Vision Based Vehicle Localization for Autonomous Navigation

Steven J. Velat, Jaesang Lee, Nicholas Johnson, and Carl D. Crane III

***Abstract*—The vast majority of Autonomous Ground Vehicles in development today operate with GPS based navigation systems. While the accuracy of GPS systems has improved greatly over the previous decade, the stability of their signal has not. The phenomenon is commonly known as "drift" and may have a magnitude of more than a few meters. This paper outlines a method for vision based correction and localization of vehicle position through consideration of a priori information and perceived road characteristics. The approach is called *Vision Based Position Correction for Urban Environments*, and it will be deployed in the 2007 DARPA Urban Challenge.**

I. Introduction

The approach outlined herein addresses the problem of localizing a vehicle's position within a roadway to compensate for large transient fluctuations in GPS position estimates. Many researchers have managed to solve one part of the problem or another through techniques including inertial measurement correction, Kalman filtering [1], and base station correction. However, the real-world problem of safely navigating urban environments introduces additional complications. Issues like satellite occlusion and inertial drift can lead even the best differential positioning systems to have massive discontinuities in their solutions.

In this paper we present methods for vision based lane detection which is then used to correct the vehicle's position in a global reference frame. To do so, several image processing and transformation techniques are used to localize road lane demarcations. The main contribution of this work is to describe the means by which meta-information generated from these demarcations can be combined with a priori information describing a road network to yield a more robust position estimate.

The procedure begins by first capturing a color frame from a camera mounted on the vehicle. This frame is then processed via color segmentation and edge detection to isolate any lane demarcations on the roadway. The resulting edge image is then run through a series of Hough transforms to obtain a geometric representation of the lane demarcations. Finally, the most dominant lines returned from the Hough transform are mapped onto a virtual 2-D plane which represents the area around the vehicle. The lines are then used to estimate the current vehicle lane, its boundaries, and the relative pose of the vehicle within the lane. This information is then compared to that of the a priori information to generate corrections to the road-network. By then projecting the raw GPS estimated position onto the vehicle's lane, an estimate of the error in vehicle position can be generated. This estimated error is then used to create a corrected position estimate which can then be used to map additional sensors.

The impetus for the development of the above method originated from shortcomings identified in the architecture of the University of Florida's 2005 DARPA Grand Challenge vehicle. During the 2005 DARPA Grand Challenge, the University of Florida's vehicle was removed from the race after driving off the graded road and into a thick growth of brush. In the weeks after the conclusion of the race, a detailed case-study was performed on the log data from the day of the race around the time the vehicle was stopped. The result of the analysis showed that a drift in the GPS solution resulted in the traversability data from the various sensors being incorrectly mapped. This error led the vehicle to believe it was no longer on the road and to attempt a course correction, ultimately driving off the road. However, while the drift was initially slow and small in magnitude, the correction was not. As the vehicle was now bogged down in thick brush, it could not re-acquire the road before the mapped sensor data had drifted back. In the previous architecture an a priori pseudo-sensor was employed named the Boundary Smart Sensor. This sensor painted a wide corridor from waypoint to waypoint which represented the divide between in and out of bounds regions. When the GPS solution corrected, the vehicle was left outside of the corridor and thus believed it was out of bounds and could no longer plan a path. Thus, a clear need was defined for a system which can correct for transient errors in position estimation. The following sections outline the new vehicle platform, world representation architecture, and vision based position correction.

II. Vehicle Platform

A. Vehicle

The University of Florida's 2007 team chose a 2006 Toyota Highlander Hybrid. This choice was made primarily due to the hybrid's ability to provide sufficient power for operating the required sensors, actuators and computer systems. Moreover, the Highlander offered the best power, performance, and interface for its price point. The vehicle, named the *N3* (NaviGATOR-III), is equipped with multiple LADAR range finders and an array of 6 high speed cameras, along with multiple differential GPS antennas. Figure 1 depicts the new platform.

Figure 1. University of Florida's 2007 DARPA Urban Challenge vehicle

B. Vision Sensors

The computer vision sensor package for the N3 consists of six USB 2.0 high-speed cameras. The cameras are mounted in various configurations around the vehicle including some actuated locations which allow the cameras to pan. Figure 2 depicts the forward facing camera array.

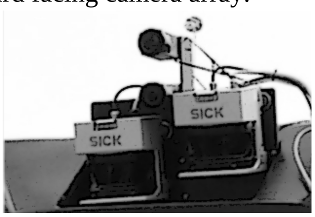


Figure 2. Center mounted camera array for forward lane finding.

Each camera provides color video at rates in excess of 90Hz. at a resolution of 640×300. This resolution is user-configured and was chosen to limit image size and improve overall processing performance.

C. Arbitration Architecture

The basis for the sensing and planning systems of the new platform is a distributed network topology. Multiple Smart Sensors consume raw sensor data, do initial processing on that data, and publish findings in a globally referenced tessellated grid known as a *traversability grid*. The grid representation of the sensor information is then passed to various arbitration and specialist systems over a DOD standard JAUS (Joint Architecture for Unmanned Systems) network [2]. Figure 3 outlines the basic topology of the sense-plan-act architecture.

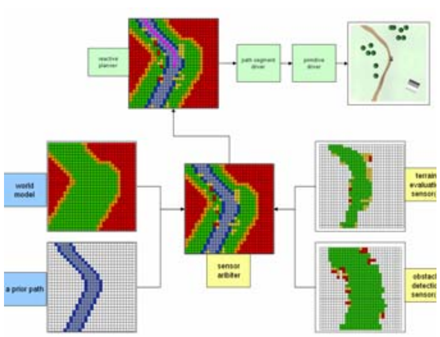


Figure 3. Sense-plan-act architecture.

The sensor information from various sensors is then fused into a cohesive model of the world by overlaying the most recent output of a sensor onto a global grid maintained by the arbitration components. Since each grid is tied to a global reference point grids can be asynchronously fused [3], [4].

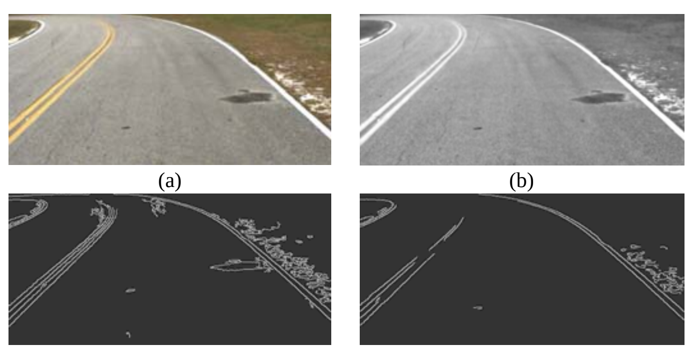
III. Adaptive Lane tracking

A. Introduction

A road is defined by several characteristics. These may include color, shape, texture, edges etc. Because the outdoor environment presents an array of difficulties including dynamic lighting conditions, poor road conditions, and road networks which are not consistent from region to region, it was decided that a multi-tiered approach was needed. Thus, for the targeted application of the Urban Challenge it was decided that a more adaptive lane tracking system would be needed. The proposed method uses two separate approaches to generate a more reliable solution. In the previous challenge, the vehicle was required to operate in an off-road environment where there were no lane demarcations [5], [6]. However, in the current context, the road will be generally paved with lanes defined by painted demarcations. These additional environmental cues allow for the extraction of color and edge information from a given image and to use that information to derive a geometric equation of a line [7].

B. Edge-finding

Edge detection is accomplished by use of the Canny edge detector [8]. The two threshold values that the detector utilizes allow a great deal of tuning and yield a sufficiently segmented image. To further enhance the edge detector's performance, only the red channel of the source image is processed. This channel is used because it has the greatest content in both the yellow and white colors and can thus provide the greatest contrast between yellow/white regions and background. Figure 4 depicts the results of the Canny edge filter with two sets of threshold values.



(a) (b)

(c) (d)

Figure 4. (a) Original road Image. (b) Red Channel Image. (c) Canny filter image with 50 / 200 threshold value. (d) Canny filter image with 130 / 200 threshold value.

C. Color-segmentation

Color segmentation is accomplished by first generating a dynamic set of color threshold values via the K-means clustering algorithm from data extracted from training areas in the source image. The K-means distributions are then used to complete a color segmentation of the source image and finally edge detection is performed on the now segmented image.

1) Training area: The selection of proper training areas is critical. Four training regions are utilized; each consisting of a 320×20 pixel region. Each region is split to extract its red channel and generate a binary image via an adaptive threshold. However, a training area can be contaminated by debris or even lens glare. Such cases are handled by checking the number of valid pixels and contours in the mask does not exceed a threshold. From these binary masks, the RGB content is extracted and used to populate the K-means algorithm. Figure 5 depicts sample training regions and their masks.

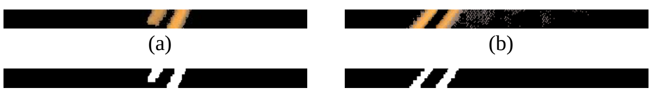


Figure 5. (a) Training area of yellow lane marker without noise. (b) Training area of yellow lane marker with noise. (c),(d) Filtered binary mask

2) K-mean Clustering: The basis of the K-mean algorithm is to minimize the distance between the members of an output set within the space of the entire data set. This process is accomplished by iteratively comparing elements of the set to minimize the distance between the two points. For the color segmentation, this process is used to cluster pixel elements returned from the training areas. The RGB values of the pixels extracted from the training areas are mapped to a Cartesian coordinate frame. The result is a point cloud of pixels as shown in Figure 6.

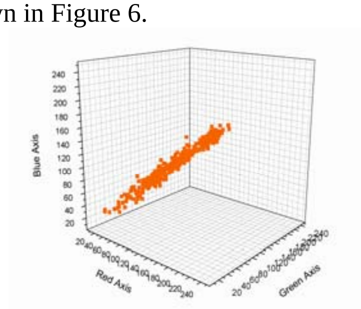


Figure 6. RGB color distribution for white channel K-means.

From the distribution, the min and max RGB values as well as their mean values are determined and used for the color segmentation process. However, due to the limitations of the deployed camera, a given distribution may contain some pixel values which do not represent the desired yellow or white values. Such pixels often appear on the fringe of lane demarcations where the CCD interpolates between the road color and the line color. Figure 7 shows a zoomed region from a training segment with such distortion.

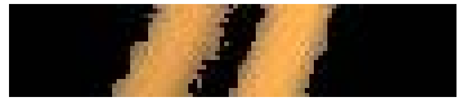


Figure 7. Block effect.

To limit the impact of the fringe values from the training areas, the K-means algorithm is tuned to return only the most tightly grouped regions of continuous color, i.e. the orange/yellow area in the center of the line segments. The K-means algorithm is defined in Equation 1 as:

$$\sum_{j=1}^{k}\sum_{i\in S_j}\left|x^i-\mu_j\right|^2 \tag{1}$$

where x is the RGB pixel vector, μ is the RGB K-mean, k is the number of clusters, and i is the number of pixels. Initial yellow and white pixel values are selected from the original road image. The results of color segmentation using the K-means distributions are presented in Figure 8.

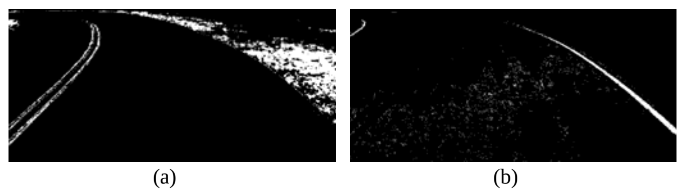


Figure 8. . Color segmentation (a) yellow lane demarcations (b) white lane demarcations.

Wherein Figure 8-a shows considerable noise on the right periphery of the lane, this image is sufficient to isolate the lane region from the image.

D. Hough-transformation

After the edge and color based images are generated, a standard Hough transformation is applied to extract lane demarcations [9]. By sorting the Hough lines found from these images a single set of the strongest Hough lines is generated. Figure 9, located below, shows sample results of this process where the yellow line represents the left boundary, the blue line represents the right boundary, the red line represents the estimated center of the lane, and the green horizontal lines represent the lane width at 5 and 20 meters respectively.

Figure 9. Results of Color based and Edge based lane finding with estimated center line.

E. Lane Estimation

In the real world, road conditions may be such that only one demarcation is visible. Even on roads in good repair with proper striping, the problem of losing a lane reference may occur. This happens in regions such as intersections, cross-roads, or traffic merges. For these instances, an estimation technique is employed to estimate the likely location of the missing lane boundary and the lane center. This is accomplished by using a previous estimated lane width to project where the boundary should be located. In the case where no demarcations are visible the methodology is incapable of yielding a result. Figure 10 depicts sample results of the lane estimation process.

(a) (b)

Figure 10. Lane Estimation (a) yellow lane marker (b) white lane marker

From the Figure, it is clear that the estimation process can effectively determine the location of the missing boundary and is useful when faced with dashed lines.

F. Global map generation

The next step is to project the line segments found using the Hough transform onto the traversability grid. The perspective transformation matrix is calculated from camera calibration parameters and the instantaneous roll, pitch and yaw. The resulting grid provides a wide field of view. However, the projection is subject to gaps in the mapped data which is an artifact of the limited pixels available in distant regions of the image. The gaps depicted above are filled via linear interpolation. Though this process can introduce some error to the grid, the un-interpolated data would otherwise hinder the planning process. Figure 11 depicts an interpolated result and the subsequent painted lanes.

(a) (b)

Figure 11. Traversability Grid. (a) With interpolation and estimated lane center line elements. (b) With painted lanes and boundaries.

The resulting grid contains numeric value which the arbitration component can convert to the appropriate traversability values according to the current vehicle behavior. The additional lanes shown in Figure 11 are created by estimating the distance in pixels between the lane boundaries. This estimated lane width is then used to draw both the center, right and left lanes. It is important to note that the left and right lanes are drawn whether or not they exist. This allows the arbitration component to adjust their traversability value according to a priori information on the number of lanes and the lane the vehicle currently occupies.

G. Meta-Data Generation

While the traversability grid representation is the end result of the image processing steps outlined above, additional information is generated along the way which makes position correction possible. Information such as lane width combined with the estimated center of the current lane is used to determine the vehicle pose within the lane. This estimated pose includes both the lateral offset from the center of the lane as determined from the distance in pixels from the center of the image to the estimated lane center. Moreover, the angular orientation in the lane is derived by comparing the instantaneous heading of the vehicle to the heading of the lane center. The result is a full description of how the vehicle is posed within the perceived lane.

IV. CORRECTION OF ESTIMATED POSITION

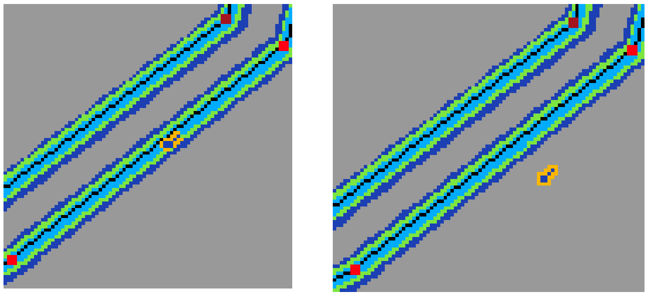
A. Introduction

For the 2007 DARPA Urban Challenge, DARPA has specified a standardized model for representing the world in a vector format. This format is known as an RNDF (Road Network Definition File). The file, generated in advance of an autonomous run, contains surveyed GPS waypoints for various lanes and roads which make up a larger network. Each *road* is composed of one or more *segments* which contain two or more waypoints. Each segment represents a lane on a given road and contains additional information such as lane borders (single/double yellow, white, dashed-white, etc.) as well as which waypoints correspond to turns and intersections. Moreover, DARPA has also specified a MDF or Mission Data File. This file contains the series of goal waypoints the vehicle must achieve and the order in which to achieve them.

B. The World Vector Driver

The architecture developed for the N3 is multi-tiered. First, a high-level planning component generates a course routing through the road network defined in the RNDF. The high-level-plan is then provided to a lower level component that is responsible for reconciling the high-level-plan (HLP) with the vision-based lane data. This component, named the World Vector Driver (WVD), generates a set of GPS waypoints which represent the immediate goals of the vehicle. A sample depiction of the output of the WVD is given in Figure 12.

The figure shows how the WVD generates straight line segments between goal waypoints along the mission. Additionally, the WVD paints a virtual lane with a discounted



(a) (b)

Figure 12. WVD a priori lanes (a) accurate GPS (b) with drift

cost in the center of the lane. The vehicle is shown as the orange and blue box in the center of the grid. However, if the GPS solution undergoes a drift, the WVD may generate a grid much like that of Figure 13 (b). Such a drift may result in the vehicle planning an unnecessary correction to re-acquire the lane. For this reason, it is necessary to correct the a priori RNDF and the GPS position.

C. *Globally Referenced Data*

The N3's architecture depends on spatial transformations to asynchronously fuse data from numerous sensors into the same common reference frame. This is only possible if each sensor's output is tied to a globally referenced point. To do so, the GPS position in Latitude and Longitude is converted to Northing and Easting or Universal Transverse Mercator (UTM) which is based on the WGS84 standard. This process means that each bit of sensor information is mapped relative to a fixed point, and thus can be retained within the world model and rolled as the vehicle traverses a given area. The process for mapping sensor information is outlined in Figure 13 below.

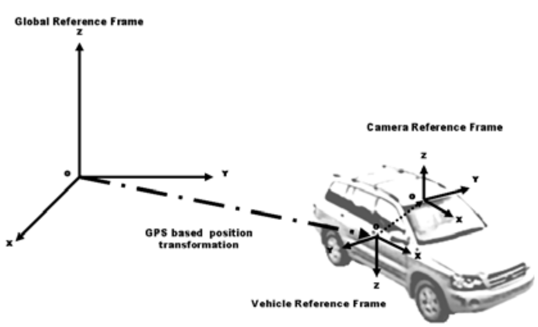


Figure 13. Process of Mapping Sensor Data to the Global Reference Frame.

By utilizing a series of transformation matrices, data is mapped from the sensor coordinate frame to the vehicle coordinate frame and subsequently to the global reference frame by multiplying by a transformation matrix. The generic homogenized transformation is defined as:

$$\begin{bmatrix} \cos\gamma\cos\beta & -\sin\gamma\cos\beta & \sin\beta & x \\ \cos\gamma\sin\beta+\sin\gamma\cos\alpha & \cos\gamma\cos\alpha-\sin\gamma\sin\beta\sin\alpha & -\cos\beta\sin\alpha & y \\ \sin\gamma\sin\alpha-\cos\gamma\sin\beta\cos\alpha & \sin\gamma\sin\beta\cos\alpha+\sin\alpha\cos\gamma & \cos\beta\cos\alpha & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

where α ,β, and γ represent rotations about the x, y, and z axis respectively and x, y, and z represent the translation along the x, y, and z axis respectively. In general, the transformation from the vehicle to the global reference frame is generated solely from GPS and inertial measurement information.

D. *Correction Methodology*

The GPS correction is generated by first determining the current occupied lane. Next, the a priori road-network is augmented when needed to adjust for angular offsets in the perceived road and the a priori straight lane segment. Third, the raw GPS solution is projected on to the a priori lane-center normal to the GPS solution. Finally, the projected position is offset position to account for the vehicle's offset (if any) from the center of the lane. The above procedures only address the issue of error in the direction normal to the heading of the vehicle. This limitation is a result of the lack of a fixed reference point in the image which can be correlated to an a priori GPS location. However, in the case where a reference point does exist (such as the stop-line at an intersection) the error in the direction of the vehicle's heading can be corrected. Through much the same process of recognition, comparison, and correction.

E. *Lane Determination*

The process of determining the occupied lane is a simple comparison and look-up process. The WVD receives information from the lane-finder stating the type of lane boundaries sensed, and the relative position of the vehicle within the boundaries. By comparing this information with that stored in the RNDF for the current road the vehicle is on, the WVD is able to ascertain the most likely lane the vehicle occupies.

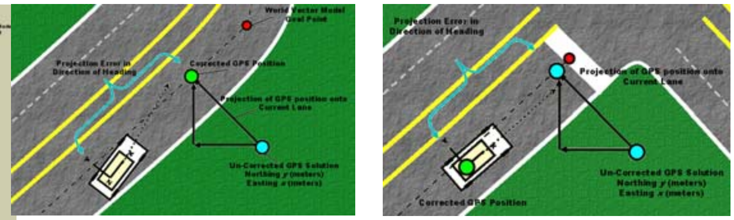
F. *World Vector Driver RNDF Correction*

The information contained within the RNDF must be considered sparse. This means that the data provided to define the road network may contain waypoints that are separated by vast distances. As such, it cannot be guaranteed that the road between two consecutive waypoints is a straight line. Thus, to make use of the a priori road-network information, and to improve the quality of its content, it is necessary to generate dynamic corrections to the vector road representation. The correction to the WVD's world representation is determined by calculating the error between the estimated lane heading and the straight-line heading of the a priori lane the vehicle currently occupies. Such a scenario is depicted in Figure 14 below.

If the angular error between the sensed lane-center and the a priori lane-center is greater than a threshold value, a series of *bread crumb* waypoints are generated. The bread crumbs are dropped both in front of and behind the vehicle along the heading of the perceived lane-center. These points are then used to re-generate the WVD representation of the world with the additional waypoints to provide a more accurate contour

(a) (b)

Figure 14. A priori lane correction (a) Initial a priori lanes (b) lanes after the addition of *bread-crumbs*



(a) (b)

Figure 15. Lateral GPS Offset. (a) Without fixed reference, lateral only (b) With fixed reference lateral and longitudinal

of the road. Figure 14 depicts the additional waypoints generated and their effect on the contour of the WVD road network

With these additional waypoints interspersed between original RNDF waypoints, the road becomes more fully defined. Moreover, this additional information is retained for future reference if and when the same segment is traversed.

G. Lateral Correction

The lateral correction of the GPS solution is accomplished by projecting the raw GPS position onto the corrected WVD lane. This projection may result in offsets in either or both the Northing and Easting directions. The normal used for the projection is determined as shown in equation 3 below.

$$\underline{P} = \underline{S} \times \left(\underline{S}_{OL} - \underline{a} \times \underline{S} \right)$$
$$P_{norm} = \underline{P} + \underline{a} \qquad (3)$$

Where $\underline{P}$ represents the location of the point along the lane normal with respect to the GPS solution, $\underline{S}$ represents the direction of the lane center, $\underline{S}_{OL}$ represents the moment of the lance center about the global origin, $\underline{a}$ is the initial solution of the GPS position estimate and P_{norm} represents the global location of the projected GPS location. By computing the normal to the lane and then translating the position estimate to this point we can in effect snap the GPS estimate to the current lane.

The northing and easting components of the correction are then calculated as the sine and cosine of the angle of the vector $\underline{S}_a^P$ respectively. Finally, we can further improve the estimate by again considering a lateral offset representative of the vehicles relative pose within the lane as perceived by the lane finder. Figure 15 (a) depicts the lateral correction process.

From the figure, it is apparent that the lateral correction is sufficient to localize the vehicle within a given road-network. This is because while there remains an error in the direction of the vehicle's heading, its magnitude is insignificant relative to the overall length of the roadway. However, this error can also be minimized when a fixed surveyed reference exists within the cameras visual range. Such a reference might be a stop-line at an intersection which is part of the RNDF surveyed road-network. As Figure 15 (b) shows, the error in the direction of travel can then be minimized by first completing the lateral corrections and then comparing any offset error between the perceived location of the stop-line with the a priori data.

V. Conclusions

The lane finding and pose estimation algorithms presented here have been developed for implementation on the University of Florida 2007 DAPRA Urban Challenge vehicle. This work addresses the problem of geo-referencing sensed and a priori information in order to achieve autonomous navigation in an urban environment under conditions where GPS data is subject to errors. This is a significant problem that must be addressed for successful realization of such systems.

VI. Acknowledgements

The authors would like to gratefully acknowledge the support of Smiths Aerospace, the Air Force Research Laboratory, and the University of Florida.

VII. References

[1] Sukkarieh, S. Nebot, E.M. Durrant-Whyte, H.F., "A high integrity IMU/GPS navigation loop for autonomous landvehicle applications" Robotics and Automation, IEEE Transactions on, Jun 1999, Vol.15,Issue.3, P572-578

[2] JAUS. (2006). Joint Architecture for Unmanned Systems Reference Architecture, version 3.2: JAUS Working Group (http://www.jauswg.org/)

[3] Crane, C., Armstrong, D., Touchton, R., Galluzzo, T., Solanki, S., Lee, J., Kent, D., Ahmed, M., Montane, R., Ridgeway, S., Velat, S., Garcia, G., Griffis, M., Gray, S., Washburn, J., "Team CIMAR's NaviGATOR: An Unmanned Ground Vehicle for Application to the 2005 DARPA Grand Challenge," Journal of Field Robotics, 2006

[4] Touchton, R., et. al, "Planning and modeling extensions to the Joint Architecture for Unmanned Systems (JAUS) for application to unmanned ground vehicles," SPIE Defense and Security Symposium, Orlando, Mar 2005.

[5] Lee, J., Crane, C., Kim, S., and Kim, J., "Road Following in an Unstructured Desert Environment using Monocular Color Vision as Applied to the DARPA Grand Challenge," Proceedings of the International Conference on Control, Automation, and Systems (ICCAS 2005), Gyeong Gi, Korea, June 2005

[6] Jaesang Lee and Carl D. Crane III1, "Road Following in an Unstructured Desert Environment Based on the EM(Expectation-Maximization) Algorithm", ICCAS 2006, Busan, Korea

[7] Apostoloff, N. Zelinsky, A.,"Robust vision based lane tracking using multiple cues and particle filtering",Intelligent Vehicles Symposium, 2003. Proceedings. IEEE, 9-11 June 2003, P558- 563

[8] J Canny, "A computational approach to edge detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 8 , Issue 6(Nov 1986), P679 – 698

[9] B. Yu, A. Jain, "Lane Boundary Detection Using a Multiresolution Hough Transform", 1997 International Conference on Image Processing (ICIP'97) - Volume 2 p. 748