively. A table containing detailed information for these experimental runs can be seen in Table 1. The average longitudinal and longitudinal errors over all of the experiments performed were 0.031 meters and 0.027 meters respectively, with an overall mean speed of 13.92 m/s.

Errors as a result of system latency should increase linearly with speed, as the latency is independent of the motion of the vehicle. The ratio of the average speeds for the 2 groups of experiments is:

$$18.16 m/s \,/\, 9.47 m/s = 1.917 \qquad (6.1)$$

The ratio of the average longitudinal errors of the 2 groups is:

$$0.77 m \,/\, 0.40 m = 1.925 \qquad (6.2)$$

An increase in speed results in an almost linear increase in the longitudinal error seen in the virtual mirror.

Assuming that the error due to latency is far greater than the errors from calibration and truck vibrations, dividing the longitudinal errors for each frame by the current speed of the truck will result in an estimate of the latency. The resulting average latency over several experiments was found to be 41.71 milliseconds, which is very close to the 42 milliseconds previously estimated. Thus if the position is projected ahead by 42 milliseconds before drawing the virtual mirror display, the longitudinal error will be reduced to be almost zero as long as the velocity is reasonably constant during the 42 milliseconds. Table 2 shows a chart with the errors for the same sets of experiments shown in Table 1 except that the position was estimated using a latency of 42 milliseconds. The lateral errors were unchanged, but there was a significant improvement in the longitudinal errors.

Experimental Run	1	2
Mean Speed (m/s)	9.47	18.16
Mean Lateral Error (m)	0.023	0.047
Lateral Std Deviation (m)	0.011	0.012
Mean Longitudinal Error (m)	0.396	0.768
Longitudinal Std Deviation (m)	0.036	0.052
Estimated Latency (ms)	41.8	42.3

Table 1 Statistics and results for experiments before correcting for latency.

Experimental Run	1	2
Mean Speed (m/s)	9.47	18.16
Mean Lateral Error (m)	0.023	0.047
Lateral Std Deviation (m)	0.011	0.012
Mean Longitudinal Error (m)	-0.006	0.006
Longitudinal Std Deviation (m)	0.042	0.051

Table 2 Statistics and results for the same experiments with latency compensation.

7 Vehicle Detection Accuracy

Just as it is necessary for the virtual mirror to be able to accurately display the surrounding road to the driver, it is critical that objects that are not stored in the database, such as other vehicles, be detected to relay this information to the driver. Therefore the algorithm described in Chapter 5 must be analyzed to determine how accurately it is able to detect the location of nearby vehicles. To this end an experiment was devised in which both the host and target vehicles are equipped with DGPS receivers and a wireless network is established to allow the vehicles to transmit their locations to one another. The relative DGPS locations are compared to the results from the vehicle detection system to determine the precision of the algorithm.

7.1 Experimental Setup

The experimental setup for evaluating the vehicle detection algorithm is similar to that used for the virtual mirror experiments in Chapter 6. The SAFEPLOW was equipped with a SR530 Leica DGPS receiver and used as the host vehicle. A SICK LMS-221 scanning laser sensor was mounted on the passenger side of the host vehicle with the sensor pointed to the right of the vehicle. The LMS-221 laser measurement system is capable of scanning a 180-degree field with an angular resolution of 0.5 or 1.0 degrees. The sensor has a range measurement resolution of 1 cm and an estimated error of +/-6 cm (the accuracy of the measurement is affected by target surface reflectivity and environmental conditions). During experimentation the sensor was configured to scan with 1.0-degree resolution and transmit the data 38.4kbd, allowing for a 10hz data update. For inter-vehicle communication the host truck uses a Breezecom AP-10X2C PRO.11 wireless LAN access point configured with a transmission rate of 1 megabit per second. The AP-10 X2C is an IEEE 802.11-compliant network device with a maximum operating range of approximately 900 meters.

While DGPS data will be used with the vehicle detection results to determine the accuracy of the vehicle detection algorithm, the virtual mirror will demonstrate an application for which vehicle detection can be utilized, and provide qualitative representation of the accuracy. For this purpose, a Hitachi KP-F100 monochrome

progressive scan digital camera was mounted on the host vehicle such that it would capture images of the view in the passenger-side mirror as the experiment was run. The passenger-side mirror was replaced with a wider mirror to provide a larger field-of-view, allowing more of the road surface to be seen in the image. Synchronization between the DGPS data and the image capture was achieved by transmitting a trigger signal immediately after receiving the data from the DGPS receiver. After compensating for various latencies, explained in detail in Chapter 7.1, the virtual mirror will render the scene consisting of the road features and the target vehicle seen in the mirror.

Figure 25 illustrates the two computer systems that were used for the data collection in the host vehicle: a Pentium 3/700mhz computer running Windows 2000 Professional to record digital images and an AMD K6-2/400mhz computer running QNX Real-Time Platform to collect DGPS data and generate the synchronization signals.

A 1994 Mazda B2300 pickup truck was used as a test vehicle for determining the accuracy of the vehicle detection algorithm using the LIDAR sensor. This vehicle will hereafter be referred to as the target vehicle. This vehicle was also equipped with a SR530 Leica DGPS receiver and an AMD K6-2/400Mhz computer with QNX Real-Time Platform. To transmit the DGPS information to the host vehicle, this truck was equipped with a Breezecom SA-10D PRO.11 WLAN station adapter. Figure 26 shows an overview of the system setup in the target vehicle.

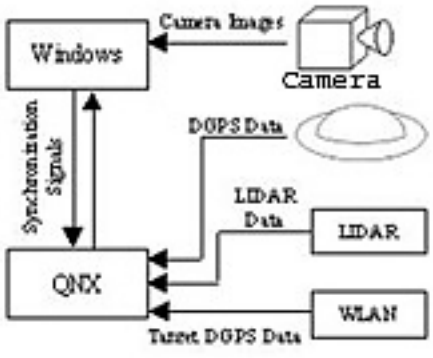

Figure 25 Overview of the system setup in the host vehicle

Figure 26 Overview of the system setup in the target vehicle

7.2 Latency Compensation

The DGPS units and the LIDAR were run asynchronously during the experiments, as hardware did not provide a synchronization signal. Therefore all data was time-stamped using a high-resolution timer (accurate to several nanoseconds) in order to extrapolate positions during post-processing. During normal operation of the system, this position extrapolation can be performed in real-time to minimize any errors due to the lack of sensor data synchronization. Each of the positions was extrapolated based upon the speed and heading of each vehicle such that the projected positions represent the location of the vehicles at the time that the LIDAR data arrived.

There are two latencies involved for each of the sensors: the measured time difference between the DGPS data and the LIDAR data, and the estimated calculation and communication time for each set of sensor data. The Leica SR530 DGPS units consume approximately 30 milliseconds to calculate a position and 12 milliseconds to transmit this data across the serial port and process it on the QNX computer. Thus a time delay of 42 milliseconds was used for projecting the target and host DGPS positions. This time delay was found to be an accurate estimation during tests conducted to determine the accuracy of the virtual mirror (Chapter 6). Therefore to project the DGPS position of each vehicle ahead in time to its estimated position at the time the LIDAR data arrived, the equations

$$\begin{aligned} \Delta t_1 &= (t_{LIDAR} - t_{GPS}) + 0.042 \\ x_{project} &= x_{GPS} + \Delta t_1 * speed * \cos(heading) \\ y_{project} &= y_{GPS} + \Delta t_1 * speed * \sin(heading) \end{aligned} \qquad (7.1)$$

are used where Δt_1 is the time elapsed from the start of the DGPS calculation to the arrival of the LIDAR data, t_{LIDAR} is the time the LIDAR data arrived, t_{GPS} is the time the DGPS data arrived, x_{GPS} and y_{GPS} are the coordinates received from the DGPS unit, and $x_{project}$ and $y_{project}$ are the estimated position coordinates at the time the LIDAR data arrived. These equations are used to estimate the host and target vehicle positions, substituting the time, heading, and position information for the relevant vehicle into the appropriate variables.

The SICK LMS-221 LIDAR unit completes a 180 degree scan in 13 milliseconds and then sends a 372 byte message across the serial port, which configured at 38,400 baud will consume 77.5 milliseconds. The position of the vehicle as determined by the algorithm is extrapolated in the same manner as the DGPS positions except that Δt is redefined to represent the time elapsed from the start of the LIDAR scan to the time the LIDAR data is available:

$$\begin{aligned} \Delta t_2 &= 0.013 + 0.0775 \\ x_{project} &= x_{GPS} + \Delta t_2 * speed * \cos(heading) \\ y_{project} &= y_{GPS} + \Delta t_2 * speed * \sin(heading) \end{aligned} \qquad (7.2)$$

7.3 Coordinate Systems

The host and target vehicle positions are in Minnesota south state plane coordinates whereas the vehicle detection algorithm identifies vehicles in a local coordinate system defined relative to the LIDAR sensor. To assess the accuracy of the vehicle detection results, the DGPS positions of the vehicles are used to calculate the location of the target vehicle relative to the LIDAR sensor in the local coordinate system. The results of the analysis are identical whether performed by comparing the real and estimated vehicle location in global or local coordinates. The local coordinate system was used as the *x* and *y* coordinates are equivalent to the longitudinal and lateral errors, simplifying the analysis. The target vehicle DGPS position was transformed into local coordinates using the following equations:

$$
\begin{aligned}
x_{target\_local} &= x_{target} * \cos(heading_{host}) \\
&\quad + y_{target} * \sin(heading_{host}) \\
&\quad - x_{host} * \sin(heading_{host}) \\
&\quad - y_{host} * \cos(heading_{host}) \\
&\quad - x_{LIDAR} \\
y_{target\_local} &= -x_{target} * \sin(heading_{host}) \\
&\quad + y_{target} * \cos(heading_{host}) \\
&\quad + x_{host} * \sin(heading_{host}) \\
&\quad - y_{host} * \cos(heading_{host}) \\
&\quad - y_{LIDAR}
\end{aligned}
\tag{7.3}
$$

where the target and host coordinates use the global coordinate system and the LIDAR coordinates are the location of the LIDAR sensor relative to the host DGPS antenna in local coordinates. Using these equations the x and y position of the target vehicle can be compared to the location of the vehicle found by the LIDAR filter to analyze the accuracy.

7.4 Analysis

The accuracy of the LIDAR filter is divided into two errors: the longitudinal and lateral error. The longitudinal error is defined as the difference in the local *x* coordinates between the actual target DGPS position and the position determined by the vehicle detection algorithm using LIDAR data. The lateral error is the difference in the local *y* coordinates.

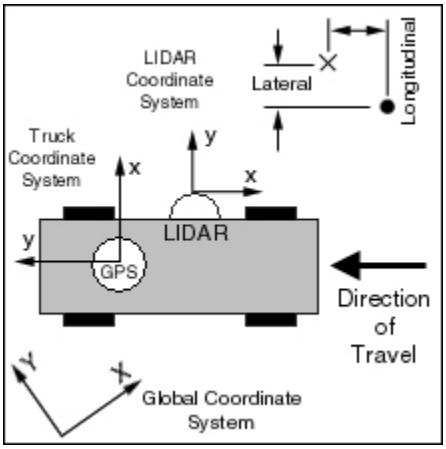

Figure 27 The host vehicle showing the local coordinate system attached to the LIDAR unit, a dot showing the target vehicle position determined by DGPS, an X for the position from LIDAR, and the longitudinal and lateral errors marked.

As seen in Figure 28, Figure 29, Figure 30, and Figure 31 the algorithm performs rather well at determining the location of the target vehicle at various speeds. Each plot represents the errors and speeds associated with a single experimental run. The experiments involved one vehicle in a stationary position with the other vehicle traveling at approximately 20mph or 40mph, representing relative speeds that can be experienced during normal driving situations. At 20mph the average lateral error over a large set of experimental runs was determined to be 0.009m with a standard deviation of 0.075m and the average longitudinal error was 0.031m with a standard deviation of 0.057m. At 40mph the average lateral error was 0.023m with a standard deviation of 0.061m, while the average longitudinal error was 0.044m with a standard deviation of 0.073m.

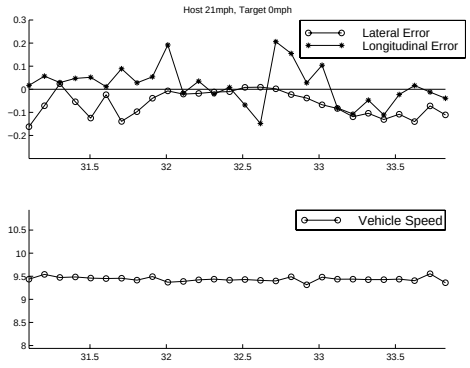

Figure 28 Longitudinal and lateral error plot and speed plot for an example experimental run. The target vehicle was stationary while the host vehicle traveled at an average speed of 21mph.

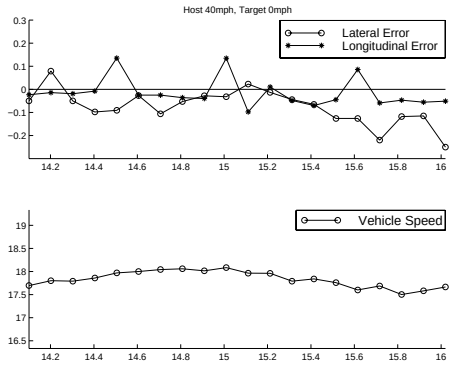

Figure 29 Longitudinal and lateral error plot and speed plot for an example experimental run. The target vehicle was stationary while the host vehicle traveled at an average speed of 40mph.

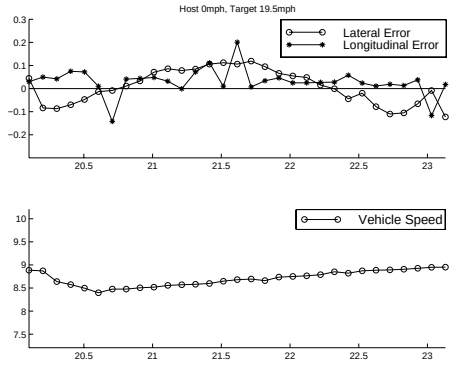

Figure 30 Longitudinal and lateral error plot and speed plot for an example experimental run. The host vehicle was stationary while the target vehicle traveled at an average speed of 19.5mph.

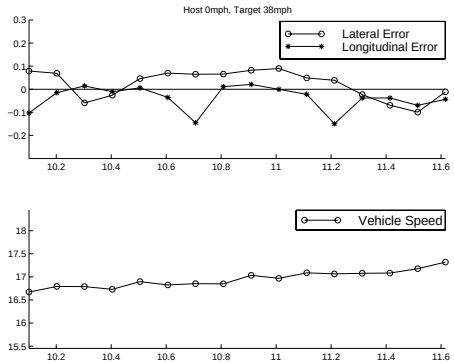

Figure 31 Longitudinal and lateral error plot and speed plot for an example experimental run. The host vehicle was stationary while the target vehicle traveled at an average speed of 38mph.

7.5 Application to the Virtual Mirror

The virtual mirror application was used to illustrate how the LIDAR-based vehicle tracking may be used. The data was transmitted via shared memory from the tracking program to the virtual mirror and converted to global coordinates based upon the current DGPS position of the vehicle and the location of the LIDAR sensor relative to the DGPS antenna. The target vehicle is represented with a bounding box to show the position of the bounding sides as determined by the vehicle detection algorithm described earlier. For illustrative purposes, the virtual mirror rendering is superimposed on the camera images gathered during the experiments. This provides a qualitative measure of the accuracy of the algorithm by observing any disparities between the bounding box and the location of the target vehicle in the image. A camera image containing a mirror view of the target vehicle is shown in Figure 32. An example of a virtual mirror rendering and the superimposed image of this scene can be seen in Figure 33.

Figure 32 A camera image showing the target vehicle as seen in the mirror.

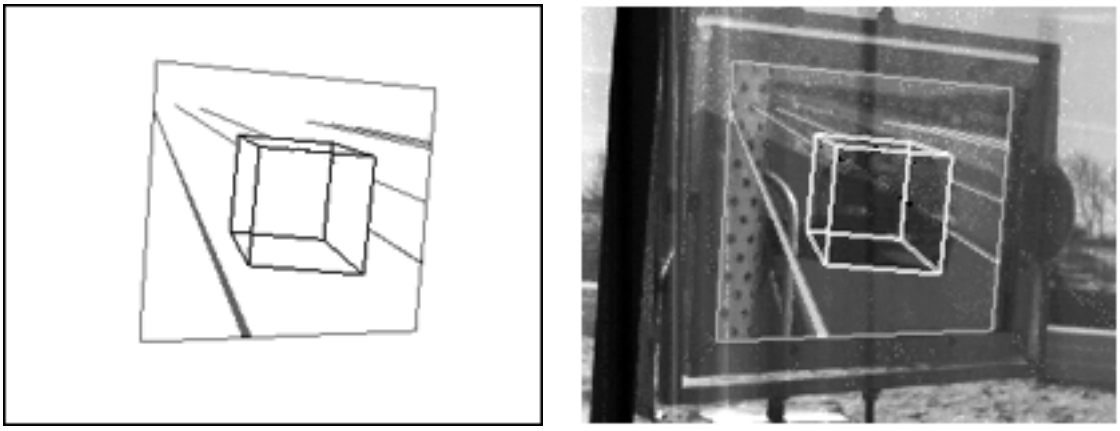

Figure 33 Virtual mirror rendering (left, with line color inverted) and the same rendering superimposed on a camera image (right). A bounding box is drawn at the target vehicle location.

8 Conclusions

This report outlines the concept and implementation of the virtual mirror. As it is a computer-generated display based on various sensor data, it overcomes many of the limitations of standard optical mirrors. The virtual mirror is capable of reproducing the road seen in a mirror to a high degree of accuracy, under 5 cm of lateral error and almost 0 cm of longitudinal error after latency compensation, through the use of high-accuracy DGPS. Coupled with the LIDAR-based detection and tracking system other vehicles can also be displayed accurately, allowing the virtual mirror to provide the driver with critical information that may otherwise be unavailable with the use of a conventional optical mirror due to blind zones or limited visibility conditions. The virtual mirror can also replicate a large mirror placed alongside the vehicle that would provide a wider field of view from a better vantage point, but would be impossible to mount on the vehicle. Placing the display inside of the vehicle also has benefits in regards to aerodynamic drag and the amount of time the driver's focus is taken away from the road.

The system was evaluated by superimposing the virtual mirror image on an image of the real mirror taken by a digital camera. This required that the position and orientation of the camera and mirror be known, but accurately measuring these values is not feasible. Therefore an algorithm was developed that would determine these parameters through the use of a set of images containing road markings. This algorithm takes a range of parameter values and alters the parameters by small increments, using a weighted sum of errors between markings in the virtual mirror image and camera image as a measure to determine the accuracy of the parameters. Once the process is done, the set of values with the smallest measure is used to generate the virtual mirror images for the accuracy evaluation.

With the virtual mirror calibrated for the camera and mirror locations during the experiments, the virtual mirror images are created and again superimposed on the camera images. By incorporating a series of grid marks along the road markings seen in the virtual mirror, the approximate height and width of the pixels can be determined. Using

these values and the number of pixels between the markings in the virtual mirror and camera images, the error can be calculated.

Evaluating the accuracy of the LIDAR-based vehicle tracking system required the location of both the target and host vehicles to be known very accurately. To accomplish this both vehicles were equipped with DGPS receivers and a high-speed wireless network was created to allow for inter-vehicle communication. After transforming the DGPS positions and the results of the vehicle detection algorithm into a local coordinate system, the values are compared to determine the accuracy. The analysis showed that the system was accurate to less than 5 centimeters on average, with a maximum error of 20 centimeters during the experiments.

While the results of the analyses of the vehicle detection algorithm are promising, there are some limitations to the situations in which the sensor used will perform well. The SICK LMS-221 sensor was mounted such that the sensor sweeps along a plane that is parallel to the ground. Therefore the height of the sensor from the ground is one of the most important factors affecting the system's utility since the ground clearance of vehicles can vary drastically. If the sensor is mounted lower to the ground it is optimal for detecting most cars, but sport utility vehicles and trucks that have more ground clearance may not be detected well or not at all. If the sensor is raised to accommodate for this, the sensor beams will hit the windows of low riding cars, causing the filter to either not see the vehicle or return erroneous results since the effects of the beams striking window glass are inconsistent.

Using multiple LIDAR units at different heights can solve the problem, as the data from both sensors can be combined to filter out erroneous readings. This implies that a re-designed LIDAR that incorporates measurements from two or three different planes may be best for this application. Another alternative is to use a rear-looking RADAR unit mounted near the front of the vehicle. The RADAR would provide an approximate location of any approaching vehicles, and the LIDAR filter could use this information to improve tracking. In the event that the LIDAR sensor does not detect the vehicle at all,

the RADAR data can be used to reexamine the LIDAR data and attempt to accurately locate the vehicle, or determine an approximate location and pass this to other programs until the LIDAR sensor is able to detect it.

The effects of sudden accelerations were not examined during these experiments. The SICK LMS-221 sensor used is capable of transmitting data over a 500kbps serial line, providing data with a total delay of 26 milliseconds. This would imply that a small change in velocity should not have a detrimental affect on the results, but further experimentation must be performed to determine the sensitivity of the algorithm in such a situation.

In conclusion, the virtual mirror is a display that has been developed to assist drivers in making safe maneuvering decisions in normal and low visibility conditions that may otherwise be difficult with standard optical mirrors. Through the use of high accuracy differential GPS systems, scanning laser range finders, and a geo-spatial database the system has been shown to provide accurate and reliable information to achieve this goal. With further testing and improvements in the sensors used, the virtual mirror system can be integrated into vehicles to improve the driving environment.

References

1 United States Code of Federal Regulations (CFR) Title 49, Transportation; Chapter V, Department of Transportation's National Highway Traffic Safety Administration; Part 571, Federal Motor Vehicle Safety Standard 111, “Rearview mirrors”, October 1998.

2 Olson P. and Sivak M., “Glare from automobile rear-vision mirrors”, Human Factors. Vol. 26 No. 3, June 1984, 269-282.

3 Ranney T., Simmons L. and Masalonis A., “The immediate effects of glare and electrochromic glare-reducing mirrors in simulated truck driving”, Human Factors. Vol. 42 No. 2, Summer 2000, 337-347.

4 Flannagan, M. and Sivak, M., “Nighttime Effectiveness of Rearview Mirrors: Driver Attitudes and Behaviors”, SAE Technical Paper Series No. 900567, Warrendale, Pennsylvania, Society of Automotive Engineers, 1990.

5 Cresswell M, and Hertz P., “Aerodynamic drag implications of exterior truck mirrors”, SAE Technical Paper Series No. 920204, Warrendale, Pennsylvania, Society of Automotive Engineers, 1992.

6 Pilhall S., “Improved Rearward View”, SAE Technical Paper Series No. 810759, Warrendale, Pennsylvania, Society of Automotive Engineers, 1981.

7 Garlich-Miller M. and Donath M., “A Connectionist Approach to the Fusion of Three Dimensional, Sparse, Unordered Sensor Data", Proceedings of the Japan-U.S.A. Symposium on Flexible Automation, Boston, MA, July 8-10, 1996.

8 Pardhy S., Shankwitz C. and Donath M., “A Virtual Mirror For Assisting Drivers”, Proceedings of IV2000 - IEEE Intelligent Vehicles Symposium, Dearborn, USA, October 2000.

9 Lim H.-M., Newstrom B., Shankwitz C., and Donath M., “A Conformal Augmented Head Up Display For Driving Under Low Visibility Conditions”, AVEC 2000, 5th International Symposium on Advanced Vehicle Control, Ann Arbor, August 2000.

10 Lim, H.M., Newstrom, B., Shankwitz, C., and Donath, M., “A heads up display based on a DGPS and real time accessible geo-spatial database for low visibility driving”, Proceedings of the 12th International Meeting of the Satellite Division of the Institute of Navigation (ION GPS '99), Nashville, Tennessee, September 1999.

11 Mazl, R. and Preucil, L., "Building a 2D environment map from laser range-finder data", Proceedings of IV2000 - IEEE Intelligent Vehicles Symposium, Dearborn, USA, October 2000.

12 Arras, K. and Vestli, S., "Hybrid, High-Precision Localisation for the Mail Distributing Mobile Robot System MOPS", Proceedings of the IEEE International Conference on Robotics and Automation, Leuven, Belgium, May 1998.

13 Javier, G., Stentz, A. and Ollero, A., "A Mobile Robot Iconic Position Estimator Using a Radial Laser Scanner", Proceedings of the IEEE Robotics and Automation Conference, Nice, France, May 1992.

14 Osugi, K., Miyauchi, K., Furui, N. and Miyakoshi, H., "Development of the scanning laser radar for ACC system", JSAE Review. Vol. 20 No. 4, 1999, 549-554.

15 Kirchner, A. and Ameling, C. "Integrated obstacle and road tracking using a laser scanner", Proceedings of IV2000 - IEEE Intelligent Vehicles Symposium, Dearborn, USA, October 2000.

16 Ewald, A. and Willhoeft, V., "Laser scanners for obstacle detection in automotive applications", Proceedings of IV2000 - IEEE Intelligent Vehicles Symposium, Dearborn, USA, October 2000.

17 Foley J.D., van Dam A., Feiner S.K., Hughes J. F., "Computer Graphics – Principles and Practice", 2nd ed., Addison-Wesley Publishing Company, 1997.

18 Rogers D.F., Adams J.A., "Mathematical Elements for Computer Graphics", 2nd ed., McGraw-Hill, Inc., 1990.

Appendix

A.1 MnROAD Test Facility

All of the experiments were performed on the low volume road at Minnesota Road (MnROAD), a Minnesota Department of Transportation research facility. MnROAD is a closed-track, outdoor pavement laboratory with an extensive sensor network used to study the effects of weather and heavy commercial truck traffic on various pavement materials and designs. The continuous track consists of two long straight roadways connected by looped sections. Figure 34 shows an image of the loop at the western end of the MnROAD test track. MnROAD was an ideal area for testing rather than a standard roadway as it provided a controlled environment with no traffic, allowing for various static and dynamic tests while avoiding interactions with other vehicles.

A database of the MnROAD low volume road was used that contained data for the location of the lane boundaries, road shoulders, and calibration marks. The position data stored in the database was based on DGPS surveyed positions in Minnesota South State Plane coordinates. Figure 35 shows an image drawn using the information stored in the database and Figure 36 illustrates a zoomed in view of a section of road containing a calibration mark that is painted on the road surface.

Figure 34 The low volume road at the MnROAD research facility. The west loop of the track is seen in the image.

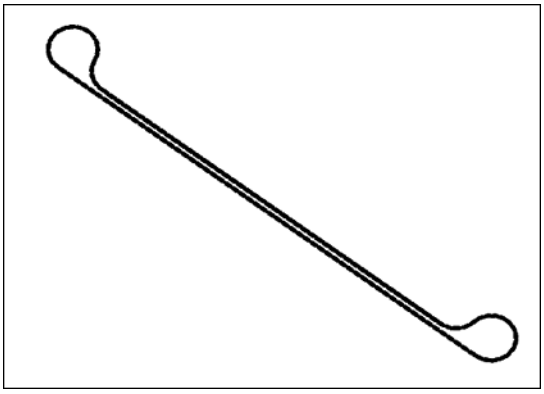

Figure 35 Overview of the MnROAD map.

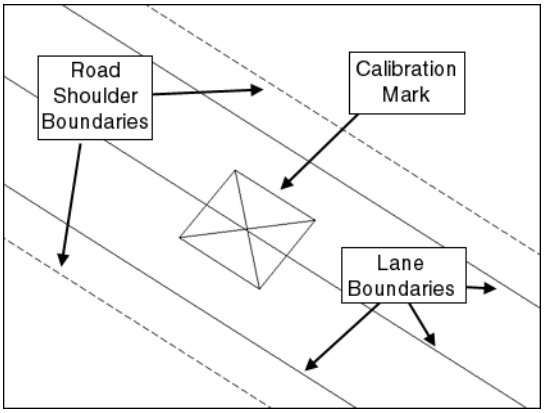

Figure 36 The MnROAD map, zoomed in at the location of a calibration mark painted on the road.