

SELF PROPELLED INTELLIGENT KINETIC ENTITY

The Citadel The Military College of South Carolina

Donald Buzanowski^{*}, Rawley Dent[†], Randall Evans[‡], Jesse Howard[§], Stephen Young^{**}
Dr. Siripong Potisuk (potisuks1@citadel.edu)

Design an autonomous ground vehicle to compete in the 22nd annual Intelligent Ground Vehicle Competition (IGVC). The IGVC helps to further research in unmanned, intelligent ground vehicles. The competition consists of three parts; vehicle design or enhancement, autonomous navigation through an obstacle course, and an interoperability profiles challenge demonstrating use of the JAUS communication protocol. Modifications were made to an existing six wheeled DC powered platform for the competition entry. There were minor mechanical changes with the majority of effort focused on enhancing the sensor and software integration and moving from AD* style environment mapping, path planning, and heading output, to real time obstacle detection and avoidance. The vehicle first determines its location then navigates through the obstacle course via sensory input from hardware mounted on the vehicle to its destination. The sensory inputs are integrated through control software to provide output to motor drives.

INTRODUCTION

The fully autonomous robot is the next step in the evolution of robotics. Today there are many competitions sponsored by both the private and defense sectors to further advances in this field. One such competition is the Intelligent Ground Vehicle Competition (IGVC). The IGVC's primary focus is the design of an autonomous robot always in contact with the ground, within specific dimensions, able to navigate to preset points within a bounded course avoiding obstacles and the design process involved.†† The Citadel Intelligent Ground Vehicle Solutions (CIGVS) team entry is the Self Propelled Intelligent Kinetic Entity (SPIKE). SPIKE is the project vehicle designed during the two semester capstone design course required for completion of an electrical engineering degree from The Citadel The Military College of South Carolina. As the team had little experience in robotics it was decided to start with a high level overview of what is expected per the contest rules and research the methodology and systems for an autonomous vehicle.

DESIGN PROCESS

During initial discussion for the vehicle it was decided to maintain the existing mechanical design. The focus would be on enhancing the sensory hardware and their functions, the software - algorithm processes and the IOP challenge. To evaluate the existing platform the team not only

* The Citadel School of Engineering dbuzanos@citadel.edu

† The Citadel School of Engineering rdent@citadel.edu

‡ The Citadel School of Engineering revans2@citadel.edu

§ The Citadel School of Engineering jhoward5@citadel.edu

** The Citadel School of Engineering syoung4@citadel.edu

†† www.igvc.org/rules.htm

worked to understand the operations of the vehicle as it was, but also met with members of past teams and asked for input into areas they saw as problematic. Multiple concerns were discussed during meetings with past team members and faculty and were used as reference points in decisions moving forward. The concerns included processing speed, vehicle speed, vehicle dimensions as related to course navigation, drive chains coming off, H-bridge / motor control replacement, and payload weight. Of these the most concern was expressed over processing speed-power as this seemed to affect multiple operations of the vehicle, possibly contributing to the speed over ground issues as well image processing used in line detection. The team tried to segment these into areas of the vehicle and how this would affect the project outline.

The outline has eight sections for development or enhancement and two for integration and testing as seen in Table 1. Three of the team members would focus on the vehicle and two would focus on the IOP challenge. While the outline was the guide for the project there were times the team needed to work tasks out of order while learning the Labview programming and understanding the sensor hardware.

Table 1. Project Outline

IGVC Robot
Requirements
Vehicle evaluation
OS / Software
Environment / Sensors
Positioning / Navigation
Mechanical
System Integration
Final Testing

Vehicle Requirements, Evaluation and Software

The first three sections were worked in parallel and focused on understanding the expectation of the vehicle per competition rules and understanding its past and current operational ability. It was determined that three high level decisions were needed to move forward in the design; the type of path planning algorithm used, the computer and operating system and the robot control software. These were decided based on the concerns over computer processing ability. These decisions would also factor into the types of input / output needed for the vehicle.

Path Planning. For the concept of the autonomous vehicle, by far, the most important decision must be path planning. The path planning approach dictates the sensory inputs and algorithms necessary for the final design. The methods researched included the A* algorithm, Simultaneous Localization and Mapping (SLAM), or design an original real time “move while detecting” algorithm.¹ The primary factor in using a real time algorithm was the concern over processing and real time avoidance seemed the least process intensive as compared with the previous AD* path planning.

OS and Control. With the top level decision of the path planning complete options were discussed for the computer and OS and the robot control software. Since processing was noted as a major concern the team started with the existing ASUS G75VW-DH72B laptop. All members agreed that the laptop configuration should be adequate to handle the needs. The next step was to look at the laptop operating system (OS) and the robot control software used and how it might affect the OS on

the laptop as well as investigate others available.

The existing configuration ran the Windows OS and Labview software. The alternatives were narrowed to using either Windows or Linux OS and running either Labview or Robot Operating System (ROS). The team agreed that the Windows OS can be process heavy but with limited Linux experience felt the learning curve would be time consuming to change to Linux. This choice also was a factor in choosing Labview over ROS as ROS is not currently available for Windows. Other factors for choosing Labview included its extensive process libraries, compatibility with much of the hardware preliminary discussions centered on and Labview is used in many robotics competitions.

It must be noted the team had no experience programming in Labview, but believed its graphical style language would mean a quicker learning time. Sub routines in Labview are called Virtual Instruments (VI) and can be run alone or built into a project.

VEHICLE ENVIRONMENT AND POSITIONING

With the decision made to use real time path planning the next discussions were on sensory input. To sense and provide reactive output to its environment is the underlying principal for any autonomous vehicle. This is accomplished through sensor hardware and their algorithms in the decision process. One broad constraint was to evaluate devices that would be readily available production units if possible. The team began with the fundamental premise the vehicle will need to locate and orient itself in its environment and detect obstacles and lines and their relationship to the vehicle. Initial discussion focused on what equipment would meet those needs. The existing sensors were evaluated and compared to other possible devices. A benefit during testing of the sensors was to isolate each sensor and develop its VI within the project VI in order to assess vehicle performance for each individual sensor. As a sensor was evaluated it was integrated into the project VI to isolate any sensor that may cause issues. The team chose to begin testing with navigation then move to obstacle avoidance then line and flag detection integrating sensors as their operation was understood and testing passed.

Navigation

The fundamental criterion for the vehicle is to move from a starting position to a given position of latitude and longitude or waypoint. Using that criterion it was decided that a GPS unit would be the best device to determine the starting position. Using rectangular to polar conversion the target distance and heading to the waypoint could be determined.

Two types of GPS units were researched. The first type, a packaged consumer unit that provided the ability to acquire a position fix, accepted variable input for waypoints and provided an output heading and continuous guidance to the waypoint. The second were chipset units that would acquire the position fix and send that data to a computer running software to provide needed output. The latter was chosen to provide more flexibility of data acquired. Compared to other units available the existing UBlox LEA-6H GPS unit was kept as it provided the GPS and antenna in one package, a level of accuracy of 2 meters and worked well with Labview. The GPS is connected to the computer via USB through the UBlox serial adaptor and uses the NMEA 0183 standard.²

With the vehicles initial position determined the next step in navigation was to determine the distance and heading to the waypoint. To do this several methods were investigated including the Haversine method.

$$a = \sin^2(\Delta\varphi/2) + \cos(\varphi_1) \times \cos(\varphi_2) \times \sin^2(\Delta\gamma/2) \quad (1)$$

$$c = 2 \times \tan^{-1} 2(\sqrt{a}, \sqrt{1-a}) \quad (2)$$

$$\varphi_1 = lat1 \times \frac{\pi}{180} \quad (3)$$

$$\varphi_2 = lat2 \times \frac{\pi}{180} \quad (4)$$

$$\Delta\varphi = (lat2 - lat1) \times \frac{\pi}{180} \quad (5)$$

$$\Delta\gamma = (long2 - long1) \times \frac{\pi}{180} \quad (6)$$

$$d = R \times c \quad (7)$$

Where d is the distance to target and R is earth's mean radius.

$$\theta_y = \sin(\Delta\gamma) \times \cos(\varphi_2) \quad (8)$$

$$\theta_x = \cos(\varphi_1) \times \sin(\varphi_2) - \sin(\varphi_1) \times \cos(\varphi_2) \times \cos(\varphi_1) \quad (9)$$

$$\theta = \text{atan2}(\theta_y, \theta_x) \times \frac{180}{\pi} \quad (10)$$

Where θ is the bearing to target using the atan2 function to account for upper and lower quadrants.

This was compared to a VI written to convert latitude and longitude to a decimal form providing rectangular coordinates. The rectangular coordinates are converted to polar to provide a GPS bearing and default velocity from the current location to the next waypoint with input for multiple waypoints in succession. Based on testing results the team ultimately chose the rectangular to polar conversion over the Haversine method. With the vehicle able to calculate where it is and a heading to its next point it needed to know its current orientation to those points.

With distance and bearing to the target determined a method was needed to provide the heading offset angle based on the vehicle orientation to the target angle. The team looked at several options including inertial navigation to determine vehicle orientation. Ultimately it was decided that the cost for a reliable inertial navigation system was not warranted and the existing Devantech CMPS03 would be sufficient for this task. The compass provides feedback as to the vehicles relation to true north to the Spartan 3E FPGA and that data is passed to the main project VI for target bearing calculation. Two TRDA-20 R1N100RZD encoders also providing data through the FPGA are used to provide control feedback for the motor control VI. During testing with the Devantech CMPS03 offset target headings began to wander. Once the compass was rewired and recalibrated problems were corrected and data was reliable.

Initially the vehicle operated by calculating the bearing of the current target GPS way-point. Using this bearing the current heading offset was calculated in order to determine the target angular and forward velocity. During testing it was realized that this target bearing would not allow the vehicle to travel a path in the direction opposite the GPS way-point. For example traveling away from the GPS way-point may be necessary in the occurrence of a switch back in the course or a dead-end. Upon further examination it was seen that using the current heading and current target GPS way-point bearing this inability could be eliminated. A probability function was generated for both the current bearing and the current target GPS way-point bearing which would allow the combination of the two to be weighted and then combined. This combination makes up the total probability function as seen in Figure 1. The maximum probability in this function is then set as the target bearing and velocity for the vehicle to the motor control VI. Once this bearing is set the vehicle traverses to this target until an obstacle is detected in the path to the target GPS way-point.

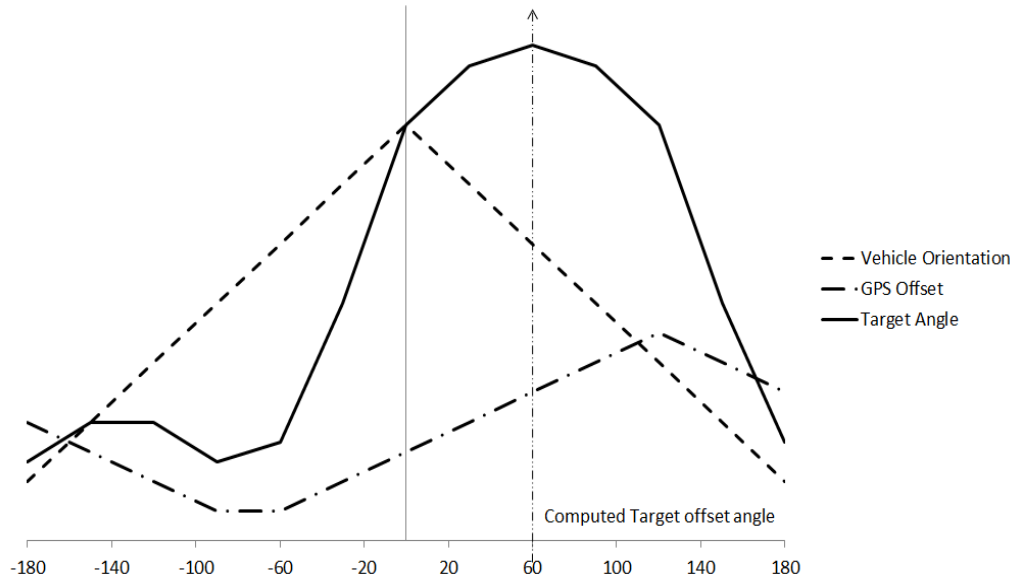


Figure 1. Example of target heading offset

Obstacle Avoidance

There were multiple options available for object detection to be used for the course obstacle avoidance. The options ranged from low cost sonar units to very high cost LiDar. The decision was made to keep the existing URG-04LX-UG01 LiDar. The unit connects to the system using a USB cable. The sensor searches in a clockwise sweep from -120° to $+120^{\circ}$ relative to the current heading. This provides a 240° field of view as shown in Figure 2.

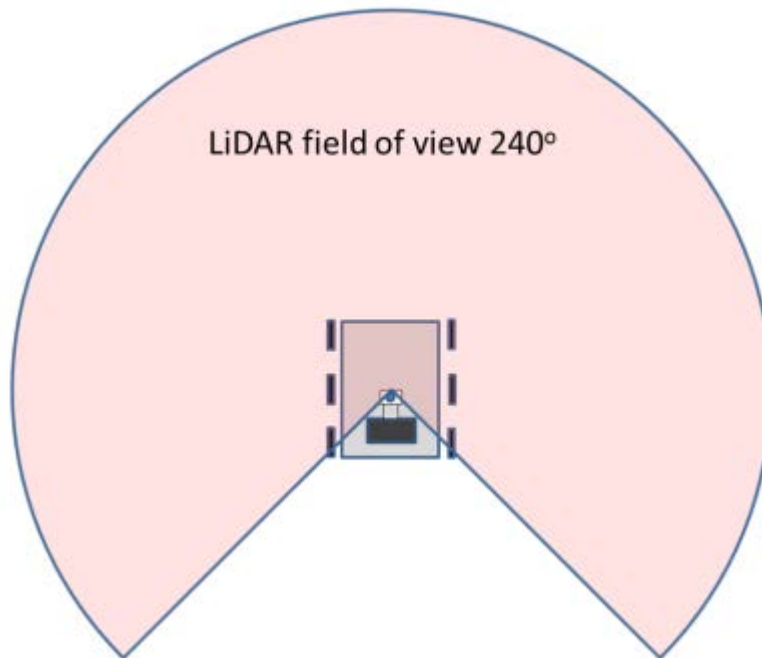


Figure 2. LiDAR field of view

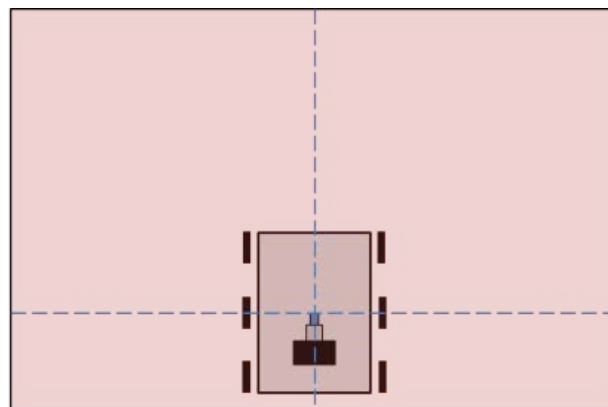
If an object is detected, the LiDar returns the magnitude and angle at which the detection occurred. The data acquired is in polar form for angle and distance to the object. This data is converted to rectangular coordinates relative to the center of the vehicle and stored in an array. The data in this array is then output by the LiDar VI. The location of the object is mapped in relation to the vehicles current heading. One issue discovered was that direct sunlight will affect the operation of the unit. To address this issue a sun guard was fashioned to block direct sunlight. The sun guard was deemed an acceptable solution to upgrading to an external unit due to cost

Lane Following – Line Detection

For the line detection an optical solution using a camera seemed the best option. With limited experience in image processing, research was conducted on camera options, and image processing using Labview.³ The existing PC88WR-2 camera was tested but on multiple occasions the entire project VI would freeze. It was believed this was associated with the cameras need for analog to serial conversion. The team also believed the cameras limited field of view would be a handicap especially if two cameras were used to increase the field of view. With the large number of camera and lens options available, the search focused on three areas; the need to mitigate processing power associated with imaging, a large field of view, and due to the limited image processing experience, a manufacturer whose camera and driver software are compatible with Labview.

Based on the concerns noted the choice was made for a new camera and lens combination. This would not only provide the greater field of view but offered several different communication protocols for faster data speeds. The protocols included GigE, firewire and USB 3.0. After researching documentation from National Instruments it was determined that USB 3.0 fit our application best.⁴ The benefits of the USB 3.0 camera include higher bandwidth and reduced processing requirements. The DFK 23UV024 with Verifocal T2Z 1816CS lens provided an increase in field of view from 92° to 175°, the best fit I/O, documentation, software compatibility and a reasonable price.

The camera provides the image capture data which then uses Labview vision acquisition and filtering software to provide a usable image to detect the white lines used to represent lane boundaries. The camera image is calibrated to determine distance and angle values to each pixel in rectangular coordinates. To eliminate unnecessary areas of the image, such as the vehicle itself, a mask is created assigning zeroes to the pixels in those areas to create the final field of view as seen in Figure 3.



Camera field of view after Labview filter
10 meter x 7 meter at origin

Figure 3. Camera field of view after Labview calibration and filtering

For use in line detection the 32 bit color image is converted to an 8 bit greyscale. The 8 bit greyscale is then filtered to a discrete image to determine the necessary white components of the image. With each pixel converted to zero for black or one for white each transition is populated as a point. A best fit line is then plotted to represent the real world boundary. The line data is then further processed in the obstacle VI.

ELECTRICAL

The main electrical supply for the vehicle is a 12 VDC system powered with a standard rechargeable marine deep cycle battery. The battery is connected to the system through a high amp main disconnect. The vehicles electrical system is segmented into two areas.

The primary side is connected through a 4-position fuse block with four 40-amp fuses to 4 relays. The relays are connected to the 4 CIM brushed drive motors two per side for redundancy. The relays are connected to the wired and wireless safety switches as required per the competition rules. The primary side also powers the auxiliary USB RAM 234 smart hub. The smart hub is an external powered USB interface between the laptop and hardware. This prevents those devices from putting an extra drain on the laptop battery. All components of the primary side except the motors are accessible from the top of the vehicle after removing the shaped acrylic weather guard. This allows those components to be quickly serviced or replaced as required.

The secondary 12 VDC side is connected to a Mini box M4-ATX DC to DC power supply added for the sensitive electronics. The device not only supplies more stable 12 VDC for sensitive devices but also steps down the voltage to 5 VDC and 3.3 VDC as needed. The secondary side provides power to the Spartan 3E FPGA board and PIC18F8722 microcontroller. The FPGA as discussed previously provides the logic circuits for the Devantech CMPS03 compass and TRDA-20 R1N100RZD encoders. The PIC18F8722 microcontroller is used for the Pulse Width Modulation (PWM) motor control to be discussed in the mechanical section. All components on the secondary side are easily accessed by lowering the hinged acrylic weather enclosure on the front of the vehicle. This also allows those components to be quickly serviced or replaced as required.

MECHANICAL

The vehicles overall dimensions are 3.2 feet long, 2.3 feet wide and 5.6 feet tall. It has 6 wheels 3 per side with the two center and two rear wheels powered by sprocket and chain drive with four 12VDC brushed motors 2 per side. The choice to maintain the current platform narrowed down several of the decisions associated with the competition as related to the physical vehicle size, mechanical operation and safety devices. The vehicle met all the size and drive system requirements, but the team felt due to the concerns given with the speed over ground, chain drive and H bridges we would need to look at the drive system more closely. The team first looked at the safety devices to insure they met requirements.

Safety

The safety devices include an operation light and two types of emergency stops. When the vehicle main power breaker is engaged the light is given constant voltage. When the autonomous mode is initialized a simple loop structure inside the main VI is enabled to flash the light through a timed routine.

The manual and wireless stops control four relays wired between each of the four drive motors that interrupt main power to the motors when activated. The manual stop is a simple button switch that will lock in the open position when pushed and is wired in series with a Linear DXR-702 wireless receiver that will open a contact if activated by the Linear DXT-21 wireless transmitter.

Drivetrain – Motor Control

The issues the team felt related to the drivetrain and motor control were speed over ground, drive chain derailing and H-bridge motor control overheating. The drivetrain consists of four CIM FR801-001 brushed motors. There are two motors per side connected to Andymark Toughbox gearboxes. The gearboxes have one output drive axle with two 9 tooth sprockets attached. The two axle sprockets on each side are connected by #35 chains to the 65 tooth sprockets on the rear and center wheels on each side. The four motors voltage input is controlled by four Victor 888 Speed controllers. Signals were sent to the motor controllers by two Velleman Pulse Width Modulators (PWM) one per side. The PWMs were controlled by input signals from the National Instruments Digital Acquisition Device DAQ 6008 using a motor control VI.

The team first looked the gearing ratios for speed verses torque and this seemed sufficient but the vehicle did exhibit issues with the skid steering. Next the existing Pulse Width Modulation (PWM) controls and the H-bridges were reviewed. The current configuration used two PWMs and the vendor specifications for these as used with the motor controls did not seem to provide enough resolution for speed control so the team chose to replace the two existing Velleman PWM controllers with a PIC18F8722 microcontroller. The change to microcontroller along with updates to the PWM coding for the motor control VI was an improvement, but there still seemed to be an issue with the skid steer. It was actually found that one of the gearboxes had an inside gearing issue. The gear box was removed and repaired solving the skid steer problem and vehicle speed over ground. It was also believed this may have been part of the issues with motor controllers overheating due to overloading the motor on one side.

Next the team investigated the chain derailing. It was noticed that due to the size of the hubs of the two 9 tooth sprockets on the drive axle the chain to the rear wheel was partially riding on the hub of the sprocket going to the center wheel. Due to the length of the axle there was no room for adjustment so replacement sprockets with smaller hubs were researched. While looking for new hubs there were several incidents when the chain did derail during operation. Unable to find readily available sprockets the team opted to replace the current axle with a longer axle with the same diameter. Once the axle was replaced there were no more instances of chain derailment.

INTEGRATION AND FINAL TESTING

The final undertaking of the project was to integrate all the tasks from the design process into the autonomous control algorithm. As discussed the control algorithm would center on real time sensory input to provide the real time control output.

Sensor Integration to Path Planning

With all the sensors and their functionality defined they were integrated into the control routine-VI. The control VI initializes the sensor subroutine VIs. The GPS and compass as discussed determine the initial position and the offset target angle and velocity. Velocity is defined as the voltage input to the motor control VI. With this input the vehicle begins to traverse the course with the LiDAR and camera scanning their field of views for obstacles and lines. When the LiDAR detects an object in its field of view the angle and magnitude of the objects data points populate the obstacle VI array. Should the vision subroutine detect a variation in the ground contrast within defined parameters it uses the edge detection VI as previously discussed to determine the angle and magnitude of data points, which also populate the obstacle VI array. The obstacle subroutine is designed to compute the distance between detected obstacles - physical and lines – determine which areas have room enough for the vehicle to travel then determine the least deviation to original offset heading to target. The process runs continuously until the target waypoint is reached per GPS reading. Unfortunately two issues arose during integration testing. The first was a shadow effect with

the LiDar data points and the second the sensor output data from the camera proved too volatile for the control algorithm.

Obstacle Avoidance Issues

The LiDar obstacle avoidance problem occurred when two objects closer to the vehicle but separated by a distance too narrow for the vehicle to traverse shadowed a third whose distance from those objects was an opening large enough for the vehicle to traverse as seen in Figure 4. The calculation of consecutive distance values returned by the LiDar in a counter-clockwise direction caused the appearance of traversable paths by the distance calculated between obstacles and designated as openings 1 and 2 in Figure 4. Though these paths are large enough for the vehicle to traverse the true path does not exist due to the nonconsecutive data points in close proximity designated as path 3 in Figure 4. The distance between these two closer obstacles would in reality be impassable.

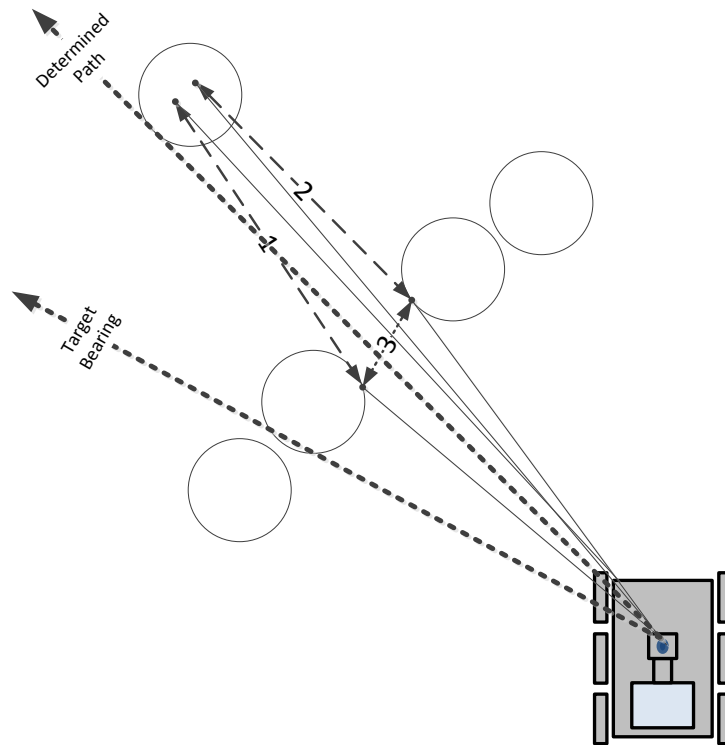


Figure. 4 Obstacle Avoidance Shadow

The second issue related to intermittent pixel values creating false edge detection which output false data into the obstacle array which output erratic control values to the motor control VI. The data from the camera was treated as obstacles and thus input to the obstacle avoidance algorithm VI. The data from the image processing proved to be quite volatile and varied greatly based on lighting conditions and the position of line detection intervals. These issues created even more issues for this process. Based on these findings it was decided to attempt to use some of the original concepts from this process to come up with a new method to make decisions.

Astar

With the research done on the various path planning methods it was realized that the A* method may be utilized to account for the errors experienced in the obstacle avoidance algorithm. The same sensor data can be used in the A* algorithm. Traditionally the A* algorithm operates on a user defined grid subdivided into nodes. These nodes are defined as certain levels of cost by the user. The grid could encompass anywhere from the entire course down to the vehicle's immediate operating area. The number of nodes would be defined by the resolution required for accurate obstacle and line placement. An open path is defined as a traversable node (low cost) and obstacles are defined as non-traversable nodes (high cost). In order to implement A* environmental sensors provide feedback to assign cost to the nodes. Given a starting node and a target node A* then explores the cost associated with every node route to its target and then chooses the least cost route. An example of the typical A* grid is seen in Figure 5. The previous team's data was kept and continuously built upon for the entire run of the course. Given the concerns over processing power the team chose to use data from real time sensor feedback with A* and use to create a smaller grid that refreshes with more real time data.

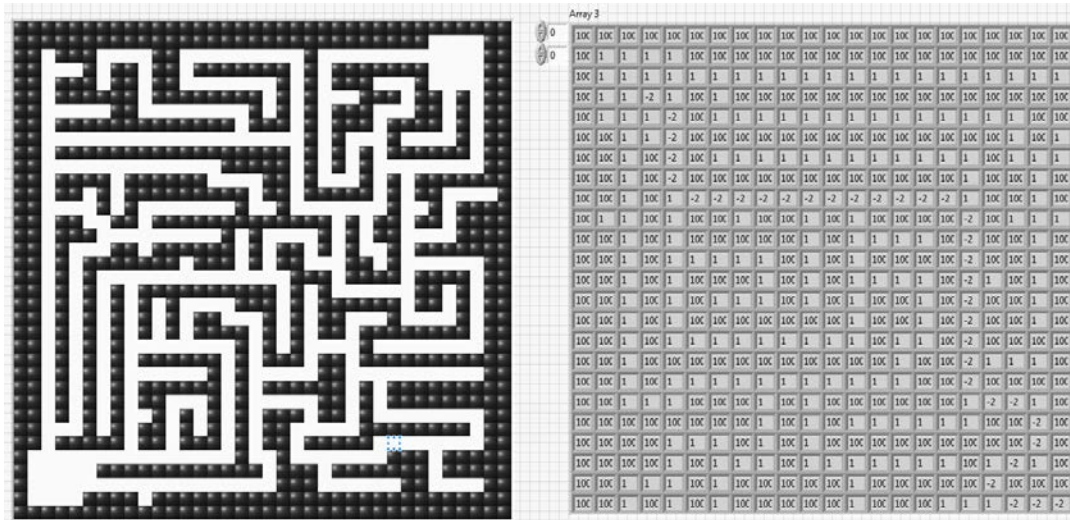


Figure 5. Typical A* Grid with nodes and control array

For the team's implementation of A* a user defined grid is generated with dimensions of 10 meters by 7 meters with this grid divided into 100 cm² nodes. By greatly reducing the grid size and updating with real time data the processing efficiency of the control algorithm was greatly increased. The grid is defined in the (x, y) plane from (0, 0) to (99, 69). The relative A* starting point and origin is defaulted to the center of the vehicle at (50, 20). The target heading offset angle generated by the GPS and compass probability functions creates a dynamic endpoint on the edge of the grid. The LiDAR obstacle distance and angle data is now converted to rectangular coordinates. The x and y coordinates are used to populate the grid with high cost nodes. The camera data, which is already in x and y coordinates, is also used to populate the grid with high cost nodes. It was decided to create a threshold around the existing obstacles and lines in order to reduce the errors from the shadow effect and inconsistent lines errors discussed earlier. All nodes within the specified threshold of obstacles-boundaries are also set to high cost. All other nodes are considered low cost and the grid is passed to the A* algorithm. The path chosen by the algorithm is returned as an array of x and y positions on the grid. These values generate a linear trend line which is used to calculate the corrected path to avoid obstacles. An example of this process is shown in Figure 6.

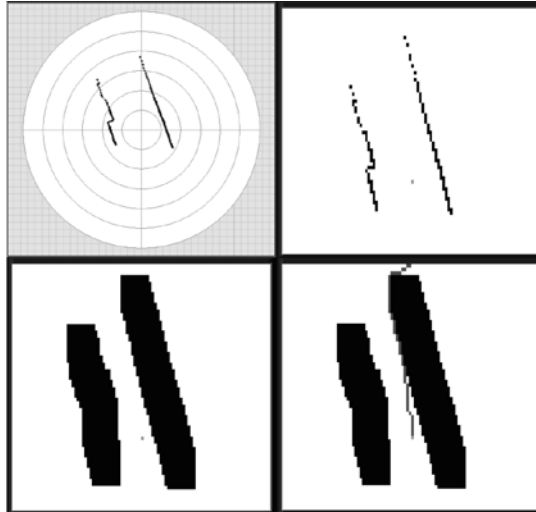


Figure 6. Local real time A* grid

This path is then used to determine the target forward and angular velocity while obstacles and/or lines are detected in the path of the vehicle. The target forward and angular velocity is then input into a motor control VI to output the voltage levels for the motors.

BILL OF MATERIALS

See Appendix A

FACULTY STATEMENT

See Appendix B

CONCLUSION

The design of the autonomous vehicle crosses multiple disciplines including mechanical engineering, electrical engineering, and computer science. The needed environmental feedback with today's advances in electronics and computing is readily achievable at a reasonable cost. Location, orientation and movement of the vehicle can be accomplished with the use of a GPS, compass and encoders. Detection of objects in its path with LiDar provides very reliable feedback. With the use of readily available software for image processing, the ability to detect contrast in images provides reliable line detection but a good background in digital signal processing (DSP) is an advantage. The benchmark of the project is the integration of the provided environmental feedback into reliable control. For this integration and the autonomous vehicle to succeed the most emphasis must be placed on the path planning algorithm.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the assistance of Mr. Bart Knapp, Mrs. Carmela Trez and guidance of Dr. Siripong Potisuk, Dr. Harold Askins, and Dr. Johnston Peeples. The past teams CLETIS, Hawk-eye, and ZERGLING, with special thanks to members Mr. Michael Lacey, Mr. Jonathan Hagar, and Mr. Kendall Nowicin. We would also like to thank the following for their financial contributions to the project: The Citadel, Google and SAIC.

REFERENCES

¹Giorgio Grisetti, Rainer Kummerle, Cyrill Stachniss, Wolfram Burgard, "A tutorial on graph-based SLAM," *Intelligent Transportation Systems Magazine, IEEE*. vol. 2, Issue 4, pp. 31-43, winter 2010

² NMEA Standard 1083, Nov. 2008

³ Christopher G. Relf. (2003, July 28). *Image Acquisition and Processing with Labview* [online]. Available: www.crc.press.com

⁴ National Instruments, "NI Announces USB 3.0 Camera Support with NI Vision Acquisition Software August 2013," Mar. 2013

APPENDIX A: BILL OF MATERIALS

Team	Div	Component	Distributor	Part Number	Quantity	Price Per [\$]	Total Amount [\$]
P	AF	Plate, Aluminum	Grainger	3drz6	1	\$ 39.10	\$ 39.10
P	AF	Paneling, Lexan	Lowes	IPC0036A	1	\$ 70.00	\$ 70.00
P	AF	DC to DC Power Converter mount	McMaster-Carr	M5x 9.94	4	\$ 1.25	\$ 5.00
P	AF	Fastener	McMaster-Carr	M5x 14.29	2	\$ 2.50	\$ 5.00
P	AF	Fastener	McMaster-Carr	M6x 20.64	2	\$ 2.50	\$ 5.00
P	AF	Fastener	McMaster-Carr	M3x 17.46	2	\$ 2.50	\$ 5.00
P	AF	Fastener	McMaster-Carr	M5x 9.94	4	\$ 1.25	\$ 5.00
P	AF	Main disconnect mount	McMaster-Carr	M6x 20.64	2	\$ 2.50	\$ 5.00
P	AF	Mount, Computer	McMaster-Carr	M5x 16	4	\$ 1.25	\$ 5.00
P	AF	Mount, misc.	McMaster-Carr	M3x 9.53	14	\$ 0.36	\$ 5.04
P	AF	Mount, Motor Controller	McMaster-Carr	M3x 22.23	8	\$ 0.63	\$ 5.04
P	AF	Mount, Power Block	McMaster-Carr	M5x 14.29	2	\$ 2.50	\$ 5.00
P	AF	Mount, Toughbox	McMaster-Carr	M6x 55.56	8	\$ 0.63	\$ 5.04
P	AF	Mount, Vesa	McMaster-Carr	M4x 12	8	\$ 0.63	\$ 5.04
P	AF	Connectors, 45x45mm	Rexroth	93543-07	40	\$ 3.00	\$ 120.00

P	AF	Extruded aluminum, 45x45mm	Rexroth	74351-22	25	\$ 3.00	\$ 75.00
P	BD	ToughBox gearbox	Andymark	AM-0145	2	\$ 88.00	\$ 176.00
C	BD	Toughbox Long output shaft	Andymark	am-0211	2	\$ 13.00	\$ 26.00
P	BD	Bearings	Igus	EFOM-BB1-P12-101	12	\$ 12.00	\$ 144.00
P	BD	Motors	Innovation First Intl, Inc.	CIM FR801-001	4	\$ 28.00	\$ 112.00
P	BD	Axle, 3/8"	McMaster-Carr	94624A535	6	\$ 26.94	\$ 161.64
P	BD	Drivechain	Northern Tool	136410	12	\$ 1.00	\$ 12.00
P	BD	Wheels w/ Sprocket	Northern Tool	13340	4	\$ 40.00	\$ 160.00
P	BD	Wheels w/o Sprocket	Northern Tool	136411	2	\$ 10.00	\$ 20.00
C	BD	3 wire cable controller to PWM (4pack)	Vex Pro	276-1976	1	\$ 19.95	\$ 19.95
P	BD	Victor Motor Controller	Vex Pro	217-2769	4	\$ 69.99	\$ 279.96
P	C	Button, Stop	DataCommElectrical	3SB35001HA20	1	\$ 32.50	\$ 32.50
P	C	RF Receiver	Linear DX	DXR-702	1	\$ 48.00	\$ 48.00
P	C	Lights, Warning Amber	McMaster-Carr	8720T561	2	\$ 3.79	\$ 7.58
P	C	Xbox wireless USB receiver	Micosoft	xc14	1	\$ 12.99	\$ 12.99
P	C	Xbox wireless controller	Microsoft	B4F00014	1	\$ 54.99	\$ 54.99
P	E	Connectors, crimp	Altex	65-4342	16	\$ 1.39	\$ 22.24
P	E	Connectors, crimp 1/4"	Altex	65-4342C	1	\$ 12.09	\$ 12.09
P	E	Connectors, crimp ring #10	Altex	65-1546C	28	\$ 7.99	\$ 223.72
P	E	Disconnect, Automotive Ckt Brkr	Fastenal	Buss CB185-120	1	\$ 93.39	\$ 93.39
P	E	Wirenut	Home Depot	30-451	1	\$	\$

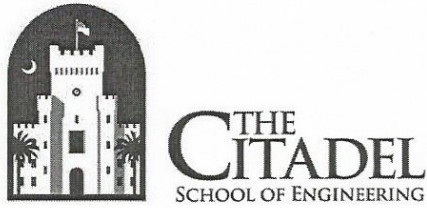
						7.42	7.42
P	E	Power Cube	McMaster-Carr	HNG-90	4	\$ 4.00	\$ 16.00
P	E	DC to DC Power Converter	Mini-Box	M4-ATX	1	\$ 102.56	\$ 102.56
P	E	Relay	SonicElectronix	Omega AU-7	4	\$ 1.99	\$ 7.96
P	E	Battery	West Marine	1231109	2	\$ 150.00	\$ 300.00
P	M	Spacers assorted	Amazon	M3 Nylon Spacer	1	\$ 10.54	\$ 10.54
C	M	Marking flags Red (100pk)	Grainger	3LUK2	1	\$ 15.95	\$ 15.95
C	M	Marking flags Blue (100pk)	Grainger	3LUK4	1	\$ 16.85	\$ 16.85
C	M	Board, Wooden	Home Depot	POPLR1/2x64- 3PL	1	\$ 7.92	\$ 7.92
C	M	striping paint white (6pak)	Lowes	76666	1	\$ 26.85	\$ 26.85
P	S	µBlox GPS	3D Robotics	LEA-6H	1	\$ 89.99	\$ 89.99
P	S	GPS Adapter	3D Robotics	GS406/GS407	1	\$ 17.90	\$ 17.90
P	S	Encoder, Wheel	AutomationDirect	TRDA-20 R1N100RZD	2	\$ 109.00	\$ 218.00
P	S	Lidar	Hokuyo	URG-04LX- UG01	1	\$ 1,174.00	\$ 1,174.00
P	S	Webcam	Logitech	C310	1	\$ 29.99	\$ 29.99
C	S	Perfboard	RadioShack	2760158	1	\$ 3.49	\$ 3.49
P	S	Compass, Magnetic	RobotShop	Deventech CMPS03	1	\$ 68.18	\$ 68.18
P	S	Camera	Super Circuits	PC88W R-2	1	\$ 100.00	\$ 100.00
C	S	USB 3.0 Cable	The Imaging Source	CA-USB30- AmB-BLS/1.5	1	\$ 21.00	\$ 21.00
C	S	USB 3.0 Color Camera	The Imaging Source	DFK 23UV024	1	\$ 318.40	\$ 318.40
C	S	Vari-focal Lense	The Imaging Source	T2Z 1816 CS	1	\$ 216.00	\$ 216.00
P	SH	Laptop	Asus	G75VW-DH72B	1	\$ 1,749.95	\$ 1,749.95
P	SH	Cord, USB	CablesToGO	#27329	6	\$	\$

						11.99	71.94
C	SH	PIC 18 Microcontroller Board	Microchip	DV164136	1	\$ 165.00	\$ 165.00
P	SH	LabVIEW 2012	National Instruments	776678-35	1	\$ 4,300.00	\$ 4,300.00
P	SH	LabVIEW Robotics Module	National Instruments	781220-35	1	\$ 2,000.00	\$ 2,000.00
P	SH	Multifunction DAQ	National Instruments	USB-6008	1	\$ 280.00	\$ 280.00
C	SH	External battery charger computer	Porta Power Technology	PLT-A42- G75WP	1	\$ 75.00	\$ 75.00
P	SH	USB Hub	RAM Mounts	RAM-234- HUB1U	1	\$ 82.97	\$ 82.97
P	SH	Spartan 3E FPGA board	Xilinx	XC3S500E	1	\$ 199.00	\$ 199.00
P	SP	ToughBox gearbox	Andymark	AM-0145	1	\$ 88.00	\$ 88.00
C	SP	PWM Signal Driver	Vex Pro	PWM- DRIVERPWM Signal Driver	1	\$ 14.95	\$ 14.95
C	SP	Victor Motor Controller	Vex Pro	217-2769	1	\$ 69.99	\$ 69.99
C	SP	Compass, Magnetic	Zagros	Deventech CMPS03	1	\$ 58.80	\$ 58.80
						TOTAL	\$ 13,907.91

Team	P - Previous builds	12851.76
	C - Current Build	1056.15
	total	<u>13907.91</u>

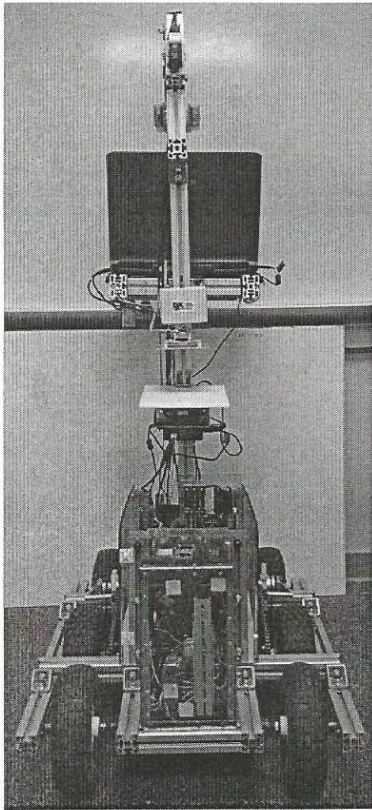
Div	AF - Frame body	364.26
	BD - Drivetrain	1111.55
	C - Control	156.06
	E- Electrical	785.38
	M - Miscellaneous	78.11
	S - Sensor	2256.95
	SH - Software/Hardware	8923.86
	SP - Spare	231.74
	total	<u>13907.91</u>

APPENDIX B: FACULTY STATEMENT



SPIKE

Design Report Self Propelled Intelligent Kinetic Entity



Team CIGVS

Donald Buzanowski
Randall Evans
Rawley Dent
Jesse Howard
Stephen Young

I certify that the engineering design of the vehicle described in this report has been significant and equivalent to each team member earning six semester hours of design credit for their work on this project.

Dr. S. Potisuk

A handwritten signature in blue ink, which appears to read "Suz Potisuk", written over a horizontal line.