Articulated Unmanned Ground Vehicles (UGVs) Controlled with Complimentary Semi- & Autonomous Mobility Behaviors

Mr. Gerald R. Lane
Applied Research Associates Inc. (ARA)
48320 Harbor Dr
Chesterfield TWP MI 48047
glane@ara.com

Dr. Charles Reinholtz PhD.
Virginia Technological University
Blacksburg, Virginia, 24061
creinhol@vt.edu

Dr. Mohan Krishnan PhD.
University of Detroit Mercy
ECE Department
4001 W. McNichols Road, Detroit, MI 48221
mohank@udmercy.edu

Dr. Mark J. Paulik PhD.
University of Detroit Mercy
ECE Department
4001 W. McNichols Road, Detroit, MI 48221
paulikmj@udmercy.edu

Dr. Nassif Rayess PhD.
University of Detroit Mercy
ECE Department
4001 W. McNichols Road, Detroit, MI 48221
rayesna@udmercy.edu

ABSTRACT

This paper will discuss and provide detailed descriptions of unique configurations of small articulated unmanned systems and describe the design of several approaches resulting in substantial increases in autonomous/semi-autonomous mobility and resultant survivability. The greater mobility results from unique chassis configurations which do not require the space claim for human operators. Each human occupant can consume 2 cubic meters of volume. This volume reduction results in a corresponding weight reduction also contributing to mobility enhancements. The UGVs can be designed with less stringent suspension

requirements which are no longer needed for human comfort or performance. UGVs still require suspension for stability of sensors and ride performance but it is greatly reduced from 2.5Gs for human operation to up to 5Gs for unmanned operation. Novel techniques in chassis design are much easier to implement without the required crew compartment. This will be shown in the articulated and polymorphic configurations.

1.0   ARTICULATION BACKGROUND

The US Army has fielded several versions of articulated vehicles and has also experimented with high performance articulated test beds. In the 1960s, the US Army fielded two articulated tactical vehicles: the Gamma-Goat and the GOER. The Gamma-Goat light duty one ton carry-all utility vehicle was liked by the troops for its agility but was removed from duty due to its high logistical support requirements. The GOER, in the 10 ton class, was modelled from the earth mover design, very efficient and stable at low speeds. After fielding, it was determined that the tires, providing the only aspect of the suspension, would not give the high speeds required for effective military transport.

Figure 1: US Army's articulated Small Unit Support Vehicle (SUSV)

In the late 1960s and early 1970s, the US Army TACOM developed two advanced test beds to further assess the military potential of articulation. The Lockheed Twister, a two chassis platform each with four wheels and powered by two separate engines, allowed the army to assess the advantages of articulation coupled with a high HP/Ton ratio. Another test assessed how two highly mobile tracked vehicles (M113s) coupled through articulation could negotiate battlefield obstacles designed to stop combat tanks. The Army has the articulated Small Unit Support Vehicle (SUSV) currently in its fleet shown in Figure 1. It has been fielded on a limited basis, primarily to units assigned to winter environments requiring high snow flotation.

All the Army production and tested systems were subject to the limitations and logistics burdens of mechanical articulation enabled through multiple hydraulic cylinders and mechanical linkages for steering coordination.

The US Army Tank-automotive Research Development & Engineering Center (TARDEC) working closely with the Jet Propulsion Laboratory utilized the JPL unmanned lunar rover prototypes built by General Motors Delco Electronics for advanced robotics mobility development entitled Computer Aided Remote Driving (CARD). The articulated rover shown in Figure 2 provided greater mobility characteristics over terrain obstacles while significant reductions in turning

**Figure 2: JPL Lunar Rover with
TARDEC Computer Aided Remote Driving**

radii for negotiation of tight spaced obstacles in the rover's path.

Today, UGVs are not limited by hydraulically powered joints or mechanical steering linkages. Torque motors, hybrid electric drive with computer controlled differential steer, enable unique configured demonstrators equipped with autonomous mobility behaviours to exhibit very agile behavior.

There have been several sources of novel UGV design. The DARPA Tactical Mobile Robotics Program underscored the need for small robots to support dismounted, security and law enforcement operations. A second source of novel UGVs was the Intelligent Ground Vehicle Competition (IGVC), founded and sponsored by the Association for Unmanned Vehicle Systems International (AUVSI).

Universities, after several years of competing, determined that a compliant /twisting chassis could more easily negotiate the complex obstacles in the course and achieve the highest performance speeds. In 1997, the University of Colorado-Boulder (UCB) developed and competed with an articulated vehicle with differential steer. UCB integration of autonomous mobility algorithms into the articulated robot resulted in very agile obstacle negotiation. In 2001, Virginia Tech built its first of two articulated robots which have shown remarkable autonomous performance at the last several IGVC events. In 2006, University of Detroit Mercy entered an articulated platform with multiple modes of machine vision selected to operate as vision environments changed.

Figure 3: University of Colorado Boulder Robotic Autonomous Transport (RAT)

2.0 VIRGINIA TECH GEMINI ARTICULATED VEHICLE DECRIPTION

Virginia Tech has produced a number of small autonomous ground vehicles for student competitions and research. One of the most successful of these has been a vehicle named Gemini, which alludes to the twin bodies reminiscent of the well-known constellation of the same name. Gemini has proven to be an enduring competitor in the IGVC. In its first year of competition, Gemini finished 2nd overall. Using redesigned software, Gemini finished first overall in both the 2005 and 2006 Intelligent Ground Vehicle Competition. Gemini exhibits the stability of a four wheel vehicle with the agility of a three wheel differentially steered vehicle.

Gemini, shown in Figure 4, is a two-body platform that provides zero-radius turning capability of its front body along with exceptional stability on uneven terrain. The keys to this enhanced performance lie in the drive system and in the two-degree-of-freedom joint connecting the front and rear chassis. Independent motors drive the two front wheels, and a vertical axis of rotation between the body sections allows the front platform to pivot relative to the rear platform. The steering pivot point is centered directly between the front wheels,

Figure 4: Gemini in a Turn

allowing the front platform to make a zero radius turn about its center without moving the rear platform. A second, longitudinal, axis of rotation between the body sections allows the front body to roll relative to the rear body. This allows Gemini to keep all four wheels on the ground on extremely uneven terrain without the complications associated with suspension. In essence, the rear platform follows the front platform like a trailer, but with one important difference. Since there is no joint to allow pitching motion between the front and rear, the rear platform gives the front platform stability in the fore and aft direction, which allows the front "towing" platform to have only two wheels. Steering is accomplished by driving the two front wheels at different speeds, just as with any other differentially steered vehicle. The weight distribution of 60/40 favors the front drive wheels to provide increase traction.

2.1 Gemini originally designed for IGVC

Although Gemini has been used extensively as a research and development platform, it was originally designed by a student team to compete in the 2004 Intelligent Ground Vehicle Competition (IGVC). In recent years, the IGVC has included three main competition events, namely, the Autonomous Challenge, the Navigation Challenge, and the Design Competition. The Autonomous Challenge and Navigation Challenge are functional events where the vehicles operate independent of human interaction. The Design Competition requires teams to conduct and document a rigorous design process ultimately presenting the vehicle design in a written report and formal oral presentation.

The objective of the Autonomous Challenge is for a fully autonomous ground vehicle to negotiate an outdoor obstacle course bounded by solid and dashed lines. The course must be completed within five minutes while staying within a defined 5 mph speed limit and avoiding obstacles on the track [1]. Performance is judged based on speed and ability to negotiate through the complex terrain without contacting obstacles or boundaries.

The objective of the Navigation challenge is to autonomously travel from a starting point to a number of target destinations (waypoints or landmarks) and return to the initial starting position, given only a map of the target position coordinates. The performance is based upon speed and accuracy.

The Design competition is focused on the evaluation of the design process that the design teams followed to create their vehicle. The vehicle design judging is performed by a panel of expert judges familiar with the field of autonomous unmanned vehicle systems. Teams produce formal written and oral presentation. The final scoring for the overall award is based on weighted and combined scoring from the three events.

2.2 Gemini's Sensing and Control Architecture

Four separate sensing devices enable Gemini to navigate in the context of its surroundings. These devices are listed and described in Table 1. The camera and laser rangefinder are employed in the Autonomous Challenge portion of the competition to locate the pathway boundaries and various physical obstacles. The digital compass and GPS allow Gemini to determine its heading and location while the laser rangefinder provides data on the position of obstacles during the Navigation Challenge.

Table 1: Summary of the Four External Sensors Used in Competition.

Device	Function
Digital Firewire Camera	Determines pathway boundary and negative obstacles
SICK LMS-221 Laser Range Finder	Detects obstacles
PNI TCM2-20 digital compass	Determine vehicle heading
Novatel Differential Global Positioning System (GPS)	Determine absolute global location

Gemini uses separate software algorithms for the Autonomous Challenge and the Navigation challenge. Both algorithms were developed using National Instrument's LabVIEW software. Gemini's onboard computer is a Pentium 4 class, 3.2 GHz, notebook that resides on the rear section of the vehicle frame. The computer communicates with the motors and sensors by either USB, through a serial to USB converter, or an IEEE 1394 firewire port. The operating system used on the machine is Microsoft Windows XP professional.

The intelligent navigation software that operates Gemini is preloaded on the onboard laptop computer prior to deployment. The software developed to accomplish the autonomous lane following challenge utilizes a digital Firewire camera for lane detection and LIDAR for obstacle avoidance. The algorithm combines the sensor inputs in order to generate the best forward path. The general approach for the autonomous challenge is detailed in Figure 5. The process cycles continuously at about 10 hertz during navigation, adjusting the vehicle heading and speed to effectively navigate the course until it is commanded to stop.
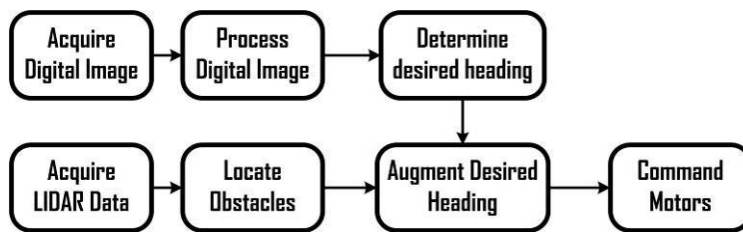

Figure 5: Autonomous challenge algorithm.

The image processing method is detailed in Figure 6. The acquired image is pre-processed by extracting specific image values from the composite RGB color image. To reduce processing time, the image is down sampled from the 640 by 480 pixel native camera resolution to 160 by 120 pixels and converted to grayscale. The image is then split in two, representing the view to the left and to the right of the vehicle. To determine the strongest linear relationship in the images, a Hough transform is used and the dominant line occurring in each half of the image is identified. This method works equally well for solid or dashed lines. A decision tree is implemented to determine a vehicle heading based upon situational line detection cases. The obstacle avoidance capability subsumes the vision derived heading. The final vehicle heading, being the composite of the vision and obstacle avoidance data, is then used to command the motors.
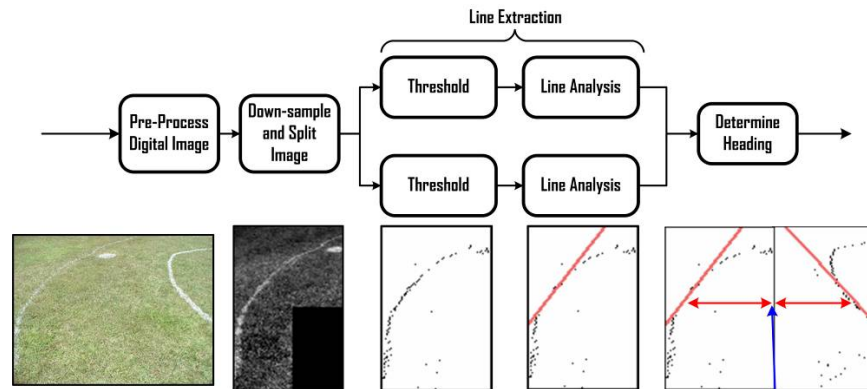
Figure 6: Image processing algorithm for boundary line extraction.

2.3 All-Wheel Drive Capability and Adaptive Vehicle Geometry

In the configuration described above, where only the forward wheels of a two-bodied articulated vehicle are powered, it is important to distribute the majority of the vehicle weight over the drive wheels to produce sufficient traction. Since each wheel of the articulated vehicle is separately powered, it is possible to power all four to produce maximum traction. Such a four-wheel drive configuration may also improve weight distribution, payload capacity and stability.

A kinematic model and computer simulation of the articulated vehicle has been developed to aid in the design of a four-wheel drive variation of the platform. Figure 7 shows a plot of successive positions of the front and rear body segments of an articulated vehicle as the vehicle enters a turn. In this figure, the vehicle starts at the bottom and moves upward and to the left. Red represents the right wheels and blue represent the left wheels. The crosses represent front wheel positions and the circles represent rear wheel positions. The vehicle starts at the bottom of the figure with the front and rear sections aligned.

In general, the rear wheels do not follow the same track as the front wheels. The velocities of the rear wheels will be dependent on the velocities of the two front wheels and the position of the steering joint between the two sections. By measuring the position of this steering joint with an encoder, we have been able to develop a set of compatibility relationships that allow us to power both rear wheels. The motion of the platform is controlled by commanding the front section wheel velocities as if the vehicle were a simple differentially driven vehicle, while the rear wheels are commanded to drive so that their motion matches the kinematic constraints imposed by the instantaneous vehicle position. Such a control scheme might seem complex for a manned vehicle, but adds relatively little computational and control complexity to an autonomous vehicle.

A second issue that is often a concern when dealing with multibody vehicles is tracking. As noted earlier, the rear section does not track the front section except when the vehicle is going straight. Typically the rear wheels track to the inside of the front wheel in a turn, clipping the corner. This is clearly true for the geometric configuration described so far in this paper, where the steering joint axis intersects a line through the front wheel drive axes. It is possible, however, to adapt the geometry of the vehicle to prevent the rear

body from cutting corners and potentially colliding with obstacles.

One method of achieving this adaptive geometry is to connect the front and rear sections with a four-bar linkage, as shown in Figure 8. Using the dimensions shown in the figure, the rear body will swing a wider arc than the front body entering a turn. This feature greatly reduces the possibility of the vehicle hitting an object it is attempting to go around.

It is a relatively easy matter to adjust the linkage pivot locations to adapt the path of the rear body to the particular driving conditions. For example, when the vehicle is passing an obstacle on the inside of a turn, it will adapt the geometry of the linkage to prevent the rear body from cutting the corner and clipping the obstacle with its rear wheels. On the other hand, when the vehicle is passing an obstacle that is on the outside of a turn, it will adapt the geometry of the linkage to guide the rear body on a tighter radius turn. Such adaptive behavior might be difficult to implement in a manned vehicle, but it is a natural extension of the usual object mapping and path planning associated with autonomous vehicles.
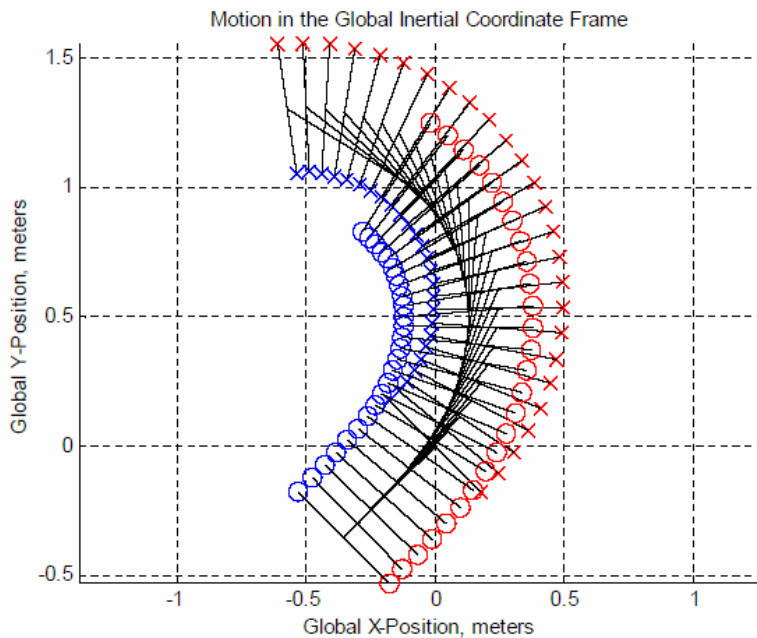

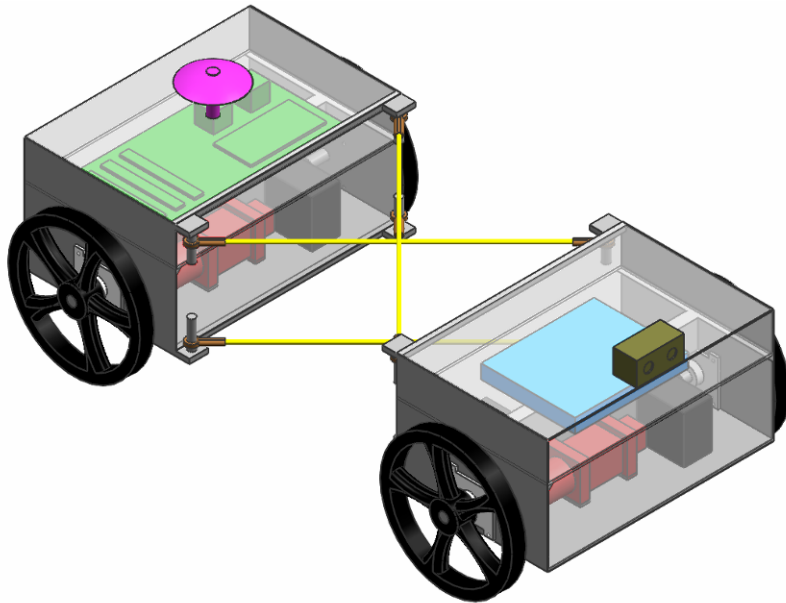**Figure 7: Front and Rear Body Tracking**

Figure 8: Adjustable Linkage-Based Articulated Vehicle

3.0 UNIVERSITY OF DETROIT MERCY ARTICULATION APPROACH

3.1   Description of Vehicle Architecture

The choice of vehicle architecture is perhaps the highest value strategic decision. It affords the flexibility that largely determines the ability of the vehicle to carry out the decisions of the autonomous controller. In essence, what the vehicle is able to perform physically either complements or negates its autonomous functions. That said flexibility is a double edged sword. There is a happy medium beyond which the benefits of added flexibility will be outweighed by the excessive demands on the motion controller. The experience of the UDM team showed that this happy medium is the four-wheeled two-body articulated vehicle with a two-degree of freedom hitch connecting the front and back halves as shown in Figure 9. The rear wheels are free-wheeling while the front wheels are driven.
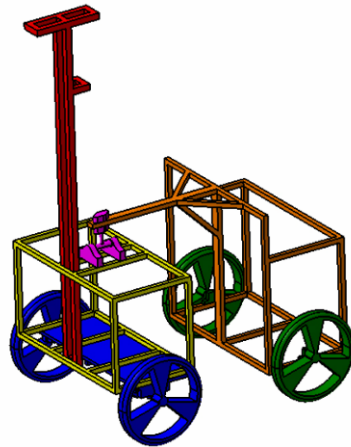
Figure 9: Design rendering and finished vehicle

This architecture affords a great many functional and performance benefits. The camera and laser ranging unit that are mounted on the front part of the vehicle allow the vehicle to "look around" without causing any sideways motion that can result in sideswiping obstacles. This functionality can be obtained with a separate complex panning sensor package, but is built into this design. The vehicle is very stable on uneven terrain and is able to "snake" around obstacles, an ability that simplifies the demands on the navigation algorithms. Other functional benefits include the added packaging room and the physical separation of the two bodies; the latter allows for segregation of sensitive equipment from rugged and noisy ones, e.g. the electric generator. On the other hand, the disadvantage of the two-body architecture is that it is not easy to reverse the vehicle, an ability that could come in handy when it navigates into a trap. Thus the near zero turning radius of the vehicle has to be used instead to exit from this situation. But all in all, the positives outweigh this negative.

As indicated, the vehicle architecture and design is one of three main requirements for a successful autonomous vehicle. The other two are the image processing algorithms and the navigation strategy, and these are discussed below.

3.2   Vision Algorithms

The fundamental requirement of the image processing system (IPS) is to provide lane, terrain, and obstacle information to the vehicle navigation system. The algorithms used have been discussed in detail in [2]; what follows is a briefer discussion. The vision strategy implemented for THOR is based on parallel analysis, in that it processes images with three distinct algorithms. Individual results are combined to generate both a set of lane coordinates as well as a confidence measure. These are then fed into a parallel behavioural-programming-based navigation program. This unique approach has been designed to take advantage of the individual strengths of statistical, region-based, and derivative-based image processing techniques. Furthermore, the use of confidence measures allows for the possible incorporation of powerful fuzzy inference techniques in the navigation algorithm.

The first algorithm uses a row-adaptive statistical filter to establish an intensity floor, followed by a global threshold based on a reverse cumulative intensity histogram and *a priori* knowledge about lane thickness and separation. The second method first improves the contrast of the image by implementing an arithmetic combination of the blue plane (RGB format) and a modified saturation plane (HSI format). A

global threshold is then applied based on the mean of the intensity image and a user-defined offset. The third method applies the horizontal component of the Sobel mask to a modified gray scale image, followed by a thresholding method similar to the one used in the second method.

The Hough transform is applied to each of the resulting binary images to select the most probable line candidates. Finally, a heuristics-based confidence interval is determined, and the results sent on to a navigation module, which fuses the image data with the obstacle data from the laser scanner to make a driving decision.

Combining three different methods of lane line identification provides redundancy and capitalizes on the strengths of each method, while mitigating their weaknesses. The main drawback is the substantial processing time required. However, this difficulty can be overcome to some degree by code optimization.

3.2.1 First Algorithm: Adaptive Statistical Filter

An adaptive statistical filter (hereafter referred to as the "low filter") was developed to enhance lane contrast. The purpose of the low filter is to drop out pixels that are considered too low in value based on statistical information. The low filter works row by row, and deletes pixels with intensities less than a threshold determined by a weighted combination of the local mean and standard deviation. Pixels above the threshold are unaltered. This filter is applied independently on all three color planes since the lines are assumed to be white or close to white in intensity.

The low filter takes out the majority of pixels considered to be noise in rows with lines, while at the same time it does not increase the noise in rows without lines as seen in 10ure 9. It also normalizes the illumination because it capitalizes on the local statistical properties of the image. With the improved line contrast produced by this filter, a more traditional low-pass or localized order-statistical filter is far more effective in attenuating noise. The final binary image produced is shown in Figure 11.

10. 9 (a): Pre-processed Image. (b): Image after low filter.

Fig. 11: Image after threshold

3.2.2 Second Algorithm: Cross-normalization

For the second method, a modified saturation plane (from an HSI image) was used to enhance the contrast

of the blue plane of an RGB image. White lines have a low saturation, while the green grass has a higher saturation. Also, in the intensity plane, white lines have a higher intensity in general than the green grass. The inverse of the saturation plane therefore yields an image with strong lane lines. Additionally, a modification of the equation used to compute the saturation plane from R, G, and B values was used to produce a higher contrast and a more uniform image. Specifically, the traditional saturation equation was modified by removing the normalization term in the denominator. This modification tended, for example, to cause deeply saturated green and modestly saturated green pixels to take on numerically closer values (variations can be caused by illumination, shadows, and variations in foliage density). Such improved uniformity made the application of a threshold more effective. Figure 12 illustrates a standard and a non-normalized inverted saturation image.

This inverted and modified saturation plane was then multiplied by the blue plane (which tends to represent painted lines on green grass better than a simple intensity image), thus further enhancing the contrast of the lines. Figure 12a illustrates the result of combining the blue plane and the modified saturation image.
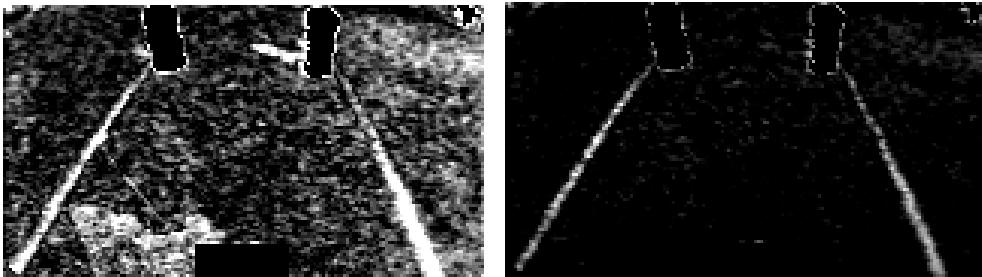

Fig. 12 (a): True saturation plane of image. (b): Modified saturation plane.

A threshold was then applied to this contrast-enhanced image using the mean of the non-zero pixels in the image plus a user defined constant (selected as a low dropout threshold). Figure 13b presents the results of the thresholding operation.
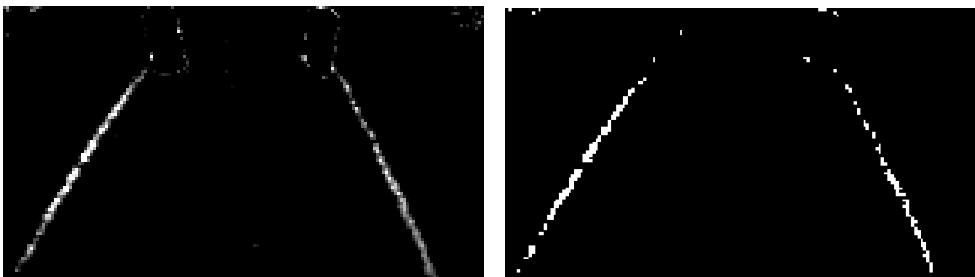

Fig. 13 (a): Modified saturation plane dot-multiplied by the blue plane of the image. (b): After a mean based threshold.

It is interesting to note that a general contrast improvement for a gray-level intensity image can be produced by exploiting the natural high lane contrast found in the Blue plane (due to low blue content in grass), and the correspondingly low lane contrast found in the Green plane (due to green grass showing through the painted white lines). Such an approach was described in [3] and found to be valuable in cases where lane lines are painted lightly or faded. This approach subtracted the green plane from a scaled blue

plane (2B-G was used) to form a higher contrast intensity plane, which could be used in any algorithms which work on gray-scale images. This technique demonstrates that the exploitation of *a priori* knowledge about the likely images for a specific application is, in general, an effective image processing technique.

3.2.3 Third Algorithm: Edge Detection

The third method used a time-honored edge-detection method for finding the edges of the lanes. Using a standard horizontal Sobel mask on an intensity plane, the edges of the lanes were found. Note that it is also possible to use a contrast enhanced image (such as the 2*Blue-Green or Blue Plane alone) as a post processing mask to remove noise. After the horizontal Sobel mask application, a threshold based on the mean of the non-zero image elements was applied. Figure 14 shows the Sobel-processed, contrast image masking, and thresholded image results.

Fig.14(a): Image after the Sobel mask applied to the blue plane. (b): Sobel.*blue. (c):After threshold.

3.2.4 Hough Transform Post-Processing

The Hough transform is applied to the images resulting from the application of each of the methods described above. It effectively filters out all pixels which are not co-linear and all pixels which are co-linear but do not occur in sufficient numbers. A cutoff threshold was applied to the Hough accumulator to ensure false lines were not read (this value is adjustable). In addition to a pixel-total cutoff threshold, line angles were used to delete lines that had angles which were unlikely to occur in this application.

3.2.5 Confidence Measure

A confidence measure was developed based on the correspondence found between the results from the three processing methods. A separate confidence measure was developed for the right and left lines. The program first checked the right and left sides to see if all three methods agree. If so, a confidence of 3 is sent to the navigation program. If all three methods do not agree, the program checks to see if two out of three methods for each line agree. If so, the angles of the lines are then checked to make sure there is not a cross over of two lines with greatly different angles, which can result from a patch of noise. If all is well, a confidence of 2 is assigned. If none of the methods agree, the program looks for the highest Hough accumulator value, and if that value exceeds a floor threshold, the line from that method is used and a confidence of 1 is assigned. The resulting images for two levels of confidence for the previous example are shown in Figure 15.
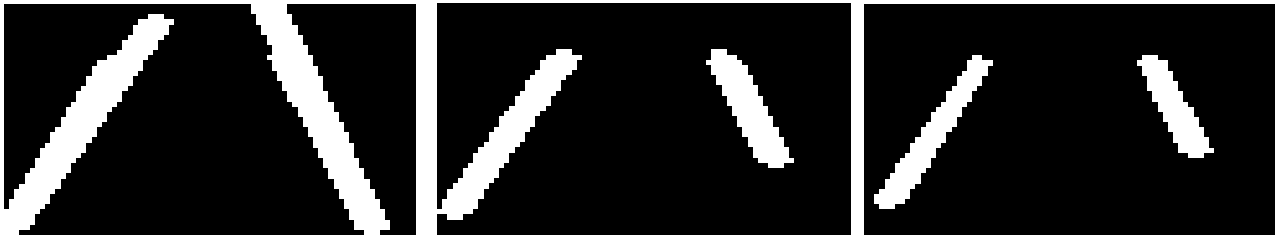
Fig. 15 (a): All three methods combined. (b): Two out of three of the methods agree. (c): All three methods agree.

As a final step, a set of heuristics are applied in an effort to interpret the final line images. Multiple cases are considered and a direction is selected and sent to the navigation program. Examples of some of the possible cases include (1) two vertical parallel lines (drive straight), (2) left vertical and right horizontal lines (go right) and (3) two horizontal lines (use map history to choose right or left turn) etc. Ultimately, the navigation program chooses the actual path to travel, based on fusing the LADAR ranging data and image processing data.

3.3 Navigation Algorithms

A behavior-based programming architecture is used to implement vehicle navigation. The overall objective is to optimally fuse goal-following behavior as determined by the vision algorithms and obstacle-avoidance behavior as (predominantly) determined by the LADAR to provide a composite steering angle to the vehicle's motion controller. Some of the behaviors are meant to cover situations where the input data is uncertain and involve the use of fuzzy inference techniques. The structure of the adopted strategy is shown in Figure 16 and permits seamless incorporation of other behaviors in the future based on experience.

Essentially a number of algorithms are executed in parallel, each of which employs simple calculations and/or heuristics to determine a vehicle heading. Environment sensors and situation-based confidence measures are used to select a dominant behavior by subsuming other commands via the use of an Arbiter program. Inputs include compass heading, e-stop/remote control, lane, barrel, and pothole positions from image processing, general obstacle locations from the LADAR, and DGPS path history. Most of these inputs are accompanied by a statistically or heuristically determined confidence measure, which is used to help the Arbiter decide between the parallel algorithms. For example when there is high confidence in the quality of the lane and obstacle location inputs from the vision algorithm, Algorithm 1 determines the composite steering angle. When the vision results are less certain, Algorithm 2 is selected. It invokes fuzzy inference techniques to combine the IP and LADAR inputs to establish the resulting steering angle. Similarly, Algorithm 3 comes into play when sensors indicate that the vehicle has navigated into a trap. It invokes a ballistic behavior which uses the ability provided by the zero turning radius of THOR to turn around and exit the trap using a stored local map. Another feature of the navigation module is the incorporation of a fuzzy speed controller that suggests an appropriate speed for the vehicle based on the necessary turn angle and the closest object in front of it. When properly tuned, this feature enables the course traversal time for the vehicle to be optimized.
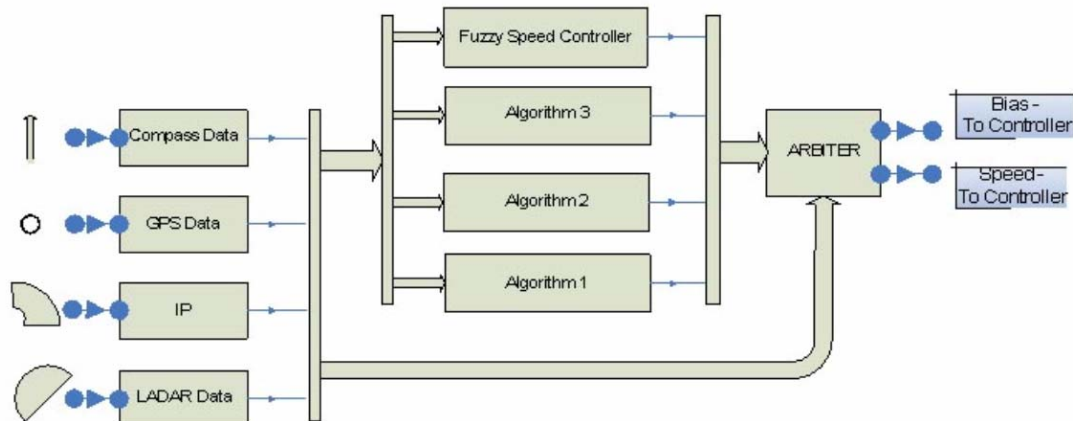
Figure 16: Arbitrated parallel navigation strategy

3.4 Results and Conclusions by University of Detroit Mercy

Many image sets where collected in a variety of lighting conditions to fine tune the effectiveness of the overall algorithm. An outdoor test course was also constructed to generate an additional image set. Using the test course image set (500 pictures), lanes were identified with at least 2 out of 3 confidence, 95% of the time. The remaining 5% were mostly images in which lanes were not identified at all, which is key because it is preferred that the program should fail by not returning anything, rather than returning lines that do not exist. Furthermore, the vehicle performed very well at the 2006 IGVC competition where it finished 4th amongst about 28 vehicles that participated in the Autonomous Design Challenge segment where vision was used.

The algorithm presented in this paper achieves a very high accuracy in lane identification over a wide variety of course situations. The algorithm could be enhanced by addition of shadow identification features. Also, further development of heuristics for applying specific image processing techniques to specific scenarios would be useful. Finally, a global map could be developed to provide an accurate means of reversing and retracing the course to escape trap situations.

4.0   FUTURE OF ARTICULATED VEHICLES

Articulated vehicles natural inherent mobility demonstrated in both manned and unmanned configurations lends itself well to the platform of choice for the future battlefields. Within the intelligent ground vehicle community articulation enables steering advantages for tight turns in an obstacle cluttered battlefield. TARDEC realized those

**Figure 17: 1985 TARDEC Concept Articulated UGV
engaging enemy armor with anti-tank weapon**

advantages in the 1980s with the JPL collaboration during CARD development. Requirements for autonomous performance as defined by the US Army Armor Center Combat Developments Directorate in 1983, led TARDEC to articulated battlefield UGVs such as the one shown in Figure 17.

Future articulated vehicles uses include a next generation EOD reconnaissance vehicle or a robot companion to police SWAT units. Articulated vehicles are inherently stable dealing with high slopes, steps and uneven terrain.

REFERENCES

1. www.igvc.org
2. "Lane Identification and Path Planning for Autonomous Mobile Robots", Robert T. McKeon, Mark Paulik, and Mohan Krishnan, SPIE Optics East 2006, Boston, MA, October 2006.
3. A. R. Bacha. "Line Detection and Lane Following for an Autonomous Mobile Robot." Master's Thesis, Virginia Tech, Blacksburg, Virgina. May 2005.