

Development of an Integrated Sensor System for Obstacle Detection and Terrain Evaluation for Application to Unmanned Ground Vehicles

Carl D. Crane III^a, David G. Armstrong II^a, Maryum Ahmed^a, Sanjay Solanki^a, Donald MacArthur^a, Erica Zawodny^a, Sarah Gray^b, Thomas Petroff^b, Mike Grifis^c, Carl Evans^d

^aUniversity of Florida, Center for Intelligent Machines and Robotics, Gainesville, Florida

^bAutonomous Solutions, Inc., Young Ward, Utah

^cThe Eigenpoint Co., High Springs Florida

^dApplied Perception Inc., Wexford Pennsylvania

ABSTRACT

This paper describes the development and performance of a sensor system that was utilized for autonomous navigation of an unmanned ground vehicle. Four different sensor types were integrated to identify obstacles in the vicinity of the vehicle and to identify smooth terrain that could be traversed at speeds up to thirty miles per hour.

Two sensors were used for obstacle detection. The first was a Sick LMS200 lidar attached to a spinning shaft with an encoder and a slip ring that transmits range data at a 500 kbaud rate. It creates a 3D representation of the world that is orthogonally projected into a 2D space. The second obstacle detection sensor was a stereo vision system that was manufactured by Videre Design. The system utilized 12.5 mm focal length lenses which provided a horizontal and vertical field of view of 50 degrees and 38 degrees, respectively.

Two sensor systems were utilized to evaluate the smoothness of terrain and to identify regions of good traversability. A stationary Sick lidar was mounted to provide range information at a location on a theoretical flat ground-level plane twenty meters in front of the vehicle. Monocular cameras and vision processing algorithms were used as the second sensor system to identify traversable terrain. Here, parts of images that closely resemble the region of the image directly in front of the vehicle were identified. This assumes that the region in front of the vehicle is a traversable path.

The paper also describes a sensor fusion approach that was developed whereby the output of all sensors was in a common grid based format. The environment around the vehicle was modeled by a 120×120 grid where each grid cell was 0.5m× 0.5m in size and where the orientation of the grid lines was always maintained parallel to the north-south and east-west lines. Every sensor output an estimate of the traversability of each grid cell. For the three dimensional obstacle avoidance sensors (rotating lidar and stereo vision) the three dimensional point data was projected onto the grid plane. The terrain traversability sensors, i.e. fixed lidar and monocular vision, estimated traversability based on smoothness of the spatial plane fitted to the range data or the commonality in appearance of pixels in the grid cell to those directly in front of the vehicle, respectively.

1 INTRODUCTION

The sensor architecture was originally developed for the perception system on the Team CIMAR NaviGator. The NaviGator was developed for the 2004 Defense Advanced Research Projects Agency (DARPA) Grand Challenge, held in March of 2004. The competition was to develop an unmanned ground vehicle which could autonomously navigate and avoid obstacles over approximately 140 miles of the Mojave Desert. Once that event was over, these ideas were applied to related applications at the Air Force Research Laboratory.

For autonomous navigation, the vehicle should be equipped with reliable sensors for sensing the environment, building environment models and following the right path. The environmental features generally used for building models are the RGB color space, range data, and texture. Any of these sensors may be preferable, depending on the surrounding environment. This paper discusses the implementation of multiple sensors.

2 SYSTEM ARCHITECTURE

The system architecture is based on the Joint Architecture for Unmanned Systems (JAUS) Reference Architecture, Version 3.0 [1]. JAUS defines a set of reusable components and their interfaces. At the highest level, the architecture consists of four fundamental elements as depicted in Figure 1:

- Planning Element: the components that act as a repository for *a priori* data such as known roads, trails, or obstacles, as well as data that specify the acceptable boundaries within which the vehicle must operate, along with the components that perform off-line planning based on that data.
- Control Element: The components that perform closed-loop control in order to keep the vehicle on a specified path

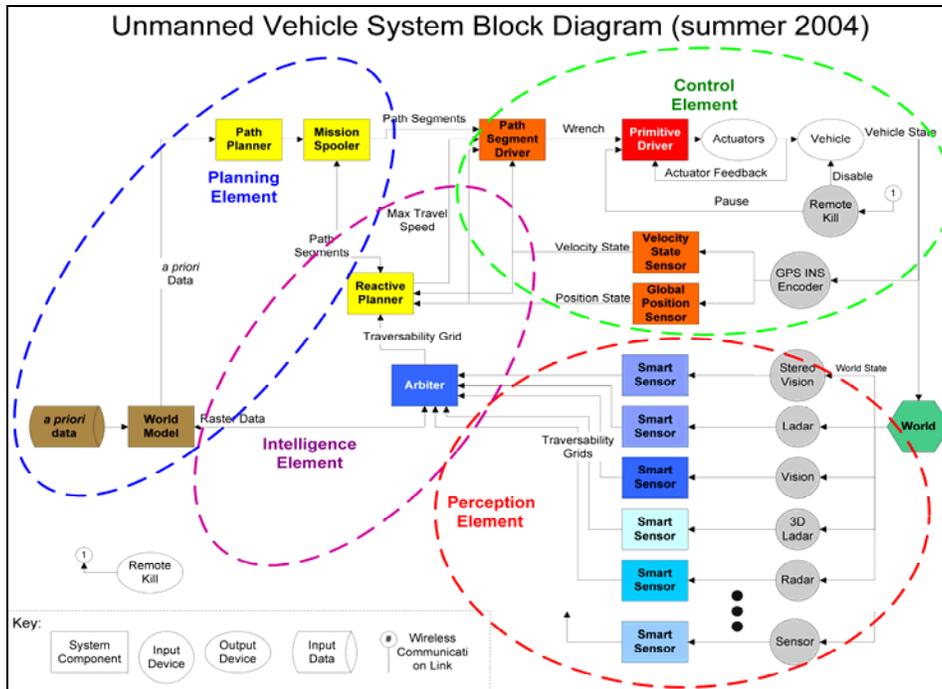


Figure 2.1: System Architecture

- Perception Element: The components that perform the sensing tasks required to locate obstacles and to evaluate the smoothness of terrain. The four sensors that were used for these tasks are discussed in detail in Sections 4 and 5 of this paper.
- Intelligence Element: The components that act to determine the ‘best’ path segment to be driven based on the sensed information. Generation of a single traversability grid that combines the inputs from multiple sensors. Sensor fusion is discussed in Section 3.

Recognizing the power of JAUS to simplify system integration, Team CIMAR used it throughout the NaviGator. The exception to this was the perception system where experimental messages that defined the smart sensor messaging architecture were used.

The planning element and control elements and the application of JAUS are discussed in detail in a companion paper, “Planning and modeling extensions to the Joint Architecture for Unmanned Systems (JAUS) for application to unmanned ground vehicles”, paper # 5804-16.

3 SENSOR FUSION

The basis of the sensor architecture is the idea that each sensor processes its data independently of the system and provides a logically redundant interface to the other components within the system. This allows developers to create their technologies independently of one another and process their data as best fits their system. The sensor can then be integrated into the system with minimal effort to create a robust perception system.

A common data structure, called the Traversability Grid, was introduced for use by all sensors [2]. A visualization of this grid is shown in Figure 3.1. The upper level shows the world as a human sees it. The lower level shows the Grid representation based on the fusion of sensor information.

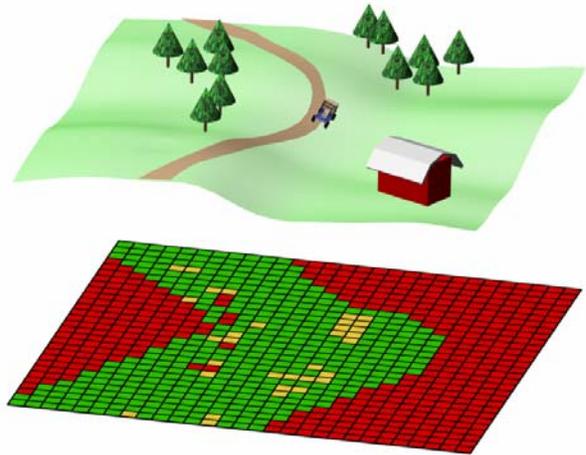


Figure 3.1: Traversability Grid Portrayal

The scoring of each cell is based on mapping the sensor’s assessment of the traversability of that cell into a range of 0 to 255 where 0 means that there is absolutely an insurmountable obstacle detected in that cell, 255 means there is absolutely a desirable, easily traversed surface in that cell, and 127 means that the traversability of that cell is unknown.

All of these characteristics are the same for grids sent from every sensor, making seamless integration possible, with no predetermined number of sensors. The grids are sent to the arbiter which is responsible for fusing the data. The arbiter then sends a grid with all the same characteristics to the planning element of the system. Since the input and the output of the arbiter are the same format, it is possible to remove the arbiter and the array of sensors, and replace them with a single sensor. The single sensor can then send its grid directly to the planning element, with no fusion necessary. This makes it possible to test each individual sensor’s performance with ease.

The messaging concept for marshalling grid cell data from sensors to the arbiter and from arbiter to planner is to send only those cells that have changed. Thus, for each scan or iteration, the sending component must determine which cells

in the grid have new values and pack the row, column, and value of that cell into the current message. This technique greatly reduces the network traffic and message-handling load for nominal cases (i.e., cases in which most cells remain the same from one iteration to the next). In order to properly register a given sensor's output with that of the other sensors, the message must also provide the latitude and longitude of the center cell (i.e., vehicle position at the instant the message and its cell values were determined).

With the Traversability Grid in place to normalize the outputs of a wide variety of sensors, the data fusion task becomes one of arbitrating the matching cells into a single output finding for that cell. To accomplish this, the Arbiter must take in each new message from a given sensor, adjust its center-point (assuming that the vehicle has moved between the instant in time when the sensor's message was built and the instant in time when the arbitrated output message is being built) and apply the new data values to the baseline grid for that sensor. This step must be repeated for each sensor that has sent a message. At this point, all input grids are aligned and contain the latest findings from its source sensor. Now the arbiter must traverse the input grids simultaneously, cell-by-cell, and merge the data from each corresponding cell into a single output value for that row/column location. Naturally, this process can be optimized by keeping track of which cells experienced a change and, if no change occurred in any of the input cell values, then there is no need to compute a new output value for that cell. For early testing, a simple average of the input cell values was used as the output cell value. Future work will investigate other algorithms, including heuristic ones, to perform the data fusion task. The Arbiter component was designed to make it easy to experiment with varying fusion algorithms in support of on-going research. Once all cells have been treated in this fashion, the Arbiter packs up its output grid message and sends it on the dynamic planner.

4 OBSTACLE DETECTION

4.1 Spinning Ladar

The spinning Ladar generates a 3D point cloud of the surrounding environment. The technology implements a spinning actuation of 2D laser range finder to generate 3D data. The sensor is mounted forward facing in front of the vehicle and the sensing area of the laser is hemispherical. The viewing angle of the system is 360° by 180° . Figure 4.1 shows typical output from the spinning Ladar.



Figure 4.1 Output from the spinning Ladar

4.2 Stereo Vision

Stereo vision can be used to detect the location of objects in a robot's world. It can also give valuable information about objects, such as color, texture, and patterns that can be used in intelligent systems to classify objects.

Artificial stereo vision systems retrieve three-dimensional information about their surroundings using the same principles as the human visual system. Two cameras each take a picture of a scene at the exact same time. Since these images were taken from different viewpoints, features appear at different locations in each image. The closer the

feature is to the cameras, the greater the difference in location of that feature in the images. The measure of this distance between features is called disparity. Based on the disparity, the three-dimensional location of objects can be found.

Camera calibration generates a mathematical model of the imaging system. This model transforms data from the three-dimensional world coordinate system to the two dimensional image planes of the cameras.

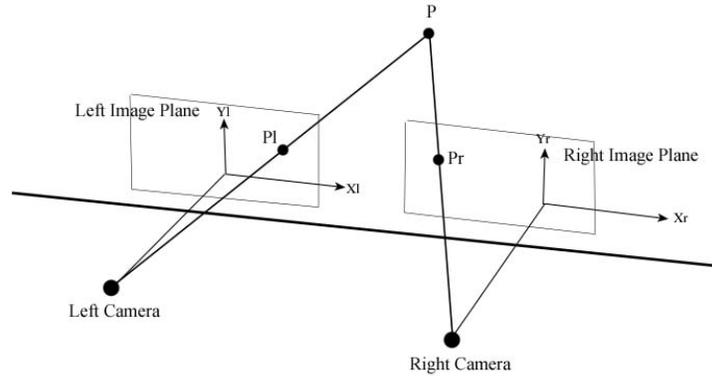


Figure 4.2: Stereo Vision Geometry

Figure 4.2 shows the geometry of a stereoscopic vision system. Point P is projected onto each image plane. The projection is shown as P_r in the right image and P_l in the left image. When cameras are lined up horizontally, there is a horizontal difference in the location of P_r and P_l . There is no vertical difference. Each horizontal line in one image has a corresponding horizontal line in the other image. These two matching lines have the same pixels, with a disparity in the location of the pixels. The process of matching the corresponding lines is called image rectification. The process of stereo correlation finds the matching pixels so that the disparity of each point can be known.

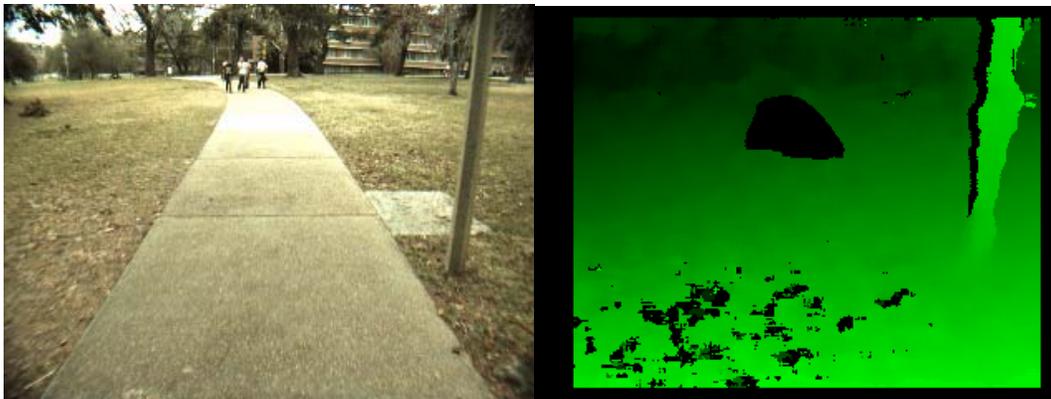


Figure 4.3: Left: Left image in a stereo pair, Right: Disparity image

Figure 4.3 shows the left image in a stereo pair along with its corresponding disparity image. In the disparity image, a brighter shade of green means a larger disparity. Black means that correlation between points could not be performed so there is no disparity found for those points. The pole on the left side of the images can be clearly seen in the disparity image. The bright patch of concrete shows an area that has no features or texture in the original images. No disparity information was found for this section, so the disparity image shows it as black.

The CIMAR stereo vision system is based on the SRI Small Vision System (SVS) [3]. SVS handles the tasks of camera calibration, image rectification, and stereo correlation.

Videre Design stereo vision camera systems were originally used. They consist of two CMOS imagers and an IEEE1394 Firewire interface for transferring the digital images to the computer which does the stereo processing. The Videre STH-MD1-C is shown mounted next to the SICK ladar system on the Center for Intelligent Machines and Robotics NTV2 in Figure 4.4.



Figure 4.4: Videre Design STH-MD1-C and the SICK ladar

Current work involves capturing images with a capture card from two rigidly mounted cameras equipped with auto-iris lenses. In outdoor situations, the amount of light in the environment can not usually be controlled. The lens iris is an adjustable aperture which controls the amount of light that comes through the lens. An auto-iris is controlled by the camera so that when the scene is very bright, the iris automatically closes an appropriate amount. When the scene is dark, the iris opens. This is essential to outdoor mobile robotics because a person cannot be constantly adjusting the iris. If the images are too dark or are washed out, rectification and correlation processes cannot find the matching lines and pixels.

5 TERRAIN SENSORS

5.1 Stationary Ladar

This component uses range data information for terrain mapping and classification. Laser measurement system LMS200 from Sick Inc. is used for range measurement. The sensor operates on the principle of time of flight. A single laser pulse is sent out and reflected by an object surface within the range of the laser. The elapsed time between the emission and reception of the laser pulse serves to calculate the distance between object and the laser. The sensor acquires single line range images. The output of the sensor is available in either a 180° arc or 100° arc. Although three range resolutions are possible (1, 0.5 or 0.25°) the resolution of 0.25° can only be achieved with a 100° arc scan. The accuracy of the laser measurement is +/- 15 mm for a range of 1 to 8 m and +/- 40 mm for a range of 1 to 20 m. For this application the 100° range with a 0.25° resolution is used. A high speed serial interface card is used to achieve a higher baud rate of 500 kBb.

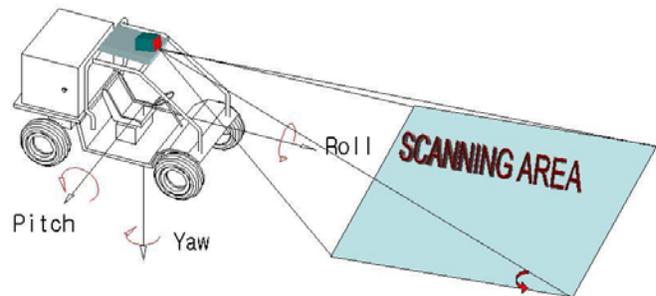


Figure 5.1 Laser Sensor for Terrain classification

Figure 5.1 shows the schematic of the mounting of the laser sensor. The actual mounting of the sensor is shown in Figure 4.4. The sensor is mounted forward facing at an angle of 6° towards the ground. With this set-up, for a flat ground the laser scans the ground at a distance of 20 m ahead of the vehicle. The range data from the laser is converted into Cartesian coordinates local to the vehicle.

The co-ordinate system is attached to the center of the vehicle with the Z axis vertically downwards and the X axis in the direction of the vehicle. The data is then transformed into the global co-ordinate system as referred to in the sensor fusion section above. Each cell in the traversability grid is evaluated individually and classified for its traversability value. The criteria used for classification are,

1. The range of the elevation (height) of the data points within the cell.
2. The slope of the best fitting plane through the data points in each cell. The least square solution technique is used to find the best fitting plane through the data points.

The first criterion is a measure of the spread or dispersion of the elevation of data points in a given cell. The range is given as,

$$\text{Range} = Z_{\max} - Z_{\min}$$

where Z_{\max} and Z_{\min} are the values of maximum and minimum elevation of the data points within the cell.

The second criterion is a measure of the relative orientation of the data points. The equation for the best fitting plane, derived using the least square solution technique is given as

$$S_{\text{optimum}} = (G^T G)^{-1} G^T b \quad (5.1)$$

where

' S_{optimum} ' is the vector perpendicular to the best fitting plane

' G ' is a matrix on 'n' number of points given by,

$$G = \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ - & - & - \\ x_n & y_n & y_n \end{bmatrix} \quad (5.2)$$

' b ' is a length of 'n' vector given by

$$b = \begin{bmatrix} -D_{01} \\ -D_{02} \\ - \\ -D_{0n} \end{bmatrix} \quad (5.3)$$

Assuming D_0 equal to 1, the above equation is used to find S_{optimum} for the data points within each cell. Once we know the vector perpendicular to the best fitting plane. The slope of this plane in the 'x' and 'y' directions can be computed.

The traversability value between 0 and 255 is assigned to each cell, depending on the evaluated slope and range information. Ideally a value of 255 would be assigned to a cell with a horizontal plane and zero range. It is required that a cell contain a minimum of three data points. If the number of data points are less then three the cell is flagged as unknown. This also helps in eliminating noise. Figure 5.2 shows a frame of the mapped terrain from the sensor data and the Figure 5.3 shows the cells which are classified as traversable.

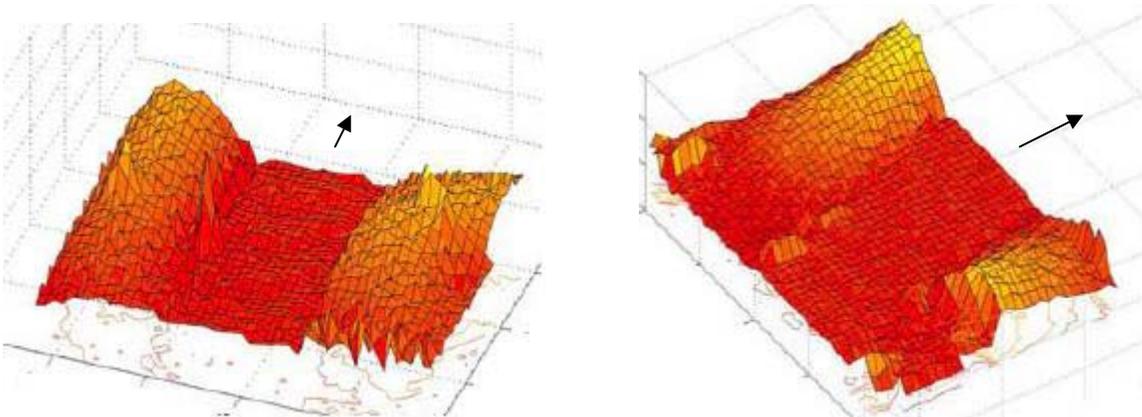


Figure 5.2 Terrain mapping from the sensor data

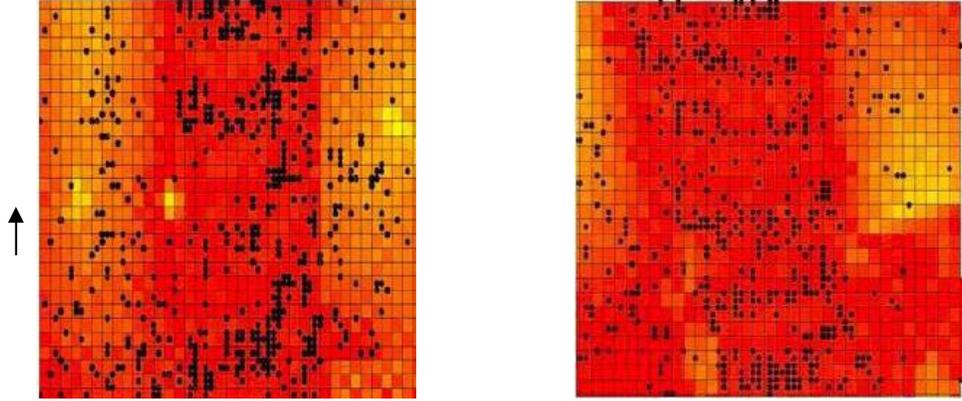


Figure 5.3 Grid classification of the above terrain

5.2 Monocular Vision

A series of three stationary cameras are used to obtain additional information about the terrain. The camera system provides the arbiter with information about the path that the vehicle is currently traveling on. Adaptive color filtering is used to extract the road from the surrounding environment. The color information extracted from the images is used to create a model of the road or path. Then, this model is used to hypothesize the properties of the current road or path. All three cameras are forward facing and provide several perspectives of the environment in front of the vehicle.

A statistical color model was used to differentiate pixels composing the traversable path from the rest of the image. The color model was constructed in RGB color space [4]. The feature vector was defined as

$$\tilde{x} = \begin{bmatrix} r \\ g \\ b \end{bmatrix} \quad (5.4)$$

where r, g, and b are the eight bit color values for a pixel.

A color model of the current path was constructed using by recurrently sub-sampling a region of the image immediately in front of the vehicle. This allowed the color model to easily adapt to a changing path. By using several images to construct the color model, erroneous data was filtered from the model.

The mean color vector is calculated as

$$\tilde{\mu} = \frac{1}{n} \sum \tilde{x} \quad (5.5)$$

where n is the number of pixels in the selected region.

The covariance matrix is then calculated as

$$[\Sigma] = (\tilde{x} - \tilde{\mu})^T (\tilde{x} - \tilde{\mu}). \quad (5.6)$$

The mean and covariance of the color model are then used to develop a classification technique. The equation used for a classification metric is

$$p = \left(\frac{1}{\exp((\tilde{x} - \tilde{\mu})^T \Sigma^{-1} (\tilde{x} - \tilde{\mu}))} \right). \quad (5.7)$$

The classification metric is similar to the likelihood probability except that it lacks the pre-scaling coefficient required by Gaussian distributions. This allows for the classification metric value to be constrained from 0 to 1. The analysis

was performed by selecting a threshold that was a factor of the color model variance in order to classify the traversable path. The algorithm was made more robust by using the uncertainties in the color model in order to establish the current classification threshold.

After the classification was performed the path pixels were translated from image coordinates to local vehicle coordinates. This information was then translated into the northerly raster grid. The update process for the local grid map merges previous and current grid map data and position and orientation information to provide a robust assumption of the road or path.

Results of the algorithm using single images are shown in Figure 5.4.



Figure 5.4 Results of Monocular Vision

6 SUMMARY AND CONCLUSION

This paper has presented the implementation of four different types of sensor technologies for application to unmanned ground vehicle systems. The sensor fusion approach discussed here allows for implementation of multiple sensors for obstacle detection and terrain evaluation. New components that been developed to augment the Joint Architecture for Unmanned Systems (JAUS). To date, testing of these new experimental components has been accomplished at slow speeds. Current efforts are focused on the complete integration and deployment of the sensor system on a vehicle capable of traveling at speeds up to thirty seven miles per hour.

ACKNOWLEDGEMENTS

The work was supported in part by the Air Force Research Laboratory at Tyndall Air Force Base, Florida, contract number F08637-02-C-7022.

REFERENCES

1. Joint Architecture for Unmanned Systems (JAUS) Reference Architecture, Version 3.0, <http://www.jauswg.org>.
2. Carl P. Evans III, "Development of World Modeling Methods for Autonomous Systems Based on the Joint Architecture for Unmanned Systems," *M.S. Thesis*, University of Florida (2004).
3. Konolige, Kurt; Beymer, David; *SRI Small Vision System User's Manual*, SRI International (September 2003)
4. Horn, Berthold Klaus Paul; *Robot Vision (MIT Electrical Engineering and Computer Science Series)*, MIT Press, McGraw-Hill, (1986)